

Analysis and modeling of human driving behavior in highway merging areas using a combined approach of stochastic methods and artificial intelligence

Xinzhe Zhang^{a,*}, István Varga^{a,b}, Tamás Tettamanti^{a,b}

^a Department of Control for Transportation and Vehicle Systems, Faculty of Transportation Engineering and Vehicle Engineering, Budapest University of Technology and Economics, Műegyetem rkp. 3, Budapest, H-1111, Hungary

^b Systems and Control Laboratory, HUN-REN Institute for Computer Science and Control, Kende u. 13-17, Budapest, H-1111, Hungary

ARTICLE INFO

Keywords:

Human driving behavior
Highway merging areas
Dynamic merging window
Bayesian Gaussian mixture model
Hierarchical Bayesian inverse reinforcement learning

ABSTRACT

The paper investigates the driving behavior patterns of mainline human-driven vehicles during highway ramp-merging interactions. Using the exiD dataset, a dynamic merging window was defined in time and space to identify mainline vehicles with potential interactions with ramp vehicles. Using the neighboring-vehicle information for each ramp vehicle within the merging window, one-to-one merging events between mainline and ramp vehicles were extracted. To identify heterogeneous driving behavior patterns within the merging area, a Bayesian Gaussian Mixture Model was employed for cluster analysis of driving behaviors. Subsequently, the Hierarchical Bayesian Inverse Reinforcement Learning framework was adopted to infer global and cluster-specific bias weights that represent relative reward preferences across behavioral patterns. Afterward, using reinforcement learning, the driving behavior trajectories of human-driven vehicles in each cluster were reproduced in SUMO (Simulation of Urban MObility), providing a more comprehensive representation of lateral and longitudinal driving behavior. Finally, the exiD-trained policy was deployed in a high-fidelity SUMO merging scenario representing the Hungarian M1 highway. The results showed stable closed-loop execution and supported the cross-scenario portability of the framework. Overall, this study presents a configurable empirical framework for on-ramp merging analysis that connects traffic interaction representation, human driving behavior heterogeneity, data-driven preference inference, and closed-loop traffic simulation.

1. Introduction

Future transportation systems are expected to feature mixed traffic composed of Human-Driven Vehicles (HDVs) and Autonomous Vehicles (AVs). Mixed traffic involving HDVs and AVs is not just a future concept anymore. It is a reality in certain regions, e.g., Waymo's deployments in the United States. Automated traffic systems offer several clear advantages, including optimizing traffic flow, reducing emissions, and assisting vulnerable travelers. However, there are significant differences between HDVs and AVs, including the modeling of driving behavior, decision-making interactions, and asymmetric information structures. These differences may lead to conflicts, especially at critical traffic spots, such as highway merging areas, where higher vehicle speeds, combined with shorter reaction times, increase the likelihood of accidents (Péter et al., 2025). Therefore, simulating the driving behavior of HDVs has become a popular topic. In general, the safety and efficiency of mixed traffic flows depend strongly on the proper modeling of HDVs.

Driving behavior classification can be divided into multiple categories based on the different types and the amount of available

driving data. However, it is often difficult to capture the complexity of drivers. A driver's age (Kuniyoshi et al., 2021; Chu, 2020), personality (Bucsuházy et al., 2020; Zhai et al., 2023; Tselentis and Papadimitriou, 2023), or habitual behaviors (Sayed et al., 2022; Chen and Zhang, 2026) can impact traffic and even contribute to traffic accidents. Scholars have so far classified driving styles in several ways, e.g., from the behavioral performance (Houston et al., 2003), through the context-personality interaction model (Abou-Zeid et al., 2011), or from a technological viewpoint leveraging image processing (Shahverdy et al., 2020). The mathematical approach of driver behavior modeling tries to abstract the behavioral decisions of HDVs into interpretable formulas and frameworks. Its main purpose is to explain and predict the driving trajectories and decisions of HDVs on the traffic network at a mathematical abstraction level. Traditional formula-based models are simple, interpretable, and computationally efficient, so they are important in traffic behavior modeling. However, driving behavior often shows clear heterogeneity and interaction dependence. When one fixed model structure or one shared parameter set is used for all

* Corresponding author.

E-mail address: zhangxinzhe@edu.bme.hu (X. Zhang).

drivers, it may not fully capture the diverse and overlapping behavior patterns in natural driving data. This limitation is especially relevant in highway ramp-merging scenarios, where merging behavior is jointly shaped by the merging vehicle and nearby mainline vehicles. The speed, lateral position, and gap response of mainline HDVs directly influence merging safety, efficiency, and trajectory evolution. Therefore, to better understand and reproduce the heterogeneous responses of mainline HDVs during merging interactions, this study focuses on highway merging areas and builds a configurable empirical framework based on naturalistic exiD trajectory data. The framework first extracts mainline vehicle merging behavior windows, then uses probabilistic behavior clustering to identify heterogeneous behavior patterns. It further learns cluster-specific relative reward preferences and finally uses reinforcement learning to reproduce the corresponding mainline vehicle trajectories. The framework links traffic interaction analysis, human driving behavior interpretation, data-driven inference, and closed-loop traffic simulation. This framework aims to learn and reproduce the observable merging behavior of mainline HDVs. The resulting behavioral representations may provide empirical inputs for future mixed-traffic simulation and AV merging-strategy studies. To clarify the research basis of this study, the following section reviews related studies on behavior clustering and learning-based methods.

The classification methods in the relevant literature, based on statistical features, mainly conduct statistical analysis on dedicated vehicle dynamic indicators in the HDVs' natural driving dataset, such as lateral and longitudinal acceleration, relative speed, or lane-changing frequency. In these studies, the mean, median, variance, and standard deviation in the analysis output are typically used to explain differences and regularities in driving behavior. [Murphey et al. \(2009\)](#) and [Itkonen et al. \(2017\)](#) analyzed the mean, standard deviation, and quantiles of the driver's jerk curve. [Feng et al. \(2017\)](#) defined three classification methods for driving styles based on speed, distance to the preceding vehicle, and approaching an accident using the jerk indicator. [Eboli et al. \(2016\)](#) introduced a modified "g-g diagram" composed of lateral and longitudinal acceleration and speed to determine the safety threshold for driving behavior and classify driving styles.

Besides the statistical approach, machine learning technology, having the capability of analyzing large amounts of data for pattern prediction, can also be exploited for driving behavior modeling. For the classification and cluster analysis of driving behavior, some researchers have conducted extensive exploration of mainstream algorithms. [Yang et al. \(2022\)](#) compared relevant clustering and classification techniques and proposed an improved cross-validation mean squared error model based on driving behavior responses. The results showed that k -means clustering and the decision tree model have the highest accuracy. [Chen et al. \(2023\)](#) established a comprehensive assessment model for driving behavior risks, also confirming the good performance of k -means clustering in driving behavior classification. In addition, the k -means algorithm can also be applied to the identification of driving behavior patterns in construction areas ([Khanfar et al., 2022](#)), signal light intersections ([Khanfar et al., 2023](#)), or heavy trucks ([Zhu et al., 2022](#)). The k -means algorithm-based clustering, however, simplifies the driving behavior. It requires the manual specification of the k value, introducing subjective factors. Moreover, the initial cluster center points of k -means are randomly selected, which easily leads to a local optimal solution.

In terms of addressing the issues of heterogeneity and uncertainty in driving behavior, the Bayesian Gaussian mixture model (BGMM) clustering method holds promising research value. BGMM introduces prior distributions into a Gaussian mixture model (GMM) to represent complex and overlapping data. With enough candidate components, the Bayesian prior reduces the weights of unnecessary components and retains those supported by the data. It also provides posterior membership probabilities, which can represent uncertainty for samples located near overlapping cluster boundaries. These properties make BGMM suitable for driving behavior patterns with overlapping distributions

and uncertain cluster boundaries. In the transportation and automotive domain, it has been used in problems such as inferring braking behavior ([Wang et al., 2018](#)), driver identity authentication ([Yang et al., 2018](#); [Jafarnejad et al., 2018](#)), or improving the perception ability of on-board radar sensors ([Walenta et al., 2024](#)).

Regarding the simulation and prediction of driving behavior, [Yin et al. \(2021\)](#) designed a two-layer memory-attention hierarchical model. The first layer applied a hidden Markov model to identify driving behaviors, and the second layer adopted a memory-attention recurrent neural network framework, both of which were used for motion trajectory prediction. [Tang et al. \(2019\)](#) combined Fuzzy C-means (FCM) with an Adaptive Neural Network to predict the lane-changing trajectories of vehicles. A data-driven approach is also a promising research direction for fitting driving behavior models. This approach can provide flexible alternatives to traditional physics-based models for driving behavior simulation ([Himmels et al., 2024](#)). Using the natural driving dataset, methods such as the maximum likelihood estimation method ([Dai et al., 2022](#)) and inverse reinforcement learning ([Liu and Wu, 2020](#); [Alsaleh and Sayed, 2020](#)) were employed to design a state space model that constructs a utility function with good similarity and regularity when reproducing traffic patterns, thereby more realistically simulating driving behaviors.

Inverse Reinforcement Learning (IRL) has been widely used to model and interpret human driving behavior in transportation studies ([Huang et al., 2023](#); [Bhattacharyya et al., 2022](#); [Wang et al., 2024](#)). [Xu et al. \(2024\)](#) used the CitySim dataset to jointly describe the car-following and lane-changing behaviors of human drivers in highway scenarios through IRL and linear reward functions. [Chandra et al. \(2022\)](#) utilized a data-driven approach to learn the entropy risk-related preference of HDV behaviors through IRL, and then embedded it into a risk-sensitive planner within the framework of game theory to achieve the risk-aware interaction between HDVs and AVs. [Wu et al. \(2022\)](#) designed the behavior evaluation function for HDVs and used IRL to simulate driving behaviors. The semi-Markov model was applied to classify the different driving intentions of surrounding vehicles.

Based on the above literature review, driving behavior patterns are typically continuous, ambiguous, and highly uncertain. The k -means clustering algorithm is widely used in various fields. However, in the problem of driving behavior classification, k -means, as a hard clustering algorithm, will assign sample points to a certain fixed category, which overly simplifies the driving behavior and makes it difficult to explain and reflect the uncertain driving behavior of the driver; the artificially specified k value introduces subjective factors; and the initially randomly selected cluster center points are prone to causing local optimal solutions. In the problem of modeling driving behavior, although LSTM and Transformer models can achieve good prediction performance, their generalization may depend on the training data and target scenario. In addition, their predictions do not directly provide an interpretable representation of the latent decision preferences associated with different driving behaviors.

In recent years, with the rapid development of the drone field and image detection algorithms, high-precision drone photography data of roads has provided excellent input for clustering and inverse reinforcement learning. The BGMM clustering algorithm can distinguish different driving behavior tendencies and model the uncertainty of driving behaviors. Meanwhile, Hierarchical Bayesian Inverse Reinforcement Learning (HB-IRL) can infer the global weights of the reward function and the bias weights of each cluster from the observed behavior data. This method provides an interpretable representation of the relative decision preferences associated with observed HDV behaviors and their traffic states. Together, these two methods identify heterogeneous driving behavior patterns and infer their relative decision preferences. The automatic learning of weights can reduce the subjective influence introduced by manual design, making it more closely aligned with human behavior. Collecting large-scale trajectory

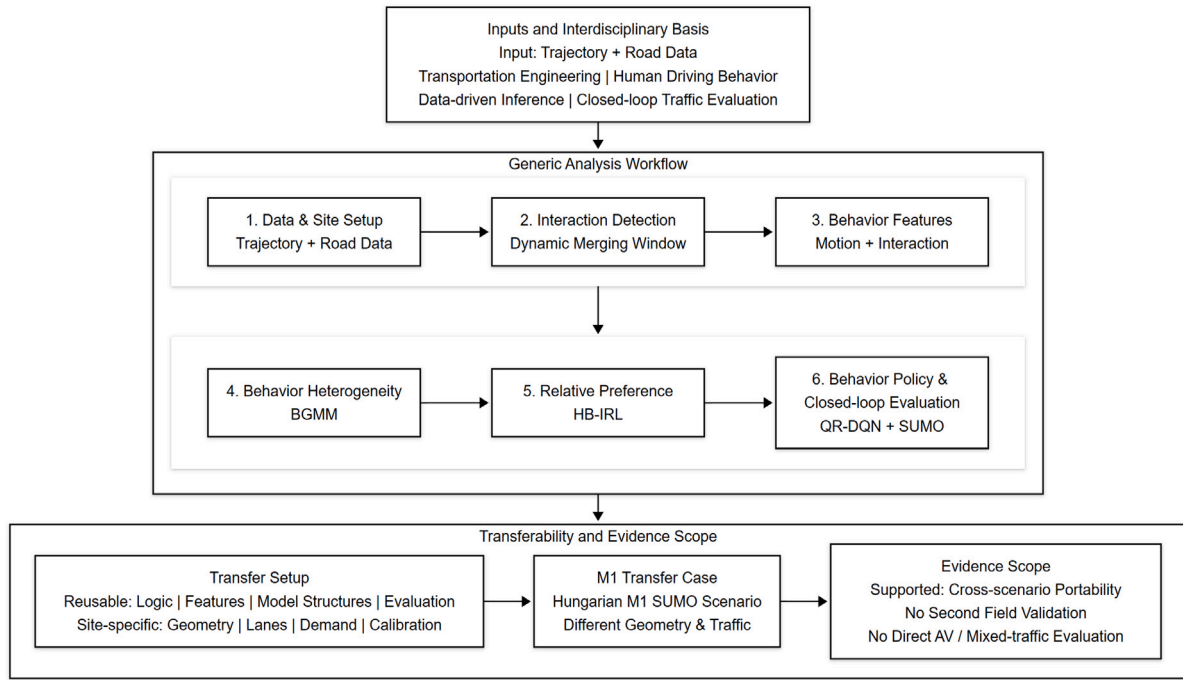


Fig. 1. Generic interdisciplinary framework, cross-scenario transfer, and evidence scope of the proposed method.

data can increase the coverage of less frequent behaviors, which can more comprehensively reproduce driving behaviors.

According to the relevant literature and related research gaps introduced above, the main contributions of this paper are highlighted below:

1. A dynamic merging window was designed, and the key areas for merging were determined.
2. BGMM is used to identify heterogeneous and overlapping driving behavior patterns, avoiding the use of a single homogeneous behavior representation in the subsequent HB-IRL analysis.
3. By using the HB-IRL method, the global weights of the reward function and the cluster-specific bias weights are learned, and the relative reward preferences of different driving behavior patterns are inferred.
4. Using SUMO (Lopez et al., 2018) traffic simulation, human driving behaviors are reproduced through RL using Traffic Control Interface (TraCI) (Wegener et al., 2008). Online posterior sampling for RL is conducted within the weight space to quantify the uncertainty of HDVs. Overall, the model's performance indicators showed reasonable agreement with the real data.
5. The exiD-trained QR-DQN policy is further deployed in a high-fidelity SUMO merging scenario representing the Hungarian M1 highway under different road geometry and traffic conditions. Stable closed-loop execution supports the cross-scenario portability of the proposed framework.
6. The proposed framework does not use AI models as stand-alone tools. It connects traffic interaction analysis, human driving behavior interpretation, data-driven inference, and closed-loop traffic evaluation.

The paper is organized as follows. First, the main methodologies employed in this paper are introduced in Section 2. Then, in Section 3, the main focus is on dataset preprocessing, merging event detection, the clustering results of the BGMM, the result analysis of the HB-IRL model, and the HDVs' driving behavior reproduced using RL. The results are evaluated against real-world data, and an additional scenario is used to examine the cross-scenario portability of the framework. In Section 4, the opportunities and limitations of applying the model to real-world

scenarios are discussed. Finally, Section 5 summarizes the study's main findings and outlines future research directions. All the field-specific terms and abbreviations are provided in this paper in Appendix C, Table C.1.

2. Methodology

2.1. Generic and interdisciplinary framework for empirical on-ramp merging analysis

This study presents a configurable empirical framework for analyzing human driving behavior in highway on-ramp merging. As shown in Fig. 1, the framework connects transportation engineering, human driving behavior analysis, data-driven inference, and closed-loop traffic evaluation. The AI models are integrated into this workflow rather than used as separate tools. The exiD naturalistic trajectory data are used as the field-data implementation, and the Hungarian M1 SUMO scenario is used later to demonstrate cross-scenario portability.

The framework starts with vehicle trajectory data and road information. The data are first checked and standardized, and merging events and interacting vehicles are then identified. Behavioral features are constructed from the observed vehicle motions. BGMM identifies heterogeneous behavior patterns, while HB-IRL infers their relative reward preferences. QR-DQN then learns the related behavior policies, and SUMO closed-loop simulation evaluates the dynamic feasibility of the learned human driving behaviors.

The framework separates reusable analytical steps from site-specific settings. As shown in Table 1, data quality rules, interaction identification principles, behavioral features, model structures, and the evaluation process can be reused. Road geometry, lane IDs, coordinate mapping, merging-zone boundaries, traffic demand, and data fields need to be adjusted for the target site. Model retraining or parameter calibration may also be needed for a new dataset.

In this study, the exiD data provide the main field-based empirical evidence. The M1 SUMO scenario is used to examine whether the same workflow can be reconfigured and executed under different road geometry and traffic conditions. Therefore, the M1 case supports the cross-scenario portability of the framework, but it is not an external

Table 1
Reusable and site-specific components of the proposed framework.

Framework stage	Main purpose	Reusable core	Site-specific setup
Data & site setup	Prepare trajectory and road data	Data quality rules	Data fields, road geometry, lane IDs, coordinates
Interaction detection	Identify merging interactions	Dynamic merging-window logic and vehicle matching	Merging-zone boundaries and local thresholds
Behavior features	Represent motion and interaction	Feature definitions	Unit conversion and normalization
BGMM	Identify behavior heterogeneity	Model structure and probabilistic membership	Candidate components and model retraining
HB-IRL	Infer relative reward preferences	Reward-learning structure	Model retraining and parameter calibration
QR-DQN	Learn executable behavior policies	Policy-learning structure	Training with target-site data, if required
SUMO evaluation	Evaluate behavior in closed loop	Evaluation logic and metrics	Road network, traffic demand, and simulation settings

validation based on a second independent field-trajectory dataset. Direct AV control and mixed-traffic system performance are also outside the scope of this study.

2.2. Merging event detection

In the vehicle trajectory dataset, the primary task is to identify potential interaction behaviors between mainline vehicles and ramp vehicles. The potential interaction behavior refers to the situation where the merging of the ramp vehicles can affect the normal driving of some mainline vehicles. Meanwhile, the mainline vehicles need to take feedback actions, and an interaction behavior exists between the two. Another situation is that during the period when the mainline vehicles passed through the “study section”, no ramp vehicles entered, and there was no obvious interaction between the two. To distinguish these two types of mainline vehicles, the merging event detection process includes three steps: determining the merging point, defining the dynamic merging time window, and identifying mainline vehicles involved in potential interaction (Fig. 2).

The key nodes during the merging of ramp vehicles were defined based on vehicle dynamics indicators, including the merging point (t_{merge}), the start frame of merging (t_{start}), and the end frame of merging (t_{end}). For details, refer to Appendix A, Table A.1. To establish the complete intended time range for the ramp HDVs to merge into the mainline, the merging time $[t_{start}, t_{end}]$ is extended to $[t_{intent}, t_{stabilized}]$ (Eqs. (1) and (2)). This extended time range can fully reflect the entire process time of ramp HDVs from having the intention to merge to starting the execution, and then to ending the execution (Fig. 3); it can also reflect that the mainline vehicles with potential interaction intentions will make predictions about the traffic environment before the ramp vehicles start to change lanes (t_{start}), and have the intention to take necessary actions in the lateral and longitudinal directions. Δt_1 and Δt_2 are dynamically defined based on the current speed of each ramp vehicle (Eq. (3)). The dynamic time window provides a consistent interaction interval for the subsequent HB-IRL analysis.

$$t_{intent} = t_{start} - \Delta t_1, \quad (1)$$

$$t_{stabilized} = t_{end} + \Delta t_2, \quad (2)$$

$$t_{DynamicTimeWindow} = \left[t_{start} - \frac{2w_{lane}}{\max(v_{pre}, 5)}, t_{end} + \frac{w_{lane}}{\max(v_{post}, 5)} \right], \quad (3)$$

where w_{lane} is the lane width. v_{pre} and v_{post} represent the average speed during the pre-lane-change and post-stabilization phase, respectively.

2.3. Modeling Bayesian Gaussian mixture model

In this study, a Finite Bayesian Gaussian Mixture Model (FB-GMM) is combined with a Dirichlet Process (DP), and a large value of K is set to approximate a DP-GMM. This approach can represent complex driving behavior data and reduce the weights of unnecessary

components when a sufficiently large candidate K is specified. The Bayesian prior also provides regularization during model fitting. At the same time, it uses the standard variational inference framework of FB-GMM to simplify the complex inference process in DP-GMM. This method combines the advantages of Bayesian inference with the adaptive capabilities of non-parametric models.

First, we assume that there are N samples (Eq. (4)), and the mixture weight vector $\pi = (\pi_1, \pi_2, \dots, \pi_K)$ follows a Dirichlet distribution with a parameter of α (Eq. (5)). For each sample x_n , there will be a latent cluster assignment variable z_n , which shows which specific cluster $z_n \in \{1, \dots, K\}$ generated the observation (Eq. (6)).

$$\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}, \quad \mathbf{x}_n \in \mathbb{R}^D, \quad (4)$$

$$\pi \sim \text{Dirichlet}(\alpha), \quad \alpha = \left(\frac{\alpha}{K}, \dots, \frac{\alpha}{K} \right), \quad (5)$$

$$z_n | \pi \sim \text{Categorical}(\pi), \quad z_n \in \{1, \dots, K\}, \quad (6)$$

where α controls the level of sparsity, and K represents the number of candidate mixture components.

The advantage of BGMM is that it assumes the k th Gaussian component is sampled from “a certain distribution”, which reflects the prior of that component. The prior distribution can be adjusted using hyperparameters (m_0, β_0, v_0, W_0). Given the precision matrix Λ_k (Eq. (7)), the mean vector μ_k of the k th Gaussian component follows a conditional normal distribution (Eq. (8)). Once the component is determined, sampling can be done from the corresponding Gaussian distribution. For example, when $z_n = k$, the sample x_n is drawn from the corresponding Gaussian distribution (Eq. (9)).

$$\Lambda_k \sim \text{Wishart}(v_0, W_0), \quad (7)$$

$$\mu_k | \Lambda_k \sim \mathcal{N}\left(m_0, (\beta_0 \Lambda_k)^{-1}\right), \quad (8)$$

$$\mathbf{x}_n | z_n = k, \mu_k, \Lambda_k \sim \mathcal{N}(\mu_k, \Lambda_k^{-1}), \quad (9)$$

where Λ_k is the precision matrix ($\Lambda_k = \Sigma_k^{-1}$) and follows a Wishart prior. v_0 controls the degrees of freedom. W_0 represents a positive definite matrix. m_0 and $(\beta_0 \Lambda_k)^{-1}$ are the mean and covariance, respectively, and μ_k is the mean vector of the k th Gaussian component.

Second, the mean-field assumption in variational inference is used to approximate the posterior $p(\pi, \mu, \Lambda, \mathbf{Z} | \mathbf{X})$ in this study, and it assumes that the variational distribution factorizes into three mutually independent components: $q(\pi)$, $\prod_{k=1}^K q(\mu_k, \Lambda_k)$, and $\prod_{n=1}^N q(z_n)$.

$$q(\pi, \mathbf{Z}, \mu, \Lambda) = q(\pi) \prod_{k=1}^K q(\mu_k, \Lambda_k) \prod_{n=1}^N q(z_n). \quad (10)$$

The variational posterior of the mixture weights π is modeled by a Dirichlet distribution with parameter γ , as shown in Eq. (11). The posterior distribution of the component parameters was approximated by the mean μ_k and the precision matrix Λ_k (Eq. (12)). Regarding the posterior distribution of the approximate cluster assignment variable z_n , it is assumed that a categorical distribution with parameter ϕ_n is

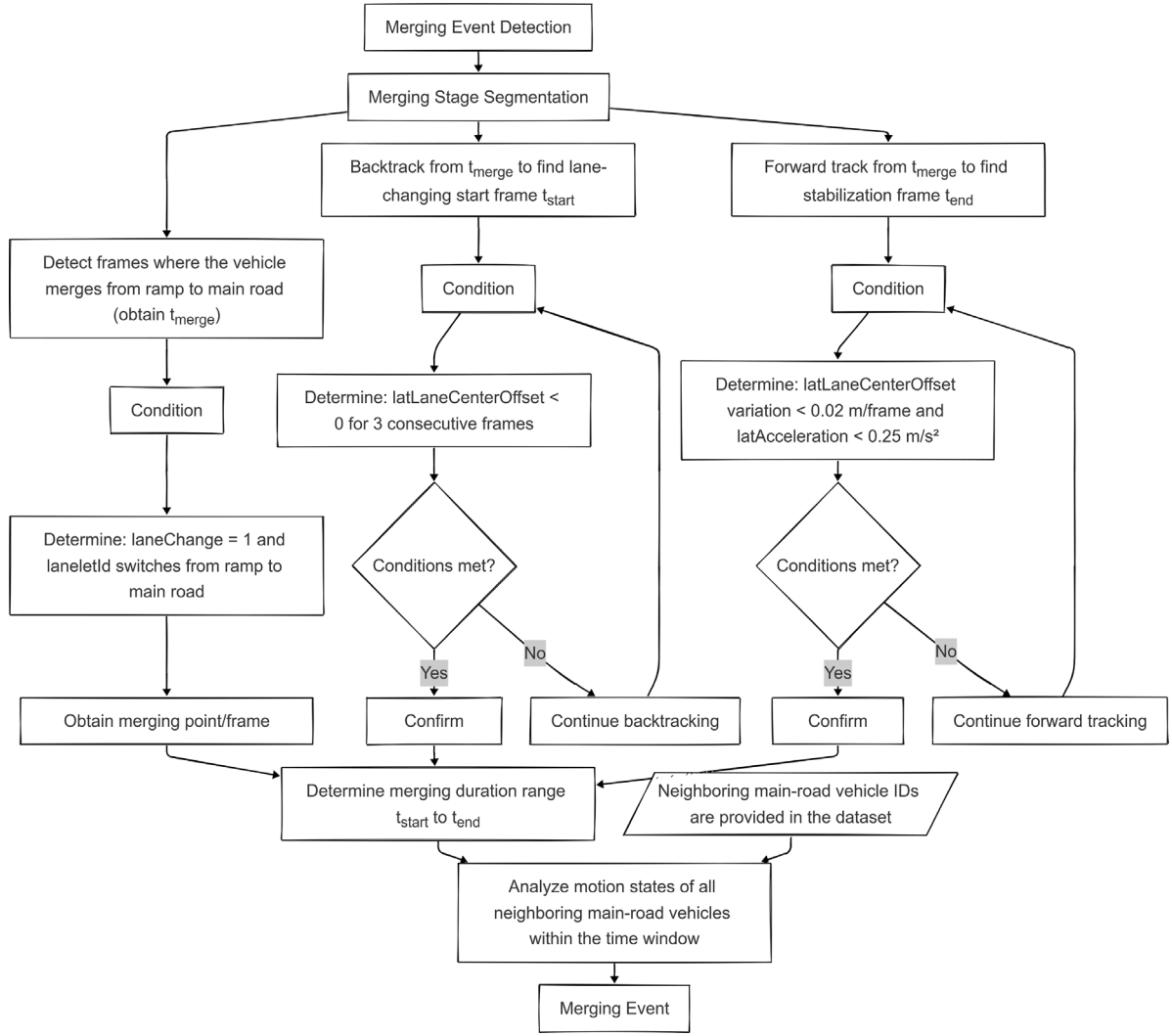


Fig. 2. Merging event detection.

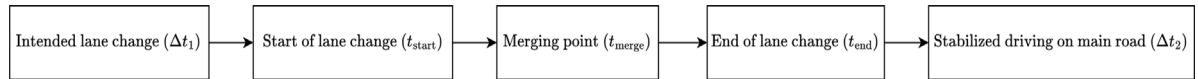


Fig. 3. Full time window.

used to make a variational approximation of the probability that a sample point belongs to the k th cluster. The main goal is to find a variational approximation for the z_n of each sample point x_n (Eq. (13)).

$$q(\pi) = \text{Dirichlet}(\gamma), \quad \gamma = (\gamma_1, \dots, \gamma_K), \quad (11)$$

$$q(\mu_k, \Lambda_k) = \mathcal{N}(\mu_k | m_k, (\beta_k \Lambda_k)^{-1}) \times \text{Wishart}(\Lambda_k | \nu_k, W_k), \quad (12)$$

$$q(z_n = k) = \phi_{nk}, \quad \sum_{k=1}^K \phi_{nk} = 1, \quad (13)$$

where $\mu_k | \Lambda_k \sim \mathcal{N}(m_k, (\beta_k \Lambda_k)^{-1})$ represents the conditional distribution of the mean and precision matrix. m_k is the posterior mean of μ_k , and ϕ_{nk} denotes the posterior probability that sample x_n belongs to cluster k .

To update the posterior probability of the sample point x_n belonging to the k th cluster, the coordinate ascent method is used for the variational update. Eq. (14) consists of two parts: the expected value of the weight term $\mathbb{E}_q[\ln \pi_k]$ (Eq. (15)) and the expected value of the Gaussian term $\mathbb{E}_q[\ln \mathcal{N}]$ (Eq. (16)). Meanwhile, Eq. (17) is used to update the

parameters γ of the Dirichlet distribution.

$$\phi_{nk} \propto \exp \{ \mathbb{E}_q[\ln \pi_k] + \mathbb{E}_q[\ln \mathcal{N}(x_n | \mu_k, \Lambda_k^{-1})] \}, \quad (14)$$

$$\mathbb{E}_q[\ln \pi_k] = \psi(\gamma_k) - \psi \left(\sum_j \gamma_j \right), \quad (15)$$

$$\mathbb{E}_q[\ln \mathcal{N}] = \frac{1}{2} \left(\psi_D(\nu_k) + \ln |W_k| - \frac{D}{\beta_k} - \nu_k (x_n - m_k)^T W_k (x_n - m_k) \right), \quad (16)$$

$$\gamma_k = \frac{\alpha}{K} + \sum_{n=1}^N \phi_{nk}, \quad (17)$$

where $\psi(\cdot)$ denotes the digamma function. $\psi_D(\nu_k)$ is the digamma function related to the Wishart distribution. $\{m_k, \beta_k, \nu_k, W_k\}$ are the parameters of the Normal-Wishart variational distribution. W_k is the scale matrix. D is the data dimension, and $(x_n - m_k)^T W_k (x_n - m_k)$ is the Mahalanobis distance.

For the k th cluster, the cumulative sufficient statistics will be repeatedly used in the subsequent parameter updates (Eq. (18)). The Normal-Wishart parameters in the variational distribution are updated

using Eqs. (19) and (20).

$$N_k = \sum_n \phi_{nk}, \quad \bar{\mathbf{x}}_k = \frac{1}{N_k} \sum_n \phi_{nk} \mathbf{x}_n, \quad S_k = \sum_n \phi_{nk} (\mathbf{x}_n - \bar{\mathbf{x}}_k)(\mathbf{x}_n - \bar{\mathbf{x}}_k)^\top, \quad (18)$$

$$\beta_k = \beta_0 + N_k, \quad m_k = \frac{\beta_0 m_0 + N_k \bar{\mathbf{x}}_k}{\beta_k}, \quad \nu_k = \nu_0 + N_k, \quad (19)$$

$$W_k^{-1} = W_0^{-1} + S_k + \frac{\beta_0 N_k}{\beta_0 + N_k} (\bar{\mathbf{x}}_k - m_0)(\bar{\mathbf{x}}_k - m_0)^\top, \quad (20)$$

where N_k represents the effective number of samples in component k based on the posterior weights ϕ_{nk} , $\bar{\mathbf{x}}_k$ denotes the mean of samples in cluster k under the weight ϕ_{nk} , S_k is the weighted scatter matrix of component k , m_k represents the updated mean, β_k is updated precision coefficient, and ν_k denotes degrees of freedom.

2.4. Hierarchical Bayesian inverse reinforcement learning modeling

This study uses Hierarchical Bayesian Inverse Reinforcement Learning (HB-IRL), which is a Bayesian variant of IRL. It estimates a posterior distribution over the reward function, which helps to reflect the uncertainty in the accuracy of the reward estimation. The original version of Bayesian IRL was introduced by Ramachandran and Amir (2007). It uses a Markov Chain Monte Carlo (MCMC) based algorithm to sample multiple times from the posterior, and the final reward function is estimated as the average of these samples.

2.4.1. Non-centered hierarchical Gaussian prior

In IRL, the prior distribution refers to the subjective beliefs or experiences assigned to the expert behavior dataset before analyzing and studying the expert's behavior. To achieve a more expressive and interpretable prior, this study adopted the idea of a non-centered hierarchical Gaussian distribution. It is modeled by dividing the prior weights into a global shared part and specific deviations for each cluster. This approach captures both the overall structure of the data and accounts for individual differences. This method enhances the generalization ability and sampling efficiency of the model, while reducing the phenomenon of overfitting. Specifically, the global weights control the overall complexity, while the bias term adjusts the differences between different clusters.

$$w_0 \sim \mathcal{N}(0, \sigma_w^2 \cdot I), \quad (21)$$

$$b_k \sim \mathcal{N}(0, \sigma_b^2 \cdot I), \quad (22)$$

where w_0 is the global weight, b_k represents the weight bias for cluster k . σ_w^2 denotes the prior variance of the global weight, and σ_b^2 is the prior variance of the bias term.

The bias b_k is parameterized in a non-centered form using a normal noise variable (z_k) and the prior standard deviation (σ_b).

$$z_k \sim \mathcal{N}(0, I), \quad b_k = \sigma_b \cdot z_k, \quad w_k = w_0 + \sigma_b z_k, \quad (23)$$

where w_k represents the sum of the global weight w_0 and the cluster-specific bias b_k .

This can be written in the form of a probability density function as:

$$P(w_0) \propto \exp\left(-\frac{1}{2 \cdot \sigma_w^2} \|w_0\|^2\right), \quad (24)$$

$$P(b) \propto \exp\left(-\frac{1}{2 \cdot \sigma_b^2} \sum_{k=1}^K \|b_k\|^2\right), \quad (25)$$

where $P(w_0)$ and $P(b)$ are the probability density functions of the global weights and the bias terms, respectively.

To make the calculation easier, the probability density functions of both are written in log form:

$$\log P(w_0) \propto -\frac{1}{2 \cdot \sigma_w^2} \|w_0\|^2, \quad \log P(b) \propto -\frac{1}{2 \cdot \sigma_b^2} \sum_{k=1}^K \|b_k\|^2. \quad (26)$$

2.4.2. Reward function

In the research of the transportation field, since a certain driving action is more dangerous/safer under a certain condition, in this study, the natural driving dataset is constructed as a state-action pair, and the reward function is established as a "state-action dependent type". According to the non-centered hierarchical Gaussian prior, the mathematical expression forms of the weight and reward function of cluster k are as follows (Eqs. (27) and (28)).

$$w_k = w_0 + b_k, \quad (27)$$

$$R_k(s, a) = w_k^\top \phi(s, a), \quad (28)$$

where $R_k(s, a)$ denotes the reward value for the current state-action pair in cluster k . $\phi(s, a)$ is the feature vector of the state-action pair, used to measure its value, and w_k represents the reward-weight vector for cluster k .

2.4.3. Q-value calculation

For each state, an action value is calculated for every possible action. The value represents the relative preference for an action under the current state.

$$Q(s_n, a_i) = \phi(s_n, a_i)^\top \cdot w_k, \quad (29)$$

where s_n denotes the state of the n th sample. a_i is the i th action in the action set. $\phi(s_n, a_i)$ represents the feature vector.

2.4.4. Softmax likelihood function

In this study, human drivers move through highway merging areas. These areas have high speeds and short reaction times. In such situations, the expert's decisions may not be fully rational and might not be the best possible choices. However, experts still tend to move toward actions that bring higher rewards. Their action choices are more likely to follow high-reward strategies. The softmax likelihood function can be used to model expert behavior that is not fully rational but still tends to choose actions with higher rewards. Softmax also allows for multiple possible actions in the same situation, with each action given a certain selection probability (Eq. (30)). The Softmax log-likelihood function (Eq. (31)) calculates the log-probability of the expert choosing a certain action based on the current environment and the given reward weight vector w , thereby evaluating whether the choice is reasonable.

$$P(a_i | s_n, w) = \frac{\exp(Q(s_n, a_i))}{\sum_{a'} \exp(Q(s_n, a'))}, \quad (30)$$

$$\log P(a_i^* | s_n, w) = Q(s_n, a_i^*) - \log \sum_{a'} \exp(Q(s_n, a')), \quad (31)$$

where $P(a_i | s_n, w)$ represents the probability of choosing action i given the current state s_n and weight w . a' covers all possible actions, and a_i^* denotes the expert's actual action in the n th data sample.

However, due to the high-dimensional space and the nonlinearity of the softmax function itself, the posterior distribution does not have an analytical solution. Usually, it can only be sampled and approximated through inference methods.

2.5. Sampling method

Sampling is the process of drawing representative samples from a posterior distribution. Since the posterior distribution cannot be calculated directly, a representative set of posterior samples is drawn to approximate the distribution and estimate the reward function. Due to the high-dimensional parameter space, diverse behaviors, and noise characteristics of the natural driving dataset, the sampling method combining Hamiltonian Monte Carlo (HMC) and Parallel Tempering (PT) was applied in this study.

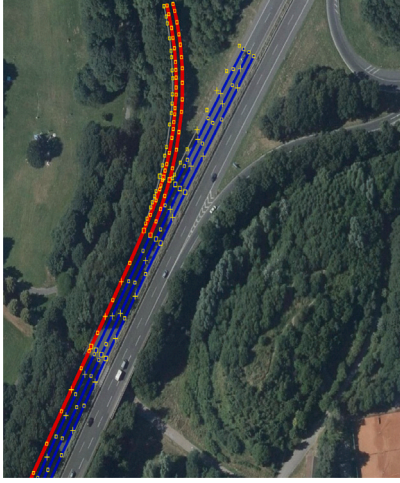


Fig. 4. Ramp and mainline lanelets in the study area of Location 2 in the exiD dataset.

3. Real-world data-based experiments

3.1. Dataset description and preprocessing

Our study focuses on analyzing and modeling one-way merging behavior in the highway merging area. To assess our methodology, we apply the real-world exiD dataset of a highway on-ramp area in Germany (Moers et al., 2022), as shown in Fig. 4.

This study first summarizes the road structure and traffic conditions covered by the selected exiD subset. Table 2 presents the main information of recordings 39–52 from exiD Location 2. This study area is a typical highway on-ramp merging scenario, with two mainline lane groups and one ramp lane group. After the required data preprocessing, the final dataset contains 4014 mainline vehicles, 1430 ramp vehicles, 1422 merging events, and 3997 mainline–ramp interaction pairs. The detailed preprocessing method is provided in Appendix A, Table A.3.

From the traffic-state perspective, the mainline traffic volume proxy ranges from 743.2 to 1161.6 veh/h across different recordings. The ramp traffic volume proxy ranges from 210.4 to 433.0 veh/h, and the merging event rate ranges from 210.4 to 424.9 events/h. These results show that the selected subset does not represent only one traffic demand level. Instead, it covers different traffic states under the same highway merging geometry. It should be noted that the road geometry of this subset is mainly limited to exiD Location 2. Therefore, this study does not generalize the conclusions to all road structures and traffic demand conditions. To further test the model's transferability in a different scenario, this study also introduces a Hungarian M1 SUMO highway merging test scenario in the following experiments.

After preprocessing, the total number of vehicles in the merging direction (study area) was 5444, and the effective trajectories accounted for 99.8% of the total vehicle trajectories. Moreover, the distribution of vehicles on the mainline and those on the ramps was reasonable. Table 3 provides a summary of the statistical results of the preprocessing in the merging direction.

3.2. Rationality check and statistical distribution of the dynamic merging window design

Fig. 5 presents the time axis of the entire life cycle (in gray) of three randomly selected ramp vehicles and the dynamic merging time window (in blue), as well as the distribution of each time point. The dynamic merging window is located in the middle-left part of the vehicle's travel process, and the $[t_{intent}, t_{merge}]$ time range is reasonable, fully capturing the interaction stage of ramp vehicles from intention through merging to stabilization.

By establishing a mechanism for detecting merging events, most of the mainline vehicles that had no interaction with the ramp vehicles were filtered out (Table 4). This reduced the amount of irrelevant information, allowing the research to focus more on the key merging events.

Finally, based on the neighboring vehicle information of the current ramp vehicles in the exiD natural driving dataset (Appendix A, Table A.2), the mainline vehicles that have potential interactions with the ramp vehicles within the merging window are identified. Each merging interaction event is treated as an independent sample for subsequent study.

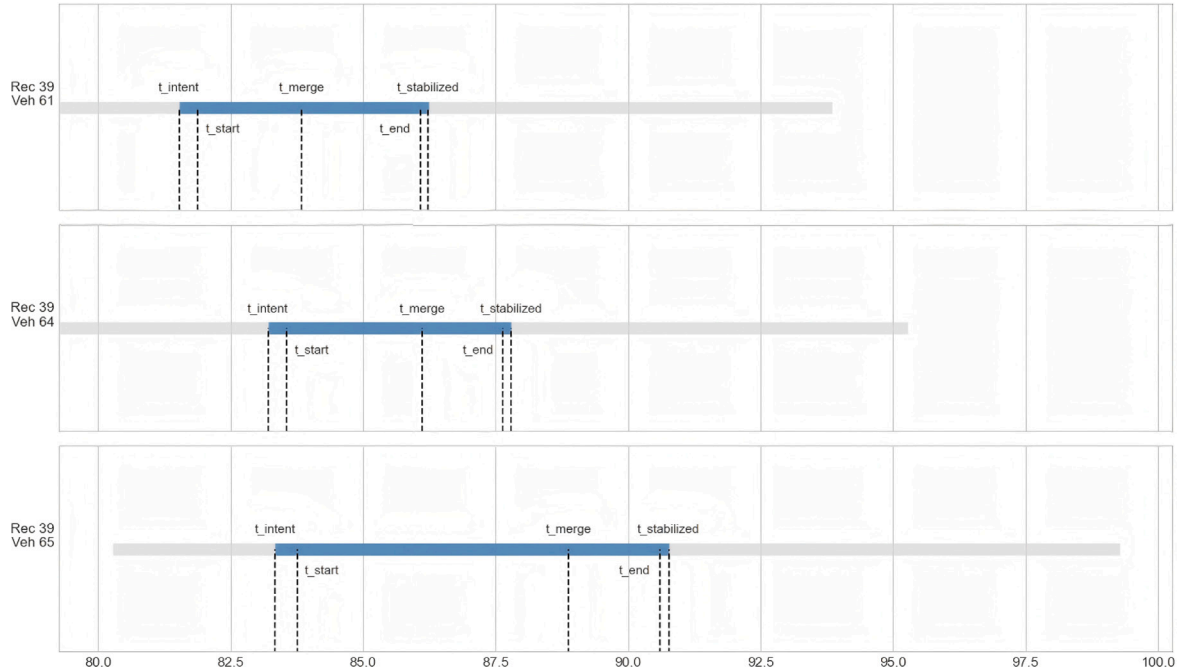


Fig. 5. Dynamic merging time windows.

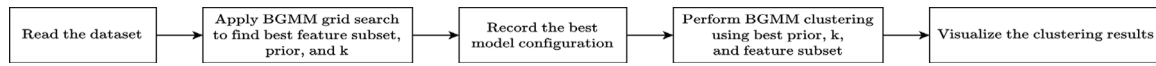


Fig. 6. BGMM workflow diagram.

Table 2

Summary of the dataset subset, road geometry, traffic state, and merging process.

Category	Metric	Value
Dataset subset	Recordings	39–52
Dataset subset	Location	exiD Location 2
Road geometry	Road type	Highway on-ramp merging area
Road geometry	Mainline lanelet groups	2
Road geometry	Ramp lanelet group	1
Sample size	Mainline vehicles	4014
Sample size	Ramp vehicles	1430
Sample size	Merge events	1422
Sample size	Interaction pairs	3997
Traffic state	Mainline volume proxy	743.2–1161.6 veh/h
Traffic state	Ramp volume proxy	210.4–433.0 veh/h
Traffic state	Merge event rate	210.4–424.9 events/h
Traffic state	Mainline speed	28.46 ± 0.74 m/s
Traffic state	Ramp speed	19.13 ± 1.10 m/s
Merge process	Dynamic window duration	6.64 ± 0.28 s

3.3. BGMM clustering analysis

To describe the different behaviors of drivers, the BGMM algorithm was designed to conduct cluster analysis on all merging interaction events (Fig. 6). To achieve a clear clustering division, the feature engineering was first conducted. A total of 12 features, including vehicle lateral and longitudinal characteristics, as well as interaction features between mainline vehicles and ramp vehicles, were selected to form a feature set. The features were normalized, standardized, and one-hot encoded to eliminate the influence of data scale and discrete data. The feature selection and processing methods are presented in Appendix B, Table B.1.

3.3.1. Correlation heatmap analysis

To select a subset of weakly correlated features from the initial feature set listed in Appendix B, Table B.1, a correlation coefficient matrix is calculated to describe the similarity among the features (Fig. 7). Redundant and repetitive features are then eliminated.

The information within the green rectangle represents the correlation coefficients of the average and extreme values of the lateral velocity. These two features are essentially of the same origin and have an extremely high degree of similarity (0.98). The blue rectangle shows the correlation coefficients of all longitudinal features. Peak and mean acceleration (0.96), peak and mean deceleration (0.98), and acceleration–deceleration ratio and peak acceleration (0.66) are highly correlated. This matches the logic that “stronger acceleration → higher ratio of acceleration → higher peak acceleration”. The correlation coefficient between acceleration and deceleration is low (0.27–0.28).

The description within the pink rectangle indicates the degree of correlation between the lateral and longitudinal features. The correlation between the two is extremely low (approximately 0), meaning

that the lateral and longitudinal features are completely independent of each other. This can serve as an important basis for selecting clustering features in the subsequent process. The orange rectangle represents the correlation coefficient of two relative characteristics (Average over the whole period and average of start–end difference), with a correlation of only 0.08. This can be used to describe the dynamic changes of “speed difference”. Based on the correlation coefficient analysis and the grid search algorithm, the final set of six features is presented in Appendix B, Table B.2.

3.3.2. Analysis of clustering results: between-cluster and within-cluster

Table 5 presents the number of sample points in each cluster, and Fig. 8 shows the three-cluster structure obtained by BGMM in the selected feature space.

From the between-cluster perspective, the number of clusters was automatically set to 3. The largest cluster accounts for 56.35%, while the smallest cluster accounts for 19.23%. There is no situation where a very small number of sample points form a separate cluster. The silhouette coefficient is 0.18. In the cluster analysis for traffic issues, a silhouette coefficient of 0.18 for the clustering is a moderate to relatively low result. The primary reason for this is that the vehicle’s driving trajectory patterns inherently exhibit certain similarities within a fixed area. From the perspective within the clusters, the average and median posterior probabilities of the sample points within each cluster were both above 98%, indicating that the sample points were accurately classified into their respective clusters.

Table 6 presents the statistical results of the dynamic indicators, and Fig. 9 shows the cluster-level radar plot based on the final six selected features. Cluster 0 occupies 19.22% of the total data volume. Its acceleration index is relatively low (−1.46), and it also has the lowest mean acceleration–deceleration ratio (−1.29). This indicates that within the merging area, Cluster 0 mainly moves at a constant speed or decelerates, tending to actively avoid and belonging to defensive driving. Cluster 1 accounts for 24.43% of the total data volume and has the highest acceleration (1.19). It also has the highest mean acceleration–deceleration ratio (1.49). This indicates that, within the merging area, acceleration frames occur more frequently than deceleration frames, suggesting an acceleration-oriented longitudinal adjustment pattern. Cluster 2 has the largest number of samples, accounting for 56.35%. The changes in acceleration indicators are relatively stable, and the mean acceleration–deceleration ratio is close to 0, indicating that the vehicle spends half of the time in the merging window accelerating and the other half decelerating. This is the most common stable driving behavior pattern. Fig. 9 visualizes the behavioral profiles of each cluster. It supports the previous quantitative explanation and shows that the three BGMM clusters have distinguishable behavior patterns across the final six features.

Table 3

Summary of preprocessing results for merging direction vehicles.

Metric	Raw data	Processed data	Change/Notes
Total frames	2,097,592	2,097,281	↓ 0.02%
Total/Valid trajectories (≥ 50 frames)	5453	5444	↓ 0.17%
Number of mainline vehicles	4022	4014	↓ 0.20%
Number of ramp vehicles	1431	1430	↓ 0.07%
Average trajectory length (frames per vehicle)	384.67	385.25	↑ 0.15%
Lane changes toward the ramp	–	726	Toward the ramp
Lane changes away from the ramp	–	1809	Away from the ramp

Table 4
Summary of merging event statistics.

Metric	Raw dataset total	Statistic	Percentage (%)
Mainline-ramp interaction pairs	–	3997	–
Valid merging ramp vehicles	1430	1422	99.4
Interacting mainline vehicles	4014	986	24.6
Average merging duration (s)	–	6.14	–
Average dynamic window duration (s)	–	6.70	–

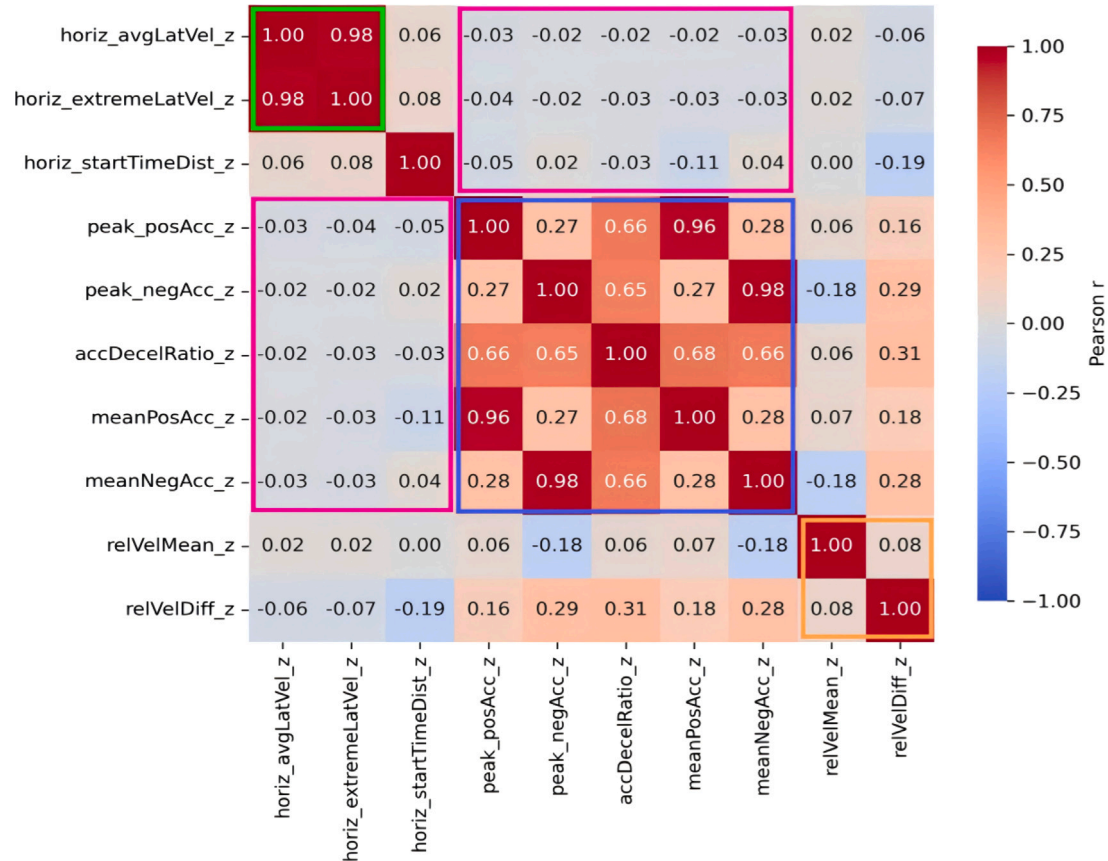


Fig. 7. Correlation matrix of standardized features.

Table 5
Sample distribution and posterior statistics.

Cluster ID	Sample count	Percentage (%)	Mean posterior prob.	Median posterior prob.
0	703	19.22	0.99	0.99
1	893	24.43	0.99	0.99
2	2060	56.35	0.98	0.99

Table 6
BGMM cluster behavioral feature means.

Cluster ID	Longitudinal acc. pattern (PC1)	Acc.–decel. balance	Lateral direction	Lateral adjustment	Response timing	Relative speed
0	–1.46	–1.29	0.08	0.62	–0.18	0.53
1	1.19	1.49	–0.05	–0.12	–0.12	0.46
2	–0.02	–0.21	–0.01	–0.16	0.11	–0.38

Note: All behavioral features are standardized using z-score normalization.

3.3.3. Clustering validation and behavioral interpretation

This part further validates and explains the clustering results obtained in Section 3.3.2 from a behavioral perspective. To ensure a fair comparison among different clustering methods, *k*-means, standard GMM, and BGMM use the same six behavioral features and the same data preprocessing procedure.

As shown in Table 7, *k*-means obtains the best geometric separation scores. This is consistent with its distance-based hard clustering objective. However, BGMM performs better than standard GMM in terms of silhouette, Davies–Bouldin, and Calinski–Harabasz scores. This indicates that, within the probabilistic mixture model framework, the Bayesian formulation can produce a more stable clustering structure.

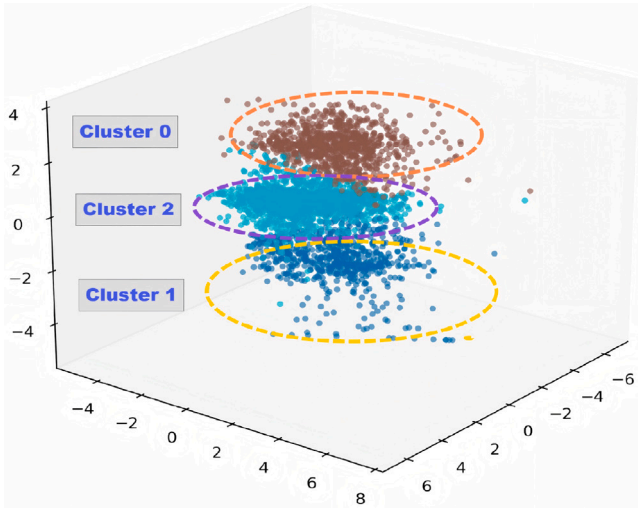


Fig. 8. Three-dimensional visualization of BGMM clustering results in the selected feature space.

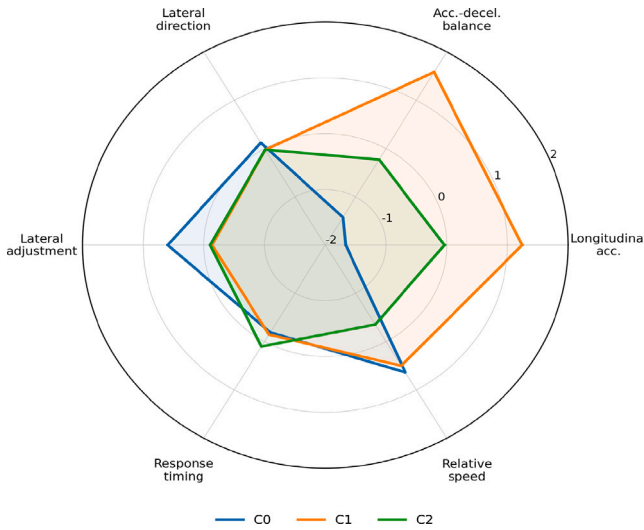


Fig. 9. BGMM cluster behavioral radar plot.

Table 7

Comparison of clustering methods under fixed $k = 3$.

Method	k	Silhouette	Davies–Bouldin	Calinski–Harabasz	Cluster size ratio
k -means	3	0.2298	1.6021	816.42	23.99/47.24/28.77%
GMM	3	0.1146	2.5195	420.39	15.21/39.06/45.73%
BGMM	3	0.1842	2.0338	563.44	15.78/22.95/61.27%

Therefore, although BGMM is not the best method for all geometric clustering metrics, it can still be used as a probabilistic clustering method that provides posterior membership, unbalanced cluster structures, and behavioral interpretability.

Furthermore, this study uses the k -means gap statistic, GMM BIC/AIC, and BGMM component weight analysis to check the selection of the cluster number. The max-gap rule of k -means selects $k = 8$, while the one-standard-error rule selects $k = 2$. The BIC/AIC results of standard GMM also prefer more components, but the corresponding silhouette scores are lower. This shows that mainline vehicle merging behavior does not have a single discrete cluster number that can be clearly determined by hard clustering or likelihood-based criteria. In contrast, when BGMM is initialized with 12 candidate components, only 3 components keep significant mixture weights. Therefore, this study

Table 8

Label similarity between clustering methods.

Method pair	ARI	NMI
k -means vs. GMM	0.307	0.331
k -means vs. BGMM	0.390	0.430
GMM vs. BGMM	0.548	0.649

keeps the three-cluster structure based on a combined consideration of BGMM component weights, behavioral interpretability, and the needs of the following HB-IRL/QR-DQN modeling. It does not claim that three clusters are the only mathematically optimal solution under all clustering criteria.

Finally, this study further compares the behavioral profiles of k -means, GMM, and BGMM, and calculates the ARI/NMI label similarity (Table 8). The results show that BGMM is more consistent with standard GMM than with k -means. This indicates that probabilistic clustering methods capture a structure that is different from hard distance-based clustering. After cluster alignment, k -means tends to split the BGMM common moderate interaction group into different clusters. In contrast, BGMM keeps one dominant common behavior pattern and two smaller but clearly different behavior patterns. This further supports the use of BGMM as the behavioral representation for the following HB-IRL reward learning and QR-DQN policy training.

3.4. HB-IRL: experimental setup, convergence diagnosis, and behavior reproduction

3.4.1. Training workflow and hyperparameter settings

The HB-IRL training is mainly divided into the following seven steps:

- Step 1. Data loading:** Load the feature behavior data of each cluster's mainline vehicles from the preprocessing results, mainly including feature tensors (5-dimensional discrete actions, 59-dimensional features), expert action indices (0/1), and action validity masks (0–4), etc. The transformation rules used to construct the 59-dimensional state–action feature vector are summarized in Appendix B, Table B.3. Store them uniformly on the GPU.
- Step 2. Build a hierarchical reward model:** Our study adopted a decentralized HB-IRL structure and constructed a global weight ($\mathbf{w}_0 \in \mathbb{R}^{59}$), the decentralized bias weight of each cluster ($\mathbf{z} \in \mathbb{R}^{3 \times 59}$), the lateral lane-changing weight ($\mathbf{w}_{lc} \in \mathbb{R}^{59}$), the lateral drift offset vector ($\Delta \in \mathbb{R}^{59}$), and the logarithmic standard deviation ($\log \sigma_b$), totaling 355-dimensional parameter vectors. $\mathbf{b}_k = \sigma_b \mathbf{z}_k$ is used to depict the bias weights of each cluster, and $\mathbf{w}_{lc} \pm \Delta$ is used to describe the lateral offset behavior of the mainline HDVs.
- Step 3. Empirical covariance and whitening initialization:** To improve the sampling efficiency of the algorithm in high-dimensional space, the ϕ corresponding to the actions in the feature tensor is downsampled to construct the empirical covariance matrix. By adding noise along the diagonal and performing Cholesky decomposition to construct a block diagonal matrix L_{ext} as the whitening matrix, the sampling is transformed into the whitened coordinate system for processing.
- Step 4. MCMC sampling and hyperparameter configuration:** In the 355-dimensional global space, a HMC sampler with PT is employed. Ten chains with different inverse temperatures are set up. The features are updated in block-wise and with low-frequency jumps, balancing both global exploration and local refinement. The total number of iterations, burn-in period, and other hyperparameters are summarized in Appendix B, Table B.4.

Table 9
Percentage of dimensions meeting MCMC diagnostic criteria.

Evaluation category	Metric	Satisfied (%)
Lenient criteria	max R-hat < 1.05	98.87%
	min ESS_bulk $\geq$ 50	94.93%
	max R-hat < 1.05 & min ESS_bulk $\geq$ 50	94.93%
	max R-hat < 1.01	85.63%
Strict criteria	min ESS_bulk $\geq$ 100	85.92%
	max R-hat < 1.01 & min ESS_bulk $\geq$ 100	85.63%

5. **Step 5. Adaptive strategy and inter-chain exchange:** We apply scale adaptation for the proposed covariance and dual-averaging adaptation for the cold chain. To improve the mixture performance of the posterior distribution in the multi-modal case, Metropolis exchanges are performed between adjacent temperature chains.
6. **Step 6. Evaluation of convergence and stability:** Calculate the effective sample size for each dimension and the Gelman–Rubin diagnostic quantity to assess the convergence and stability of the training process.
7. **Step 7. Weight export:** Calculate the mean and standard deviation of the posterior samples, and derive the various weight banks explained in Step 2, which are used as parameters of the linear reward function for the subsequent one-step greedy baseline and the RL agent.

The sampling convergence is evaluated by using the MCMC diagnostics, through the assessment of two indicators: maximum R-hat (Gelman–Rubin statistic) and minimum Bulk Effective Sample Size (ESS_bulk). And the accuracy of model training and prediction is evaluated through the Top-3 accuracy rate and Negative Log-Likelihood (NLL).

3.4.2. MCMC diagnostics for sampling convergence

Due to the complex and diverse nature of HDV behavior, it is usually difficult to achieve global convergence in high dimensions. In our study, among the 355 dimensions of the entire chain's features, 85.63% of the features have an R-hat value less than 1.01, and 85.92% of the features have an effective sample size of at least 100 (Table 9). Regarding the validity of “local convergence”, it has been proven to be correct, effective, and applicable to subsequent studies in some studies (Moins et al., 2025; Vehtari et al., 2021; Vats and Knudson, 2021).

Fig. 10 presents representative marginal density distributions for three selected dimensions (Fig. 10(a)), the rank plot (Fig. 10(b)), the autocorrelation function on a logarithmic scale (log-ACF) (Fig. 10(c)), and the convergence and effective sample size of 355 dimensions (Fig. 10(d)).

In Fig. 10(a), the posterior distribution curves of the three random dimensions overlap closely. Within the 95% interval, a unimodal and symmetric distribution is observed, with no significant long-tail phenomenon at both ends, and the mean is concentrated around 0. This indicates that these three dimensions have been fully explored, and the marginal distribution of the reward remains stable under different feature dimensions. The rank plot (Fig. 10(b)) shows the degree of between-chain mixing. The samples from all chains are evenly distributed across the ranks, with similar heights and no obvious peak phenomenon, indicating that the mixture between the chains is good and they can freely pass between different chains.

Fig. 10(c) shows the logarithm of the autocorrelation function when the total number of dimensions is 10, 120, and 240. It represents the attenuation of the autocorrelation of the sampling trajectory with respect to the lag. Generally, if the autocorrelation coefficient is below 0.1, it can be considered that the dimensions are approximately independent. In Fig. 10(c), all the curves show a tendency to decrease to 0.1. Among them, dim 120 can drop below 0.1 at lag $\approx$ 20, with short

autocorrelation and independent samples. The other two cases exhibit a slower rate of decline, yet still display a downward trend, gradually approaching the threshold of 0.1. Fig. 10(d) shows the convergence (R-hat) and effective sample size ($\log_{10}(\text{ESS})$) for all 355 dimensions. Most dimensions converge, and R-hat is close to 1.01. Moreover, the number of effective samples for most dimensions is greater than 100 ($\log_{10} \text{ESS} > 2$), and most dimensions are concentrated within the range of [2, 5], with ESS ranging from 100 to 30,000. The number of effective samples is enough to support a subsequent study.

For the locally unconverged dimensions, their posterior distributions were kept when constructing the reward function, instead of being discarded. A targeted sensitivity analysis was further conducted to test their potential influence. In this analysis, only the locally unconverged dimensions were perturbed, and the resulting QR-DQN policies were evaluated.

3.4.3. Downstream sensitivity analysis of locally unconverged HB-IRL dimensions

To further examine whether the locally unconverged HB-IRL dimensions affect the downstream policy conclusions, this study adds a sensitivity analysis on the uncertainty of HB-IRL posterior weights. Under the strict criterion of R-hat < 1.01, 304 out of 355 HB-IRL parameter dimensions meet the convergence criterion, accounting for 85.63%. The remaining 51 dimensions are marked as locally unconverged dimensions. We further check the cluster, action, and feature of these dimensions, and analyze their posterior variance, effective sample size, and reward contribution. The results show that some locally unconverged dimensions still have non-zero reward contributions. Therefore, they cannot be simply ignored only based on the convergence diagnostic results.

Therefore, this study constructs targeted perturbation settings to examine the influence of these locally unconverged dimensions on the downstream policy. Specifically, all dimensions that meet the convergence criterion are kept fixed. Only the locally unconverged dimensions are perturbed based on the posterior distribution, including posterior quantiles and mean $\pm$ one standard deviation. In total, 9 settings are tested, including the posterior-mean baseline, all-unconverged q_{90}/q_{10} , top-10 high-contribution perturbation, PCA0-block perturbation, w_0 -only perturbation, and lateral-only perturbation.

In the same-state reward/action sensitivity analysis, some perturbations do change the reward-induced action preference, especially top10_q90 and pca0_block_q90. This shows that the locally unconverged dimensions are not completely negligible at the reward/action level. Therefore, it is necessary to further examine whether this reward/action-level sensitivity is passed to the downstream QR-DQN policy performance.

In the downstream QR-DQN sensitivity analysis, each perturbation setting uses the corresponding HB-IRL reward weights to fine-tune the QR-DQN policy for 20,000 steps. Then, the policy is tested in closed-loop simulations under the same SUMO scenario and evaluation protocol. This analysis covers all three behavior clusters. Table B.5 in Appendix B reports the maximum absolute changes of all perturbation settings compared with the posterior-mean baseline. The results show that, for all three clusters, the maximum changes in collision-free rate, diagnostic pass ratio, travel time, average speed, lane-change count, drift count, and mean absolute jerk are all 0. Therefore, although the locally unconverged dimensions affect the reward/action-level preference, they do not lead to observable degradation in the downstream closed-loop QR-DQN policy performance within the tested perturbation range.

3.4.4. Behavior reproduction performance evaluation

Fig. 11 presents the top-3 accuracy rate and the reliability diagram. The top-3 accuracy rate is used to evaluate the quality of the model's reproduction of expert actions, while the reliability curve is used to assess the credibility of the model's probability outputs.

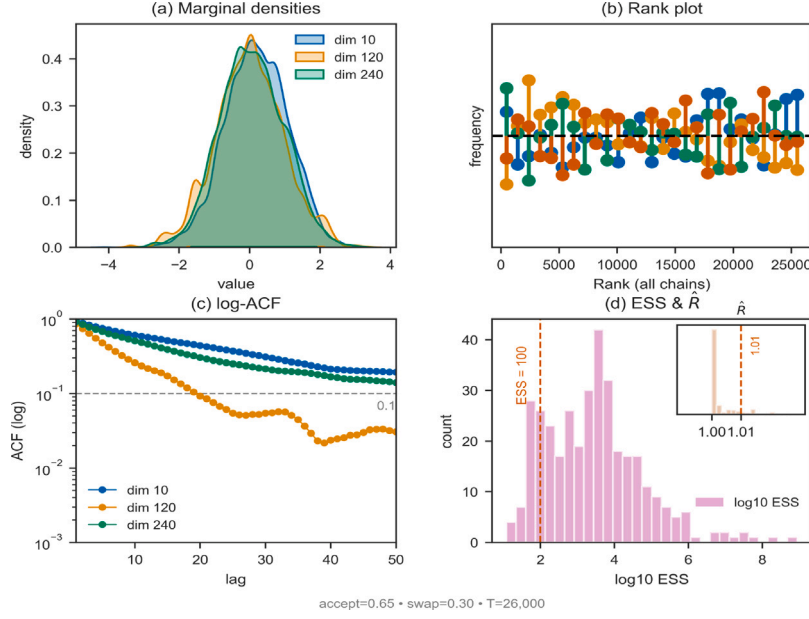


Fig. 10. Marginal, rank, ACF, and convergence statistics.

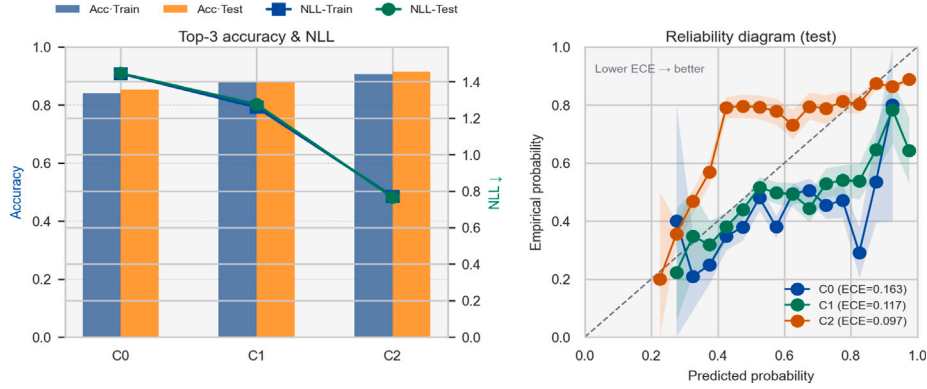


Fig. 11. Prediction accuracy, NLL, and calibration by cluster.

The training and testing accuracy rates of the top-3 features for all clusters in the model are significantly higher than the random value (20%), and are all above 0.8. Among them, the top-3 training and testing accuracy rates of Cluster 0 are around 80%–85%, indicating that there are still some differences among the sample points within the cluster. Clusters 1 and 2 have higher top-3 accuracy than Cluster 0, reaching about 90%. This shows that the learned policies capture the main top-3 action preferences in these clusters. The differences between the training and test accuracy of the three clusters are minimal, indicating a good generalization effect. In the reliability diagram, Cluster 2 is closest to the diagonal line and has the lowest Expected Calibration Error ($ECE = 0.097$), indicating that this cluster is the easiest to predict. Cluster 0 is the most difficult to predict ($ECE = 0.163$), which is consistent with the results of the top-3 accuracy plot.

3.5. RL-based behavioral reconstruction

3.5.1. Training workflow and hyperparameter configuration

Based on the weight information obtained during the HB-IRL stage, RL training is conducted in the SUMO environment using QR-DQN. The process mainly consists of the following 7 steps:

- Step 1. Scenario and environment initialization:** In SUMO, a merging area consisting of two mainlines and a single ramp road was set up, and it was encapsulated as a Gym-style environment. The traffic participants include mainline RL agents (red), mainline background vehicles (blue), and the background vehicles (gray) on the ramp (Fig. 12). Set 7-dimensional discrete actions and 16-dimensional observation vectors for the RL agent (Appendix B, Tables B.6 and B.7). The simulation step size is 0.2 s.
- Step 2. HB-IRL weight sampling and reward construction:** We load the weight information obtained in the HB-IRL stage, construct the cluster-specific weight matrix $\mathbf{W} \in \mathbb{R}^{5 \times 59}$, and simulate the corresponding driving style of that cluster. Using that weight matrix $\mathbf{W}$ and the 59-dimensional features, a linear reward function is constructed.
- Step 3. Reward mechanism:** To ensure normal interaction and safety among vehicles, a reward and punishment mechanism is designed, including the time cost for each step, the reward for successful merging, and the penalty for collisions.
- Step 4. Randomization of traffic flow:** The main traffic flow is randomly sampled within the given range, and a routing file is automatically generated for each episode. And short-duration

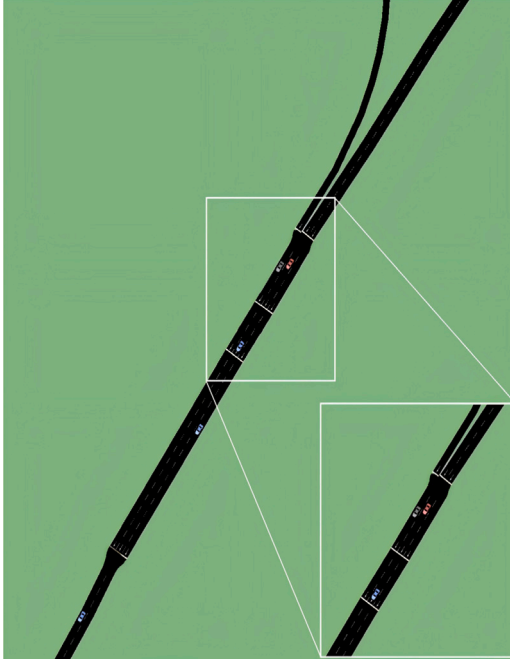


Fig. 12. The SUMO simulation environment overview.

pulse flows are injected to increase the conflict scenarios between the main lanes. The specific routing configuration can be found in [Appendix B](#), [Tables B.8](#) and [B.9](#).

5. **Step 5. Parallel environment and sampling mechanism:** We employed eight parallel environments using a subprocess-based vectorized environment (SubprocVecEnv), and set independent random seeds and weight noise seeds for each environmental process to ensure that weight sampling, route generation, and vehicle behavior are mutually independent.
6. **Step 6. QR-DQN training configuration:** We adopt sb3_contrib.QRDQN as the RL algorithm, and the specific hyperparameter configuration is shown in [Table 10](#).
7. **Step 7. Evaluation mechanism:** We designed the InfoTensorboardCallback function to extract statistical indicators from the environment.

3.5.2. Convergence analysis and cluster-wise speed comparison

Overall, Clusters 0, 1, and 2 all showed a convergent trend after 1 million training iterations ([Appendix B](#), [Fig. B.14](#)). [Fig. 13](#) presents heatmaps of the mean longitudinal speed of 500 episodes for each cluster at the end of the RL training. The mainline RL agent's grid-based data is shown, where different colored grids represent the mean longitudinal speed of vehicles within the corresponding spatial region.

In the early stage of the RL agent's movement in Cluster 0, the vehicle has a relatively high longitudinal speed. When approaching the merging area, it chooses to slow down and give way to provide merging space for the on-ramp vehicles. The traffic flow efficiency is low. The speed of Cluster 1 was significantly higher than that of the other two clusters, and the speed fluctuation was minimal. As a result, it quickly passed through the merging area and exhibited an acceleration-oriented longitudinal adjustment pattern. The speed of Cluster 2 is generally in between that of Cluster 0 and Cluster 1. During the vehicle's operation, it maintains a constant speed cruise, favoring a more comfortable and smooth driving strategy.

3.5.3. Behavioral validation of cluster-specific driving policies

To avoid the randomness of using only one representative trajectory, this study uses multi-episode statistical validation. It compares

Table 10

Main hyperparameters of the QR-DQN training and the SUMO environment.

Category	Hyperparameter	Value
Simulation	Simulation step length	0.2 s
	Max steps per episode	400
Control	Speed increment per action Δv	1.0 m/s
	Discrete action space size	7 actions
Scene randomization	Mainline flow range	[2500, 4000] veh/h
	Ramp flow range	[600, 1200] veh/h
Behavior & reward	HB-IRL behavior cluster	0/1/2
	Weight noise scale	0.5
	Reward clipping range	[-10, 10] (tanh)
	Reward scaling factor	2.0
Training (QR-DQN)	Number of parallel environments	8
	Total training steps	1,000,000
	Discount factor γ	0.985
	Network architecture	[256, 256] MLP
	Learning rate	1×10^{-4}
	Replay buffer size	500,000
	Batch size	256
	Target update interval	20,000 steps
	Exploration schedule ($\epsilon_{\text{start}}, \epsilon_{\text{final}}, f_{\text{expl}}$)	(0.6, 0.05, 0.7)

Note: ϵ_{start} and ϵ_{final} denote the initial and final exploration rate, respectively, and f_{expl} is the exploration fraction.

observed trajectories, greedy baseline trajectories, and RL-generated trajectories at the cluster level. The model calibration does not directly fit a fixed average speed. Instead, it follows the process of real driving behavior clustering – cluster-specific reward learning – cluster-specific policy training. First, BGMM identifies behavior clusters of real mainline vehicles during merging interactions. Then, HB-IRL learns reward weights from expert state-action samples in each cluster. Finally, QR-DQN trains the corresponding policy in SUMO using the cluster-specific reward. Therefore, the validation focuses on whether the RL policies can reproduce the interaction behavior tendencies of different clusters, rather than only matching the average speed.

To avoid the randomness of a single simulation result, this study uses 20 valid late-training episodes for each cluster for statistical validation. For each metric, we report the mean, standard deviation, standard error, and 95% confidence interval. We also use one-way ANOVA and the Kruskal-Wallis test to examine whether the three cluster-specific RL policies show statistically significant differences in behavior metrics. The ANOVA effect size η^2 is also reported. Based on this, this study uses average speed, acceleration, jerk, lateral offset, and statistical confidence intervals together for validation.

As shown in [Table 11](#), the RL policies of the three clusters can reproduce the main longitudinal behavior trends in the real data. For Cluster 0, the RL policy shows clear negative acceleration, which is consistent with the deceleration trend in the real data. Therefore, it can be interpreted as a conservative yielding pattern. This is also consistent with the interpretation of Cluster 0 in the BGMM clustering stage and the HB-IRL reward learning stage. It should be noted that real vehicles may enter the merging window at a high speed and then gradually decelerate. Therefore, their average speed can still remain high. In contrast, the average speed of Cluster 0 in RL is lower than that in the real data. This means that the policy learns the deceleration tendency, but it is more conservative in absolute speed. Cluster 1 shows positive acceleration and has the closest average speed to the real data, so it has the best longitudinal match. For Cluster 2, both the real and RL accelerations are close to 0, indicating a more stable speed-maintaining pattern. No hard braking or low-speed collapse is observed in the evaluated episodes, which shows that the learned policies run stably under the current safety constraints.

As shown in [Table 12](#), the RL policies can partly reproduce the lateral offset level in the real data, but their lateral fluctuation is larger overall. For Cluster 0, the mean absolute lateral offset is close to the real data, showing a better match in average lateral offset magnitude. In contrast, the RL mean absolute lateral offset values of Cluster 1 and

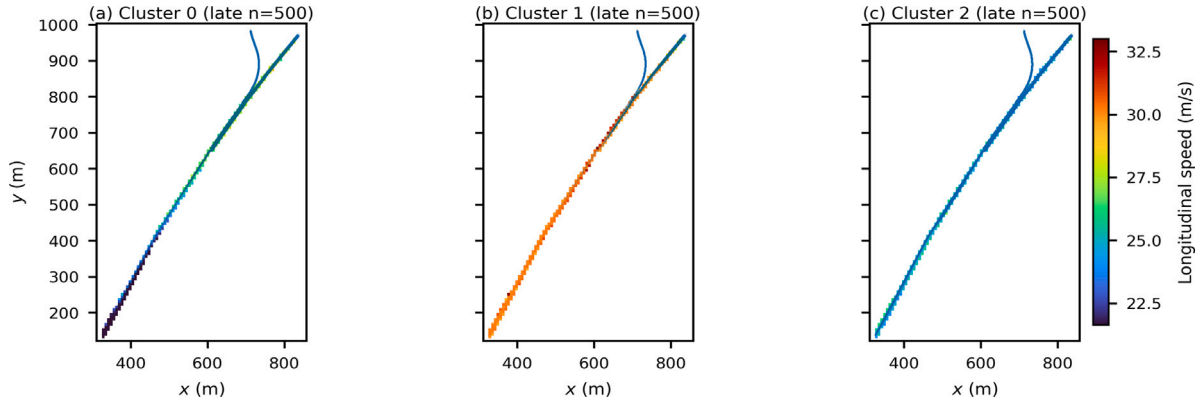


Fig. 13. Longitudinal speed heatmaps of late-stage RL trajectories (last 500 episodes per cluster).

Table 11

Multi-episode validation of longitudinal behavior among real data, greedy baseline, and RL policies.

Cluster	Behavior interpretation	Real speed [m/s]	Greedy baseline speed [m/s]	RL speed [m/s]	Real acc. [m/s ²]	RL acc. [m/s ²]
0	Deceleration-oriented/conservative yielding	32.10 (6.42)	30.40 (3.29)	25.04 (1.68) [24.25, 25.83]	−0.191 (0.285)	−0.304 (0.034) [−0.320, −0.289]
1	Acceleration-oriented/assertive	32.88 (5.56)	34.95 (3.57)	32.31 (1.18) [31.75, 32.86]	0.174 (0.156)	0.093 (0.054) [0.068, 0.119]
2	Speed-maintaining/stable	28.68 (5.92)	24.67 (2.36)	26.32 (0.39) [26.14, 26.50]	0.006 (0.068)	−0.012 (0.022) [−0.022, −0.001]

Note: Values in parentheses are standard deviations, and values in square brackets are 95% confidence intervals of RL episode-level means. Real data and the greedy baseline are based on sample-level statistics, while RL policies are based on 20 episode-level means for each cluster.

Table 12

Multi-episode validation of lateral behavior between real data and RL policies.

Cluster	Behavior interpretation	Real mean abs. offset [m]	Real lateral offset std. [m]	RL mean abs. offset [m]	RL lateral offset std. [m]	RL/Real std. ratio
0	Deceleration-oriented/conservative yielding	0.322	0.256	0.362 [0.317, 0.407]	0.425 [0.376, 0.473]	1.66
1	Acceleration-oriented/assertive	0.357	0.336	0.456 [0.436, 0.477]	0.492 [0.467, 0.517]	1.47
2	Speed-maintaining/stable	0.323	0.262	0.452 [0.392, 0.513]	0.518 [0.450, 0.587]	1.98

Note: Values in square brackets are 95% confidence intervals from 20 RL evaluation episodes. The RL/Real std. ratio is the ratio between the RL and real lateral offset standard deviations.

Cluster 2 are higher than those of the real data. At the same time, the RL lateral offset standard deviation is larger than the observed trajectories for all three clusters, with an RL/Real std. ratio of 1.47–1.98. This indicates that the current RL policies can keep some cluster-wise lateral behavior differences, but the stability of lateral motion still needs improvement.

Although the RL policies can reproduce the longitudinal behavior trends of different clusters reasonably well, their lateral offset standard deviation is still much higher than that of the observed trajectories. Based on the current implementation, this difference is related to two factors. First, the lateral offset definitions are not exactly the same. The real data use the lane-center offset based on the real lane centerline in exiD, while the RL trajectories use the lane centerline projection from the SUMO net.xml, followed by temporal smoothing and vehicle-wise centering. Therefore, this metric may be affected by the road network representation, the nearest-lane projection method, and the post-processing method. Second, the current state and reward design constrain longitudinal interaction more strongly. Lateral motion is mainly represented by limited lateral offset, lane area, and lateral

actions. Therefore, the current model can reproduce longitudinal acceleration and deceleration trends more easily, but lateral fluctuations remain relatively large. This issue is discussed as a limitation of the current model. In future work, we will improve lateral feature design, lateral reward shaping, and the lateral control mechanism.

The statistical test results in Table 13 further show that the three cluster-specific RL policies have significant differences in both longitudinal and lateral behavior. Specifically, the ANOVA p-values for average speed, mean acceleration, and mean absolute jerk are all below 0.001, with effect sizes η^2 of 0.878, 0.951, and 0.806, respectively. This indicates clear differences among the cluster policies in speed level, longitudinal acceleration/deceleration trend, and smoothness. Lateral offset variability also shows a statistically significant difference, but its effect size is smaller, with ANOVA $p = 0.024$ and $\eta^2 = 0.122$. These results show that the three cluster-specific policies do not produce the same behavior. Instead, they learn statistically distinguishable driving behavior patterns.

Table 13
Statistical tests for differences among cluster-specific RL policies.

Metric	ANOVA p-value	Effect size η^2	Kruskal–Wallis p-value	Interpretation
Average speed	< 0.001	0.878	< 0.001	RL clusters differ significantly in speed level
Mean acceleration	< 0.001	0.951	< 0.001	RL clusters differ significantly in longitudinal tendency
Mean absolute jerk	< 0.001	0.806	< 0.001	RL clusters differ significantly in comfort/smoothness
Lateral offset std.	0.024	0.122	0.0015	RL clusters differ in lateral variability
Mean abs. lateral offset	0.0039	0.177	< 0.001	RL clusters differ in lateral offset magnitude
Mean abs. lateral speed	0.0106	0.147	0.0146	RL clusters differ in lateral motion intensity

Table 14
Comparison of the exiD field implementation and the Hungarian M1 transfer case.

Aspect	exiD field implementation	M1 transfer case	Transfer treatment
Data source	Naturalistic trajectories	SUMO trajectories	Data source changed
Road environment	exiD merging site	Hungarian M1 highway	Geometry reconfigured
Merging interaction	Dynamic merging window	Same detection logic	Core logic reused
Behavior representation	Motion and interaction features	Same feature definitions	Definitions retained
Behavior model	BGMM/HB-IRL learned from exiD	exiD-derived behavior information retrained	No M1 retraining
QR-DQN policy	Trained on exiD	Deployed in M1	Policy transferred
State interface	exiD observation structure	16-dimensional M1 input	State adapter applied
Evaluation role	Field-based behavior learning	Closed-loop transfer test	Cross-scenario portability

Table 15
Experimental setup for the Hungarian M1 closed-loop transfer demonstration..

Item	Setting
Training dataset	exiD
Transfer scenario	M1 SUMO highway merging scenario
Number of merge zones	3
Background traffic	EIDM
Retraining on M1	No
Controlled vehicle	Rear mainline vehicle
QR-DQN observation	16-dimensional state vector
Decision interval	0.2 s
Safety layer	Rule-based safety shield
Baseline	EIDM-only
Matched comparison	EIDM $t_{\text{merge}} \pm 5$ s vs. QR-DQN first-control 0–5 s

3.5.4. Hungarian M1 SUMO transfer demonstration

To examine cross-scenario portability beyond the exiD field implementation, this study uses a high-fidelity SUMO merging scenario representing the Hungarian M1 highway. This scenario includes three merging areas and uses background traffic generated by the Extended Intelligent Driver Model (EIDM). The BGMM, HB-IRL, and QR-DQN models are trained only on the exiD dataset. No retraining or fine-tuning is conducted in the M1 scenario.

Compared with the exiD field site, the M1 scenario has different road geometry, merging areas, and traffic conditions. Therefore, the transfer test reconfigures site-specific settings while keeping the core analysis and policy structure unchanged. The merging-event identification logic, behavioral feature definitions, and exiD-trained QR-DQN policies are retained. The M1 road geometry, traffic demand, and vehicle-state interface are adjusted for the target scenario. In particular, the M1-to-QR-DQN state adapter converts M1 vehicle states into the 16-dimensional input required by QR-DQN. Table 14 summarizes the main mapping between the exiD field implementation and the M1 transfer case. The detailed experimental settings are then given in Table 15.

Specifically, we first run the EIDM-only baseline and record vehicle-level trajectory data. Merging events and interacting mainline vehicles

are then identified using the same detection logic. Finally, the three exiD-trained QR-DQN cluster policies are executed separately in the M1 closed-loop scenario and compared with the EIDM-only baseline.

All three exiD-trained QR-DQN cluster policies successfully completed the M1 closed-loop transfer runs, without simulation failure or collisions. This shows that the policies learned from exiD can be deployed in the Hungarian M1 merging test scenario without retraining, while maintaining basic closed-loop stability.

Table 16 summarizes the short-window safety metrics and cluster policy responses in the M1 transfer experiment. To avoid an unfair comparison between all EIDM network vehicles and QR-DQN controlled vehicles, this study builds a matched EIDM rear-mainline baseline. The EIDM baseline uses the $t_{\text{merge}} \pm 5$ s window, while QR-DQN uses the 0–5 s window after first control. The results show that the transferred QR-DQN policies do not increase the low-speed ratio or hard-braking vehicle ratio. Severe braking also rarely occurs. This indicates that the exiD-trained QR-DQN policies do not show clear short-term safety degradation in the Hungarian M1 merging scenario.

At the same time, the raw policy outputs and post-control responses still show differences among clusters. Cluster 0 more often produces negative target-speed changes and shows a clearer short-window speed reduction. This indicates a more conservative or yielding behavior. Cluster 2 shows weaker longitudinal deceleration intention. Cluster 1 shows a mixed adjustment pattern between the other two clusters.

Overall, the M1 experiment demonstrates that the proposed workflow can be reconfigured for different road geometry and traffic conditions while retaining the exiD-trained behavior policies. The results support cross-scenario portability and closed-loop executability, but they do not provide external validation using a second independent field-trajectory dataset. Direct AV control and mixed-traffic system performance are not evaluated in this transfer experiment.

4. Discussion on applicability and limitations

From an interdisciplinary perspective, this study does not use AI models as stand-alone tools for traffic data. Transportation engineering

Table 16
Matched M1 transfer safety and cluster-specific policy responses.

Method	Window	Speed < 18 m/s	Hard braking	Raw Δv mean	Raw $\Delta v < 0$	Dominant lateral intention	Δ speed 3 s
EIDM matched rear	$t_{\text{merge}} \pm 5$ s	9.88%	0.94%	—	—	—	—
QR-DQN C0	first-control 0–5 s	9.08%	0.64%	−0.021	30.9%	driftToward	−0.659
QR-DQN C1	first-control 0–5 s	8.53%	0.51%	+0.103	6.1%	driftAway	−0.572
QR-DQN C2	first-control 0–5 s	8.97%	0.58%	+0.119	0.5%	driftAway	−0.572

defines the merging scenario, vehicle interactions, and evaluation measures. Human driving behavior analysis is used to interpret behavioral heterogeneity and context-dependent responses. Data-driven methods identify behavior patterns and infer relative reward preferences, while closed-loop simulation evaluates their feasibility in a dynamic traffic environment. The framework also follows a human-centered transportation view by treating drivers as heterogeneous traffic participants with different interaction responses rather than as a homogeneous group.

This study still has several limitations. First, it mainly models mainline driver behavior during ramp merging based on vehicle trajectory data. Since eye-tracking data, questionnaires, physiological signals, and driver self-reports are not used, this study cannot directly infer drivers' psychological states or subjective intentions. Therefore, the behavior patterns and relative reward preferences in this study should be understood as behavior representations learned from observable kinematic features.

Second, the field-based evidence is still mainly obtained from the exiD naturalistic trajectory data. The M1 SUMO experiment extends the framework to different road geometry and traffic conditions, but it is a simulation-based transfer demonstration. Different road structures, traffic densities, and driving cultures may lead to different driving behaviors. Therefore, the results still need further validation on more real-world datasets and different merging scenarios.

Third, although the proposed method can reproduce the longitudinal behavior trends of different clusters reasonably well, the lateral offset standard deviation of the RL-generated trajectories is still higher than that of the observed trajectories. This shows that lateral behavior modeling and reproduction still need improvement.

Fourth, some cluster-specific policies show certain biases. For example, Cluster 0 has a relatively conservative speed profile. Although this is partly consistent with its deceleration-oriented or yielding behavior, future work should better balance safety, behavior similarity, and traffic efficiency.

Finally, the M1 transfer experiment provides a closed-loop transfer demonstration in a high-fidelity simulation scenario with different road geometry and traffic conditions. It supports the cross-scenario portability of the framework, but it does not provide external validation based on a second independent field-trajectory dataset. Future work should further evaluate the framework using real-world datasets from different road environments and locations. The present study also does not directly evaluate AV control algorithms or mixed-traffic system performance. These topics should be examined in future work.

5. Conclusion and future work

This study proposes a configurable empirical framework for analyzing heterogeneous human driving behavior during highway on-ramp merging. The framework connects traffic interaction analysis, behavior heterogeneity identification, relative preference inference, and closed-loop traffic evaluation. BGMM, HB-IRL, and QR-DQN are used to support these different stages. Based on naturalistic exiD trajectory data, it models heterogeneous mainline vehicle behavior during merging interactions. With this framework, this study identifies differences in speed adjustment, acceleration change, and lateral response among mainline vehicles. It also infers cluster-specific relative reward preferences and learns closed-loop behavior policies.

In the clustering stage, this study uses BGMM to obtain three interpretable mainline HDV behavior patterns. It also compares BGMM with k-means and standard GMM. The results show that ramp-merging behavior is continuous and overlapping, so the moderate silhouette coefficient is relatively low. This reflects that the boundaries of real driving behavior are not fully clear. Compared with hard clustering methods, BGMM can provide soft membership information, so it is more suitable for describing driving behavior patterns with unclear boundaries and high sample uncertainty.

In the HB-IRL stage, this study learns cluster-specific reward weights from observed driving behaviors in different clusters. Because the reward feature dimension is high and the driving action distribution is unbalanced, some parameters cannot fully meet the strict global convergence criterion. Therefore, this study further conducts a sensitivity analysis on the dimensions that do not fully meet the convergence criterion. The results show that perturbing these dimensions does not lead to clear degradation in downstream QR-DQN policy performance. This indicates that the main policy conclusions are relatively robust to the remaining uncertainty in the HB-IRL posterior.

In the closed-loop evaluation stage, the cluster-specific QR-DQN policies can reasonably reproduce the longitudinal behavior trends of different clusters, especially in speed adjustment and acceleration changes. However, the lateral offset standard deviation of the RL-generated trajectories is still higher than that of the observed trajectories. This shows that the current model still has limitations in lateral behavior modeling and reproduction. In addition, the Cluster 0 policy shows a relatively conservative speed profile. Although this is partly consistent with its deceleration-oriented or yielding behavior pattern, future work should further balance behavior similarity, safety, and traffic efficiency.

To further examine the cross-scenario portability of the framework, this study conducts a transfer experiment in a high-fidelity SUMO scenario representing the Hungarian M1 highway. The results show that the QR-DQN policies trained on exiD can run in a new closed-loop simulation environment while maintaining basic safety and stability. These results support the cross-scenario portability of the proposed framework. However, since the M1 scenario is still based on simulation data, future work should further validate the model's generalization ability on more independent real-world datasets and different road structures.

Overall, this study presents a configurable empirical framework that connects transportation engineering, human driving behavior analysis, data-driven inference, and closed-loop traffic evaluation for highway on-ramp merging. Future work will extend the framework to more scenarios, multi-lane conditions, and multiple datasets. It will also improve lateral behavior modeling, nonlinear reward functions, multi-agent interaction learning, and interaction decision modeling in mixed HDV-AV traffic.

CRedit authorship contribution statement

Xinze Zhang: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **István Varga:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Tamás Tettamanti:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Table A.1
Definition of key merging time points.

Time point	Judging rule
Merging point (t_{merge})	If <code>lanechange</code> = 1 and the <code>laneletId</code> changes from ramp to mainline in two consecutive frames.
Merging start frame (t_{start})	Search backward from t_{merge} and find the first frame where the lateral offset (<code>latOffset</code>) exceeds the threshold.
Merging end frame (t_{end})	Search forward from t_{merge} until <code>latOffset</code> < 0.02 m/frame and <code>latAcc</code> < 0.25 m/s ² .

Table A.2
Surrounding directions of the current ramp vehicle.

Type	LeadId	RearId	LeftLeadId	LeftRearId
Type	LeftAlongsideId	RightLeadId	RightRearId	RightAlongsideId

Table A.3
Dataset preprocessing operations and their purposes.

Operation	Processing method	Purpose
Dataset merging	Merge <code>tracks.csv</code> , <code>tracksMeta.csv</code> , and <code>recordingMeta.csv</code> Based on <code>laneletId</code> field	Enable unified field for further analysis
Highway merging/diverging segmentation		Focus on highway merging areas
Default value handling	Replace default values with NaN, -1, or -1000	Improve recognition in computation
Missing value handling	Remove or impute NaN values	Improve data integrity
Data type conversion	Convert multi-value fields into lists	Support BGMM and HB-IRL processing
Time alignment	Use "frame/frameRate (25 Hz)"	Generate a global timeline for synchronization
UTM coordinate calculation	<code>utm_x = xCenter + xUtmOrigin</code> , <code>utm_y = yCenter + yUtmOrigin</code>	Enable spatial visualization
Heading angle normalization	<code>heading_rad = heading × π/180</code>	For direction-related vector operations
Lateral offset normalization	<code>latOffset_norm = latLaneCenterOffset/laneWidth</code>	Map offset to [0, 1] range
Longitudinal position normalization	<code>longitudinal_pos_norm = lonLaneletPos/laneletLength</code>	Normalize travel distance per lanelet to [0, 1]
Valid trajectory selection	<code>numFrames ≥ 50</code>	Ensure full decision-to-execution coverage
Add <code>laneChangeDir</code> label	0 = Lane change toward ramp 1 = Keep lane 2 = Lane change away from the ramp	For downstream clustering and modeling

Code availability

Example code snippets for reproducing the main analyses are available at: <https://github.com/ZhangXinzhe9/exid-ramp-merge-analysis>. These scripts are simplified and illustrate the key steps of the methodology.

Declaration of generative AI and AI-assisted technologies in the manuscript preparation process

During the preparation of this manuscript, the authors used AI-assisted tools for spelling and language correction only. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

Funding

The research was supported by the European Union within the framework of the National Laboratory for Autonomous Systems (RRF-2.3.1-21-2022-00002).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Dataset description and preprocessing details

See [Tables A.1–A.3](#).

Appendix B. Feature engineering, model settings, training diagnostics, and simulation configuration

See [Tables B.1–B.9](#).

Appendix C. Glossary of terms and abbreviations

See [Table C.1](#).

Table B.1

Initial candidate features and preprocessing methods for BGMM clustering.

Feature group	Feature (variable name)	Type	Processing method
Lateral features	Lane-change direction (laneChangeDir)	Categorical	One-hot encoding
	Lateral speed direction (horiz_dir)	Direction	Derived from sign of lateral acceleration
	Average lateral speed (horiz_avgLatVel)	Continuous	IQR trimming + z-score normalization
	Extreme lateral speed (horiz_extremeLatVel)		
Longitudinal features	Frames from start to merge point (horiz_startTimeDist)	Continuous	IQR trimming + z-score normalization
	Peak acceleration (peak_posAcc)		
	Peak deceleration (peak_negAcc)		
	Acceleration ratio – deceleration ratio (accDecelRatio)		
	Mean acceleration (meanPosAcc)		
Mainline–ramp interaction	Mean deceleration (meanNegAcc)		
	Relative speed (overall mean) (relVelMean)	Continuous	IQR trimming + z-score normalization
	Relative speed (start–end diff mean) (relVelDiff)		

(Note: *IQR* denotes interquartile range.)**Table B.2**

Final six-dimensional behavior-oriented features for clustering validation.

Final feature	Source/transformation	Behavioral meaning
vert_pc1_white	First PCA component of peak_posAcc_z, peak_negAcc_z, meanPosAcc_z, and meanNegAcc_z	Longitudinal acceleration pattern
accDecelRatio_z	z-score normalized accDecelRatio	Acceleration–deceleration balance
horiz_dir_z	Scaled horiz_avgLatVel_z	Lateral movement direction
horiz_strength_z	Standardized absolute horiz_avgLatVel_z	Lateral adjustment intensity
horiz_tLead_z	z-score normalized horiz_startTimeDist	Lateral response timing
relVelMean_z	Scaled relVelMean_z	Relative speed to ramp vehicle

Table B.3

State–action feature transformation for HB-IRL reward learning.

Feature	Processing method
Longitudinal speed	z-score normalization
laneRegion	Two-bin one-hot encoding
ddBin_0 /ddBin_1	Two-bin one-hot encoding (longitudinal distance diff)
histMeanAcc	z-score normalization
latLaneCenterOffset	z-score normalization
deltaDist_ramp	Two-bin discretization
deltaV_ramp	Two-bin discretization
Longitudinal distance to front/rear vehicles	Binary encoding (“near” or “far”, threshold = 55 m)
Lateral distance to left/right vehicles	Binary encoding (“near” or “far”, threshold = 30 m)
Action space	One-hot encoding (accel+, accel–, keepSpeed, driftToward, driftAway)

Table B.4

Main hyperparameters of the HB-IRL training.

Category	Hyperparameter	Value
Data & structure	Number of clusters k	3
	Feature dimension D	59
	Total parameter dimension dim	355
MCMC configuration	Iterations/burn-in/thinning	60,000/8000/2
	Number of chains	10
	Inverse-temperature ladder $\{\beta_i\}$	[1, 0.90, ..., 0.36]
	Swap interval	80
Proposal & blocks	Proposal type	HMC
	Leapfrog steps L	6
	Initial scale	0.25
	Block dimension	4
	Small-block dimension	24
Adaptation	Big-jump interval	1500
	Step-size adaptation	dual-avg
	Target acceptance rate	0.60
	Covariance update interval	80
	Covariance adaptation rate α	0.02
Priors & reweighting	Prior variance σ_w^2	1.0
	Bias std scale	1.5
	Class reweight exponent γ	0.04
	IPS clipping	3
	Class-weight clipping	4

(Note: dual-avg is dual averaging, and IPS represents inverse propensity scoring.)

Table B.5

Maximum changes in validation metrics after perturbing locally unconverged dimensions.

Cluster	Max $ \Delta $ collision-free	Max $ \Delta $ pass ratio	Max $ \Delta $ travel time	Max $ \Delta $ avg speed	Max $ \Delta $ lane changes	Max $ \Delta $ drifts	Max $ \Delta $ jerk
C0	0.000	0.000	0.000	0.000	0.000	0.000	0.000
C1	0.000	0.000	0.000	0.000	0.000	0.000	0.000
C2	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table B.6

Discrete action space.

Index	Action	Description
0	speedUp	Increase longitudinal speed by a fixed increment
1	speedDown	Decrease longitudinal speed by a fixed increment
2	keepSpeed	Maintain the current longitudinal speed
3	driftToward	Apply a small lateral drift toward the ramp without lane change
4	driftAway	Apply a small lateral drift away from the ramp without lane change
5	laneChangeToward	Change lane toward the ramp
6	laneChangeAway	Change lane away from the ramp

Table B.7

State space.

Index	Feature name	Description
0	Speed	Longitudinal vehicle speed (m/s)
1–3	laneRegion one-hot	Far/near/ramp (one-hot encoded)
4	deltaDist	Euclidean distance between main vehicle and nearest neighbor on adjacent lane
5	frontDist	Distance to the front vehicle
6	rearDist	Distance to the rear vehicle
7	leftDist	Distance to the left vehicle
8	rightDist	Distance to the right vehicle
9	relV_main	Relative speed with the mainline vehicle
10	ttc_main	Time to collision with the mainline vehicle
11	relV_ramp	Relative speed with the ramp vehicle on the adjacent lane
12	ttc_ramp	Time to collision with the ramp vehicle
13	latOff	Lateral offset of the mainline vehicle (m)
14	histMeanAcc	Historical average longitudinal acceleration
15	step	Time step index

Table B.8
Configuration of background vehicles and traffic flows.

Parameter	Design Details
RL agent	r1_0
Background vehicle type	Aggressive, conservative, and stable (three types)
Entry time and position	Randomly sampled within the design range
Entry speed	Normal distribution $\mathcal{N}(25, 3.0^2)$ m/s, clipped to [18, 40] m/s
Mainline flow rate	Randomly sampled from (2500, 4000) veh/h
Ramp flow rate	Randomly sampled from (600, 1200) veh/h



Fig. B.14. Training convergence evidence for the cluster-specific RL policies (Cluster 0/1/2).

Table B.9
Traffic demand scaling under different traffic demand settings.

Traffic demand setting	Main factor	Ramp factor	Density type
Low demand	0.6	0.7	Low density
High demand	1.4	1.3	High density
Standard demand	1.0	1.0	Medium density (standard)

Table C.1
Glossary of field-specific terms and abbreviations used in this paper.

Abbreviation/Term	Full name
ACF	Autocorrelation Function
AV	Autonomous Vehicle
BGMM	Bayesian Gaussian Mixture Model
ECE	Expected Calibration Error
ESS	Effective Sample Size
exiD	Extended inD Dataset (exiD)
HB-IRL	Hierarchical Bayesian Inverse Reinforcement Learning
HDV	Human-Driven Vehicle
HMC	Hamiltonian Monte Carlo
IPS	Inverse Propensity Scoring
MCMC	Markov Chain Monte Carlo
NLL	Negative Log-Likelihood
PCA	Principal Component Analysis
PT	Parallel Tempering
QR-DQN	Quantile Regression Deep Q-Network
R-hat	Gelman–Rubin Convergence Diagnostic Statistic
RL	Reinforcement Learning
SUMO	Simulation of Urban MObility
TraCI	Traffic Control Interface
TTC	Time-to-Collision

References

Abou-Zeid, M., Kaysi, I., Al-Naghi, H., 2011. Measuring aggressive driving behavior using a driving simulator: An exploratory study. In: 3rd International Conference on Road Safety and Simulation, vol. 2011, pp. 1–19, URL: <http://onlinepubs.trb.org/onlinepubs/conferences/2011/RSS/1/Abou-Zeid,M.pdf>.

Alsaleh, R., Sayed, T., 2020. Modeling pedestrian-cyclist interactions in shared space using inverse reinforcement learning. *Transp. Res. Part F: Traffic Psychol. Behav.* 70, 37–57. <http://dx.doi.org/10.1016/j.trf.2020.02.007>.

Bhattacharyya, R., Wulfe, B., Phillips, D.J., Kuefler, A., Morton, J., Senanayake, R., Kochenderfer, M.J., 2022. Modeling human driving behavior through generative

adversarial imitation learning. *IEEE Trans. Intell. Transp. Syst.* 24 (3), 2874–2887. <http://dx.doi.org/10.1109/TITS.2022.3227738>.

Bucsuházy, K., Matuchová, E., Zúvala, R., Moravcová, P., Kostíková, M., Mikulec, R., 2020. Human factors contributing to the road traffic accident occurrence. *Transp. Res. Procedia* 45, 555–561. <http://dx.doi.org/10.1016/j.trpro.2020.03.057>.

Chandra, R., Wang, M., Schwager, M., Manocha, D., 2022. Game-theoretic planning for autonomous driving among risk-aware human drivers. In: 2022 International Conference on Robotics and Automation. ICRA, IEEE, pp. 2876–2883. <http://dx.doi.org/10.1109/ICRA46639.2022.9811865>.

Chen, S., Cheng, K., Yang, J., Zang, X., Luo, Q., Li, J., 2023. Driving behavior risk measurement and cluster analysis driven by vehicle trajectory data. *Appl. Sci.* 13 (9), 5675. <http://dx.doi.org/10.3390/app13095675>.

Chen, H., Zhang, Y., 2026. Mitigating aggression toward AVs in mixed traffic: Effects of automated driving styles and human driving styles. *Transp. Res. Part F: Traffic Psychol. Behav.* 116, 103431. <http://dx.doi.org/10.1016/j.trf.2025.103431>.

Chu, H.-C., 2020. Risky behaviors of older taxi drivers and suggested requirements for renewing their professional driver's licenses. *Transp. Res. Interdiscip. Perspect.* 8, 100272. <http://dx.doi.org/10.1016/j.trip.2020.100272>.

Dai, Q., Shen, D., Wang, J., Huang, S., Filev, D., 2022. Calibration of human driving behavior and preference using vehicle trajectory data. *Transp. Res. Part C: Emerg. Technol.* 145, 103916. <http://dx.doi.org/10.1016/j.trc.2022.103916>.

Eboli, L., Mazzulla, G., Pungillo, G., 2016. Combining speed and acceleration to define car users' safe or unsafe driving behaviour. *Transp. Res. Part C: Emerg. Technol.* 68, 113–125. <http://dx.doi.org/10.1016/j.trc.2016.04.002>.

Feng, F., Bao, S., Sayer, J.R., Flannagan, C., Manser, M., Wunderlich, R., 2017. Can vehicle longitudinal jerk be used to identify aggressive drivers? An examination using naturalistic driving data. *Accid. Anal. Prev.* 104, 125–136. <http://dx.doi.org/10.1016/j.aap.2017.04.012>.

Himmels, C., Parduzi, A., Löcken, A., Protschky, V., Venrooij, J., Riener, A., 2024. Validating risk behavior in driving simulation using naturalistic driving data. *Transp. Res. Part F: Traffic Psychol. Behav.* 107, 710–725. <http://dx.doi.org/10.1016/j.trf.2024.09.026>.

Houston, J., Harris, P., Norman, M., 2003. The aggressive driving behavior scale: Developing a self-report measure of unsafe driving practices. *North Am. J. Psychol.* 5, 269–278, URL: https://scholarship.rollins.edu/as_facpub/224/.

Huang, Z., Liu, H., Wu, J., Lv, C., 2023. Conditional predictive behavior planning with inverse reinforcement learning for human-like autonomous driving. *IEEE Trans. Intell. Transp. Syst.* 24 (7), 7244–7258. <http://dx.doi.org/10.1109/TITS.2023.3254579>.

Itkonen, T.H., Pekkanen, J., Lappi, O., Kosonen, I., Luttinen, T., Summala, H., 2017. Trade-off between jerk and time headway as an indicator of driving style. *PLoS One* 12 (10), e0185856. <http://dx.doi.org/10.1371/journal.pone.0185856>.

Jafarnejad, S., Castagnani, G., Engel, T., 2018. Revisiting gaussian mixture models for driver identification. In: 2018 IEEE International Conference on Vehicular Electronics and Safety. ICVES, IEEE, pp. 1–7. <http://dx.doi.org/10.1109/ICVES.2018.8519588>.

- Khanfar, N.O., Ashqar, H.I., Elhenawy, M., Hussain, Q., Hasasneh, A., Alhajjaseen, W.K., 2022. Application of unsupervised machine learning classification for the analysis of driver behavior in work zones in the state of Qatar. *Sustainability* 14 (22), 15184. <http://dx.doi.org/10.3390/su142215184>.
- Khanfar, N.O., Elhenawy, M., Ashqar, H.I., Hussain, Q., Alhajjaseen, W.K., 2023. Driving behavior classification at signalized intersections using vehicle kinematics: Application of unsupervised machine learning. *Int. J. Inj. Control. Saf. Promot.* 30 (1), 34–44. <http://dx.doi.org/10.1080/17457300.2022.2103573>.
- Kuniyoshi, J.R.G., Costa, A.T., Figueira, A.C., Kabbach Jr., F.I., Larocca, A.P.C., 2021. Driver's visual perception as a function of age. Using a driving simulator to explore driver's eye movements in vertical signs. *Transp. Res. Interdiscip. Perspect.* 11, 100460. <http://dx.doi.org/10.1016/j.trip.2021.100460>.
- Liu, Z.J., Wu, H.N., 2020. Driver behavior modeling via inverse reinforcement learning based on particle swarm optimization. In: 2020 Chinese Automation Congress. CAC, IEEE, pp. 7232–7237. <http://dx.doi.org/10.1109/CAC51589.2020.9327174>.
- Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wießner, E., 2018. Microscopic traffic simulation using SUMO. In: The 21st IEEE International Conference on Intelligent Transportation Systems. IEEE, pp. 2575–2582. <http://dx.doi.org/10.1109/ITSC.2018.8569938>, URL: <https://elib.dlr.de/124092/>.
- Moers, T., Vater, L., Krajewski, R., Bock, J., Zlocki, A., Eckstein, L., 2022. The exiD dataset: A real-world trajectory dataset of highly interactive highway scenarios in Germany. In: 2022 IEEE Intelligent Vehicles Symposium. IV, pp. 958–964. <http://dx.doi.org/10.1109/IV51971.2022.9827305>.
- Moins, T., Arbel, J., Dufloy, A., Girard, S., 2025. On the use of a local $\hat{R}$ to improve MCMC convergence diagnostic. *Bayesian Anal.* 20 (1), 131–156. <http://dx.doi.org/10.1214/23-BA1399>.
- Murphy, Y.L., Milton, R., Kiliaris, L., 2009. Driver's style classification using jerk analysis. In: 2009 IEEE Workshop on Computational Intelligence in Vehicles and Vehicular Systems. IEEE, pp. 23–28. <http://dx.doi.org/10.1109/CIVVS.2009.4938719>.
- Péter, B., Sütheö, G., Lakatos, I., 2025. Accident risk analysis of road accidents involving personal injury. *Period. Polytech. Transp. Eng.* 53 (3), 301–307. <http://dx.doi.org/10.3311/PPtr.38004>, URL: <https://pp.bme.hu/tr/article/view/38004>.
- Ramachandran, D., Amir, E., 2007. Bayesian inverse reinforcement learning. In: IJCAI, vol. 7, pp. 2586–2591, URL: <https://dl.acm.org/doi/10.5555/1625275.1625692>.
- Sayed, I., Abdelgawad, H., Said, D., 2022. Studying driving behavior and risk perception: a road safety perspective in Egypt. *J. Eng. Appl. Sci.* 69 (1), 22. <http://dx.doi.org/10.1186/s44147-021-00059-z>.
- Shahverdy, M., Fathy, M., Berangi, R., Sabokrou, M., 2020. Driver behavior detection and classification using deep convolutional neural networks. *Expert Syst. Appl.* 149, 113240. <http://dx.doi.org/10.1016/j.eswa.2020.113240>.
- Tang, J., Yu, S., Liu, F., Chen, X., Huang, H., 2019. A hierarchical prediction model for lane-changes based on combination of fuzzy C-means and adaptive neural network. *Expert Syst. Appl.* 130, 265–275. <http://dx.doi.org/10.1016/j.eswa.2019.04.032>.
- Tselentis, D.I., Papadimitriou, E., 2023. Machine learning approaches exploring the optimal number of driver profiles based on naturalistic driving data. *Transp. Res. Interdiscip. Perspect.* 21, 100900. <http://dx.doi.org/10.1016/j.trip.2023.100900>.
- Vats, D., Knudson, C., 2021. Revisiting the Gelman–Rubin diagnostic. *Statist. Sci.* 36 (4), 518–529. <http://dx.doi.org/10.1214/20-STS812>.
- Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., Bürkner, P.-C., 2021. Rank-normalization, folding, and localization: An improved R-Hat for assessing convergence of MCMC. *Bayesian Anal.* 16 (2), 667–718. <http://dx.doi.org/10.1214/20-BA1221>.
- Walenta, K., Genser, S., Solmaz, S., 2024. Bayesian Gaussian mixture models for enhanced radar sensor modeling: A data-driven approach towards sensor simulation for ADAS/AD development. *Sensors* 24 (7), 2177. <http://dx.doi.org/10.3390/s24072177>.
- Wang, Y., Niu, Y., Xiao, M., Zhu, W., You, X., 2024. Modeling framework of human driving behavior based on deep maximum entropy inverse reinforcement learning. *Phys. A* 652, 130052. <http://dx.doi.org/10.1016/j.physa.2024.130052>.
- Wang, W., Xi, J., Zhao, D., 2018. Learning and inferring a driver's braking action in car-following scenarios. *IEEE Trans. Veh. Technol.* 67 (5), 3887–3899. <http://dx.doi.org/10.1109/TVT.2018.2793889>.
- Wegener, A., Piórkowski, M., Raya, M., Hellbrück, H., Fischer, S., Hubaux, J.-P., 2008. TraCI: An interface for coupling road traffic and network simulators. In: 11TH Communications and Networking Simulation Symposium. CNS, ACM, New York, NY, USA, ISBN: 1-56555-318-7, pp. 155–163. <http://dx.doi.org/10.1145/1400713.1400740>, URL: <http://doi.acm.org/10.1145/1400713.1400740>.
- Wu, Z., Qu, F., Yang, L., Gong, J., 2022. Human-like decision making for autonomous vehicles at the intersection using inverse reinforcement learning. *Sensors* 22 (12), 4500. <http://dx.doi.org/10.3390/s22124500>.
- Xu, X., Han, W., Leng, B., Xiong, L., 2024. Driving Behavior Modeling Based on Inverse Reinforcement Learning. Technical Report, SAE Technical Paper, <http://dx.doi.org/10.4271/2024-01-7029>.
- Yang, K., Al Haddad, C., Yannis, G., Antoniou, C., 2022. Classification and evaluation of driving behavior safety levels: A driving simulation study. *IEEE Open J. Intell. Transp. Syst.* 3, 111–125. <http://dx.doi.org/10.1109/OJITS.2022.3149474>.
- Yang, C.H., Chang, C.C., Liang, D., 2018. A novel GMM-based behavioral modeling approach for smartwatch-based driver authentication. *Sensors* 18 (4), 1007. <http://dx.doi.org/10.3390/s18041007>.
- Yin, H., Wang, J., Lin, J., Han, D., Ying, C., Meng, Q., 2021. A memory-attention hierarchical model for driving-behavior recognition and motion prediction. *Int. J. Automot. Technol.* 22 (4), 895–908. <http://dx.doi.org/10.1007/s12239-021-0081-8>.
- Zhai, C., Jamson, S., Batool, Z., Jopson, A., 2023. The expression of anger while driving—the role of personality and self-consciousness in a sample of Chinese drivers. *Transp. Res. Interdiscip. Perspect.* 20, 100838. <http://dx.doi.org/10.1016/j.trip.2023.100838>.
- Zhu, Y., Ma, Y., Chen, S., Khattak, A.J., Pang, Q., 2022. Identifying potentially risky intersections for heavy-duty truck drivers based on individual driving styles. *Appl. Sci.* 12 (9), 4678. <http://dx.doi.org/10.3390/app12094678>.