MATRICS: A Multi-Agent Deep Reinforcement Learning-Based Traffic-Aware Intelligent Lane-Change System

Lokesh Chandra Das[1] and Myounggyu Won[2]

***Abstract*— We present MATRICS, a traffic-aware multi-agent reinforcement learning (MARL)-based intelligent lane-change system designed for autonomous vehicles (AVs). While existing research primarily focuses on enhancing the local impact of the ego vehicle's lane-change decisions, MATRICS stands out by optimizing both local and global performance, i.e., aiming not only to improve the traffic efficiency, driving safety, and driver comfort of the ego vehicle, but also to enhance overall traffic flow within a designated road segment. Through an extensive review of the transportation literature, we construct a novel state space integrating local traffic information collected from surrounding vehicles and global traffic data obtained from roadside units (RSUs). We develop a reward function to guide judicious lane-change decisions, considering both ego vehicle performance and traffic flow enhancement. Our local density-aware multi-agent double deep Q-network (DDQN) algorithm facilitates effective cooperation among agents in executing lane-change maneuvers. Simulation results demonstrate MATRICS' superior performance across metrics of traffic efficiency, driving safety, and driver comfort in comparison with a state-of-the-art MARL model.**

I. Introduction

Autonomous driving has garnered significant attention from the research community and industry, owing to its high potential for reducing traffic congestion and enhancing driving safety [1], [2]. Discretionary lane change for autonomous vehicles (AVs) is a key research focus in autonomous driving due to its substantial impact on both traffic efficiency and driving safety [3], [4]. Especially in dense or congested traffic environments, relying solely on the longitudinal motion control of AVs is insufficient for enhancing traffic flow. Timely and agile lane changes are critical to improve traffic efficiency [5]. However, making effective lane-change decisions for AVs can be particularly challenging, especially in complex and dynamic traffic environments where AVs and human-driven vehicles (HVs) coexist [6].

A substantial body of research has been conducted on lane-changing systems for AVs. The existing research work on lane-changing systems for AVs can be broadly classified into rule-based and machine learning-based approaches. Rule-based approaches rely on hand-crafted rules designed to emulate the behaviors of human drivers [7], [8]. However, despite their usefulness for specific scenarios, rule-based approaches suffer from limited generalization performance [9]. In response to the limitations of rule-based approaches, machine learning-based approaches have been the subject of active research especially concentrating on reinforcement learning (RL)-based methods [10], [6], [11].

We classify RL-based solutions into two main categories: the ego vehicle-based approaches [12], [13], [9], [14] and multi-agent-based approaches [15], [16], [2], [17], [18]. Ego vehicle-based approaches aim to enhance the efficiency and safety of the ego vehicle by incorporating surrounding vehicles as part of the state of the RL model. While ego vehicle-based approaches are highly scalable, they may result in poor performance since the complex interaction between vehicles is not effectively accounted for. In contrast, the MARL-based solutions exploit the close coordination of multiple agents to reach better lane-changing decisions for multiple autonomous vehicles, leading to higher traffic efficiency and safety. However, the majority of existing MARL-based solutions rely on onboard sensor data to make lane-changing decisions, thereby restricting the perception capability of the agents to the surrounding area [18], [1]. More importantly, the potential of intelligent lane-changing systems for improving the overall traffic flow of a road segment has not been fully leveraged.

In this paper, we propose a novel MARL-based intelligent lane-change system for AVs, namely MATRICS, that simultaneously optimizes the efficiency, safety, and driving comfort of the ego vehicle, along with the traffic flow of the road segment where the agents are located. To the best of our knowledge, this is the first study to present a hybrid intelligent lane-change approach that maximizes both local and global performance. Specifically, the proposed intelligent lane-change problem is formulated as a decentralized partially observable Markov decision process. We then develop a novel MARL framework to solve the problem. Our state space design incorporates both local traffic information obtained from surrounding vehicles of an ego vehicle, as well as global traffic information collected from the RSU responsible for managing the corresponding road segment. A novel reward function is created to effectively carry out the joint optimization for the performance of the ego vehicle and the overall improvement of traffic flow. A local density-aware multi-agent DDQN algorithm is designed to identify the optimal policy that each agent can use to collaboratively execute lane-change maneuvers taking into account the overall impact of lane-changing decisions. Extensive simulations were conducted to evaluate the performance of MATRICS in comparison with a state-of-the-art MARL model [18]. The results demonstrate the superior performance of MATRICS in all aspects of traffic efficiency, driving safety, and driver comfort.

[1]Lokesh Chandra Das is with the School of Computing, Wichita State University, Wichita, KS, United States lokesh.das@wichita.edu.
[2]Myounggyu Won is with the Department of Computer Science, University of Memphis, Memphis, TN, United States mwon@memphis.edu

II. Related Work

Our research has direct relevance to ML-based intelligent lane-changing systems. Interestingly, we found that numerous ML-based methodologies prioritize enhancing the safety and efficiency of the ego vehicle [12], [13], [9], [14] perceiving surrounding vehicles as the decision-making environment [10], [6], [11]. Wang *et al.* introduced an RL framework aimed at training a lane-changing model for the ego vehicle [12]. Their approach involved designing a Q-function based on a closed-form greedy policy and implementing a deep Q-learning algorithm tailored for continuous state and action spaces. Ye *et al.* focused on enhancing learning efficiency [13], devising a lane-changing strategy using proximal policy optimization-based deep RL. They demonstrated achieving stable performance with improved learning efficiency. Zhang *et al.* incorporated human factors into ego vehicle training [9], integrating driving styles of both ego and surrounding vehicles into their decision-making model. He *et al.* emphasized enhancing ML model robustness for intelligent lane-changing systems [14], proposing an observation adversarial RL approach to manage observation uncertainties arising from sensor noises, measurement errors, and adversarial perturbations. Despite these approaches' scalability in optimizing ego vehicle performance, they may not perform well in intricate traffic environments necessitating coordinated actions among multiple AVs [19].

Various MARL-based approaches have been proposed to overcome the limitations of ego vehicle-centric solutions [15], [16], [2], [17], [18]. These MARL-based methodologies are distinguished by their focus on enhancing traffic efficiency and safety in complex traffic environments by leveraging coordinated actions among multiple agents. Hou and Graf introduced a multi-agent deep RL approach aimed at controlling both the lane-changing behavior and longitudinal movement of the ego vehicle [17]. Their study demonstrated that this dual-control approach enhanced roadway capacity, particularly in highway merging and weaving areas. Zhou *et al.* developed a MARL-based approach utilizing the multi-agent advantage actor-critic network (MA2C) along with a novel parameter sharing scheme to foster agent collaboration [2]. Recognizing the incomplete consideration of passenger comfort in existing MARL-based approaches' reward functions, the authors prioritized designing an approach that simultaneously enhances traffic efficiency, safety, and driving comfort. Wang *et al.* devised a MARL-based lane-changing solution that operates independently of vehicle-to-everything (V2X) communication [1]. Their proposed method aims to optimize both individual and overall efficiency solely based on limited sensing results from individual vehicles. Zhang *et al.* proposed a bi-level approach, where the upper level comprises the MARL model for lane-change decisions, and the lower level manages potential conflicts during implementation [18]. This approach enhances efficiency by integrating the driving intentions of surrounding vehicles into the state space.

We made an interesting observation from state-of-the-art MARL-based approaches: although those methods significantly enhance the efficiency of lane-changing decisions through coordination among multiple agents, they often overlook the broader and more long-term impact of a lane-changing action on the overall traffic flow within a specific road segment. This oversight occurs because these approaches predominantly focus on the efficiency and safety of the ego vehicle. However, with the advent of V2X technology, agents can now obtain a more expansive view of the traffic environment. In this paper, we argue for the necessity of a novel solution that facilitates agent coordination to not only improve overall traffic flow, but also simultaneously optimize individual efficiency, safety, and driving comfort.

III. Design of MATRICS

A. Overview

Fig. 1 depicts an operational overview of MATRICS. As shown, we consider a road segment where AVs and HVs coexist. We assume that a RSU is deployed to gather traffic data from the road segment such as vehicle speed, vehicle type, *etc*. AVs utilize the lane-change model of MATRICS to make efficient lane-change decisions. This decision-making process aims at optimizing the traffic flow of the road segment while enhancing the efficiency, safety, and comfort of the ego vehicle based on the model built using both the local traffic information sourced from the surrounding vehicles via onboard sensors of the ego vehicle, as well as the global information received from the RSU through V2X communication.

The lane-change model of MATRICS is represented as a decentralized Partially Observable Markov Decision Process (decPOMDP). The decPOMDP is described by the tuple $(\{\mathcal{A}_i, \mathcal{O}_i, \mathcal{R}_i\}_{i\in\mathcal{V}}, \mathcal{T}, \mathcal{S})$ where $\mathcal{A}_i$ is an action space for an agent $i \in \mathcal{V}$, where $\mathcal{V}$ is a set of agents. $\mathcal{O}_i$ is a partial observation of the environmental state $\mathcal{S}$ for agent i. $\mathcal{T}$ is a set of transitional probabilities which is unknown. The goal is to find a decentralized policy $\pi_i : \mathcal{O}_i \times \mathcal{S} \rightarrow [0, 1]$ for each agent i to choose an action a_t at time t such that the expected cumulative reward is maximized.

B. State Space

The state space $\mathcal{S}$ for MATRICS is designed to incorporate both the local and global traffic information to simultaneously enhance the performance of the ego vehicle and the overall traffic flow. The local traffic information (*i.e.,* partial information $\mathcal{O}_i$) includes the state of the ego vehicle i and the states of its surrounding vehicles, *i.e.,* the vehicles within a 100-meter radius of the ego vehicle. The state of the ego vehicle consists of its longitudinal and lateral position (x, y), speed (v), acceleration (a), local density (ρ_c), and lateral safety distances (d) of leader and follower from two adjacent lanes (See Fig. 2 for definitions of the lateral distance, leader, and follower). The set of states for surrounding vehicles is represented as a 2D vector $N \times \mathcal{F}$, where N is the number

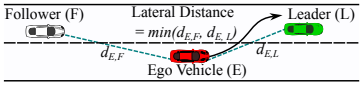


Fig. 2: The lateral distance, leader, and follower.

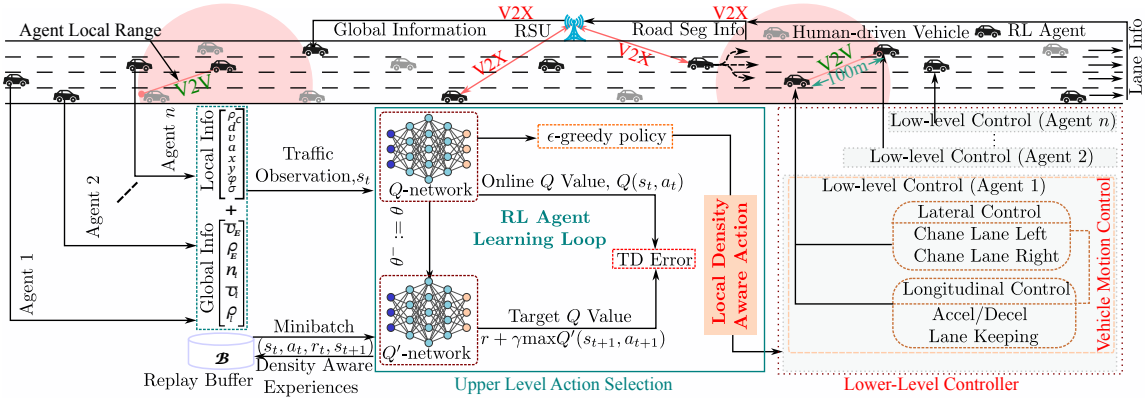


Fig. 1: An operational overview of MATRICS.

of surrounding vehicles and $\mathcal{F}$ is the feature set that consists of its distance to the ego vehicle, speed, acceleration, and an additional factor termed driver imperfection (σ). This driver imperfection element is incorporated to reflect the driving behavior of human drivers (HVs) to allow for more effective lane-change decisions [20]. Additionally, the global traffic information encompasses various key metrics of the roadway segment, such as vehicle density ($\bar{\rho_E}$), average speed of vehicles ($\bar{v}_E$), the maximum speed limit, and the number of lanes (n_l). It also includes traffic information for each lane, *i.e.,* average speed ($\bar{v_l}$) and vehicle density ($\bar{\rho_l}$) specific to that lane.

C. Action Space

The action space $\mathcal{A}_i$ of agent i is composed of five distinct discrete actions, *i.e.,* $\mathcal{A}_i = \{\text{left, right, keep, accelerate, decelerate}\}$. More specifically, agent i can undertake the following actions: (i) switching to the left lane, (ii) switching to the right lane, (iii) maintaining the current speed, (iv) accelerating, and (v) decelerating. In this study, the interval of taking an action is set to 0.1 seconds. For the implementation of acceleration and deceleration actions, we utilize a low-level controller based on the Extended Intelligent Driver Model (EIDM) [21]. More specifically, the amount of acceleration or deceleration applied $a(t+\Delta t)$ in accordance with the acceleration or deceleration action is calculated as follows.

$$a(t+\Delta t) = \begin{cases} a_{max}\left[1-\left(\frac{s^*_{i-1}(t)}{s(t)}\right)^2\right], \text{ if } s^*_{i-1}(t) \geq s(t) \\ a_{free}(t)\left[1-\left(\frac{s^*_{i-1}(t)}{s(t)}\right)^{\frac{2*a_{max}}{|a_{free}(t)|}}\right], \text{ otherwise,} \end{cases}$$

$$a_{free}(t) = a_{max}\left[1-\left(\frac{v_{i-1}(t)}{v_0(t)}\right)^{\delta}\right].$$

$$s^*_{i-1}(t) = s_0 + max\Big(0, v_{i-1}(t) * T - \frac{v_{i-1}(t) * (v_i(t) - v_{i-1}(t))}{2 * \sqrt{a_{max} * b}}\Big),$$

where v_i and v_{i-1} are the speed of the leading vehicle and following vehicle, respectively, $s(t)$ is the actual gap between the leading and following vehicles, $s^*_{i-1}(t)$ is the desired gap between the vehicles, $v_0(t)$ is the desired velocity, δ is the acceleration exponent, a_{max} is the maximum acceleration, b is the comfortable deceleration, and T is the time headway. The parameter values used for EIDM are listed in Table I.

Additionally, the lower-level controller intervenes and assumes control from the RL agent if it fails to execute a critical action necessary to maintain safe distance from the vehicle ahead. More specifically, it estimates the time-to-collision (TTC) [22] to the front vehicle using Eq. 1 and takes over the control if the $TTC \leq 0.8s$.

$$TTC = \begin{cases} \frac{x_i - x_{i-1} - len_i}{v_{i-1} - v_i}, & \text{if } v_{i-1} > v_i \\ \infty, & \text{otherwise,} \end{cases} \quad (1)$$

where x is the longitudinal position and len_i is the leader length.

D. Reward Function

The reward function R_i for agent i is comprised of five subfunctions, which represent the traffic efficiency r_e, driving safety r_s, and driver comfort r_c, lane-change utility r_u, and compliance with the low-level controller r_l:

$$R_i = r_e + r_s + w_c r_c + w_u r_u + w_l r_l,$$

where w_c, w_u, and w_l are weighting parameters for comfort, lane-change utility, and compliance with low-level control, respectively, which are tuned emperically. It is worth mentioning in MATRICS, safety is prioritized most, followed by efficiency and comfort.

Efficiency Reward: The traffic efficiency reward function r_e not only incentivizes RL agents to improve their own speed but also promotes the traffic flow within the road segment. More specifically, the efficiency reward function r_e is defined as follows.

$$r_e = w_{ge} g_e + w_{le} l_e,$$

where g_e denotes the global efficiency; l_e denotes the local efficiency; and w_{ge} and w_{le} are weighting coefficients. The global efficiency represents the effect of the ego vehicle's action on the traffic flow for the road segment. More specifically, the global efficiency g_e is defined as follows.

$$g_e = \begin{cases} -(\bar{v}_e - \bar{v}_M)/\bar{v_M}, & \text{if } \bar{v}_e > \bar{v}_M \\ (\bar{v}_e - \bar{v}_m)/\bar{v}_m, & \text{if } \bar{v}_m \leq \bar{v}_e \leq \bar{v}_M \\ -(\bar{v}_m - \bar{v}_e)/\bar{v}_m, & \text{otherwise,} \end{cases}$$

where $\bar{v}_e$ is the average speed of vehicles in the road segment; $\bar{v}_m$ and $\bar{v}_M$ are the minimum and maximum average speed known for the road segment, respectively (Table I). A positive reward is granted when the ego vehicle's lane change decision leads to an improvement in the average speed $\bar{v}_e$, provided that it lies between $\bar{v}_m$ and $\bar{v}_M$. Otherwise, a negative reward is issued. When computing the reward, we ensure that it is proportional to the difference between $\bar{v}_e$ and $\bar{v}_m$ (or $\bar{v}_M$).

The local efficiency l_e is determined based on the speed of the ego vehicle v_{ego} in comparison with the minimum speed limit, v_m and the maximum speed limit, v_M of the road segment, *i.e.,*

$$l_e = \begin{cases} -(v_{ego} - v_M)/v_M, & \text{if } v_{ego} > v_M \\ (v_{ego} - v_m)/v_m, & \text{if } v_m \leq v_{ego} \leq v_M \\ -(v_m - v_{ego})/v_m, & \text{otherwise,} \end{cases}$$

Safety Reward: The safety reward function r_s is designed to promote the safety of both the ego vehicle and surrounding vehicles. It consists of the lateral and longitudinal safety of the ego vehicle and potential collisions with surrounding vehicles:

$$r_s = w_{lon} s_{lon} + w_{lat} s_{lat} + w_{col} s_{col}.$$

The longitudinal safety component s_{lon} is designed to ensure that the agent maintains the minimum safety distance to the front vehicle when performing lane changes as follows.

$$s_{lon} = \begin{cases} (\delta_x - \delta_{t_x})/\delta_{t_x}, & \text{if } \delta_x \leq \delta_{t_x} \\ 0, & \text{otherwise,} \end{cases}$$

where δ_x is the longitudinal distance; and δ_{t_x} is a threshold for the longitudinal distance. In particular, δ_{t_x} is defined differently for different vehicle types to enforce more sophisticated control. More specifically, δ_{t_x} is defined as $\widehat{v_M}\Delta t + len + s_0$, where $\widehat{v_M}$ is the maximum speed defined individually for each vehicle type; Δt is the time step (0.1 seconds); len is the length of the vehicle; and s_0 is the minimum gap at standstill.

The lateral safety component s_{lat} is designed to encourage an agent to maintain a safe distance from the leader and follower on the target lane to avoid a collision when performing a lane-change, which is defined as follows.

$$s_{lat} = \begin{cases} (\delta_y - \delta_{t_y})/\delta_{t_y}, & \text{if } \delta_y \leq \ \delta_{t_y} \\ 0, & \text{otherwise,} \end{cases}$$

where δ_y is the lateral distance; δ_{t_y} is the minimum lateral distance, which is set to 10m according to the literature [13]. Basically, the reward function penalizes an agent if the agent attempts to change lanes when its lateral distance is smaller than the minimum lateral distance.

The collision component s_{col} imposes a significant penalty on the agent in the event of a collision with surrounding vehicles, *i.e.,*

$$s_{col} = \begin{cases} -5, & \text{if collides with other vehicle} \\ 0, & \text{otherwise.} \end{cases}$$

Comfort Reward: We employ an existing methodology to quantify driving comfort [23], [24], [25]. Specifically, the driving comfort reward function is designed based on the change rate of acceleration (Δa) as follows.

$$r_c = -\Delta a / jerk_{max},$$

where *jerk* is calculated based on the acceleration range. For example, in our simulation, an agent makes a decision every 0.1 seconds, and the acceleration range is between 2.6m/s^2 and -2.6m/s^2. Hence, the maximum possible *jerk* is $(2.6 - (-2.6))/0.1 = 52$.

Lane Change Utility Reward: Although the reward function R_i is designed to maximize traffic efficiency, driving safety, and driver comfort, we observe that agents make invalid lane-change decisions, especially during the initial training period. To facilitate the training process and achieve safe and informed lane-change decisions, we incorporate the lane change utility reward function r_u. It penalizes an agent with a negative reward when it makes an invalid lane-change decision which includes: (1) a lane-change decision to switch to the left lane when the vehicle is already in the leftmost lane (-0.5), (2) a lane-change decision to switch to the right lane when the vehicle is already in the rightmost lane (-0.5), (3) a lane-change decision when there is no traffic in front of the vehicle (-0.5), and (4) a lane-change decision when the leading vehicle in the target lane is moving slower than the agent (-0.5).

Low-level Control Reward: If an upper level action of acceleration/deceleration does not comply with the EIDM estimation, the action is corrected, and a negative feedback of $r_l = -0.01$ is provided to the upper-level controller so that MATRICS can learn correct actions in subsequent steps.

E. Local Density-Aware MARL Double DQN Algorithm

We develop a novel local density-aware MARL Double DQN (DDQN) algorithm to determine the optimal policy to facilitate efficient coordination of multiple agents in maximizing the cumulative reward. We adopt DDQN as it aligns well with our discrete action space, enables sample-efficient learning, and has low computational overhead. A notable feature of this algorithm lies in its capacity to discern agents' actions based on the *local density,* ρ_{local}, which refers to the number of vehicles in the agent's vicinity. The rationale behind this approach is the insight that an agent's decision to change lanes in a densely populated segment of the road has a more significant impact on overall traffic efficiency, driving safety, and driver comfort compared to similar decisions made in areas of lighter traffic. The MARL DDQN algorithm adopts the centralized training and decentralized execution paradigm, a strategy wherein each agent benefits from information and parameters gleaned by other agents during centralized training, yet autonomously utilizes only local observations to dictate actions during decentralized execution [26]. Given the homogeneous nature of the agents, we employ the parameter sharing method [27], enabling agents to collectively utilize a single network's architecture and parameters. Simultaneously, agents receive

distinct observations tailored to their communication range and independently execute actions [28].

The algorithm 1 outlines the pseudocode for the local density-aware MARL DDQN. As shown, at every simulation step of training, agents perform actions independently following the $\epsilon-$greedy policy [29], and store their transitions in the shared replay buffer. A novel aspect of our approach is that the probability p of an agent executing an action is proportionately determined by the local traffic density, *i.e., p* is defined as the number of vehicles in the range of the ego vehicle divided by the maximum number of vehicles for the range. Specifically, an agent located in a denser traffic area is assigned a higher probability for action execution. This method ensures that actions are more likely to be taken in areas where they can have a greater impact on improving traffic conditions. During the evaluation time, each agent uses its own copy of the learned policy network to perform individual actions. We trained the policy network by taking a minibatch of transitions sampled uniformly at random from the replay buffer at every simulation time step to optimize the network parameters according to the following equation.

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E}_{\phi,a\sim\rho(.);\phi'}\Big[\Big((r+\gamma Q(\phi',\ \underset{a}{\text{argmax}}\ Q(\phi',a;\theta_t),\ \theta_t^-)) - Q(\phi,a;\theta_i)\Big)\Delta_{\theta_i} Q(\phi,a;\theta_i)\Big],$$

where $Q(\phi,a;\theta_i)$, and $Q(\phi\prime,a;\theta_i^-)$ denote the Q-values of the network with weights θ_i and θ_i^- for iteration i, $\rho(\phi,a)$ is a probability distribution over observations ϕ and actions a; r is the reward and γ is the discount factor.

IV. Simulation Results

We implemented MATRICS and a state-of-the-art MARL-based lane-change solution [18], referred to as NP-MARL, within the SUMO traffic simulator [30]. The MATRICS and NP-MARL models were constructed using PyTorch [31], and integrated with SUMO through the libsumo interface. We trained both models using 10 different random seeds to mitigate the impact of model initialization on its performance. Fig. 3 illustrates the rewards from these 10 runs. Each training episode spanned for 6 minutes of simulation time where initial $60s$ were used as a buffer time to have enough traffic. An update interval of $0.1s$ was used during training, *i.e.,* 3000 gradient steps were performed in an episode to adjust the parameters. During testing, each episode lasted for 10 minutes of simulation time and 20 such episodes were used to represent a single data point. The optimal hyperparameter values used for training were identified based on the grid search method [32] (Table I). Training and testing of the MARL model were conducted on an Apple Mac mini equipped with M2 Chip and 16GB of RAM, running on MacOS (Sequoia 15.1.1). Fig. 4 shows the average computational time of MATRICS from 1000 inference times.

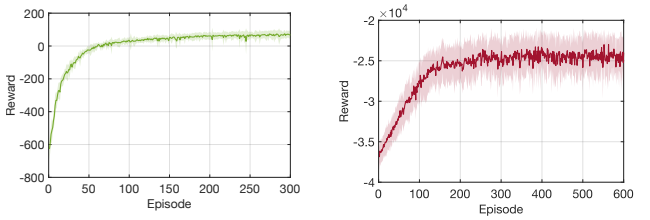
For this simulation, we considered a 3.25km long, 5-lane road segment. The initial 250m of the road segment served as a zone for vehicle injection and acceleration for both AVs and HVs. Subsequently, the next 3km was the area

Algorithm 1 Local Density-Aware MARL DDQN Algorithm

Require: Episodes ***E***, episode duration ***T***, discount factor γ, memory size ***M***, network update period C, learning rate α, max density, ρ_{max}

Ensure: Action $a \in$ {right, left, accel, decel, keep}

```
Initialize replay memory 𝒟 to size M, policy network Q
with random weights θ, target network Q̂ parameters θ⁻
with policy network weights θ
for episode e = 1 : E do
  for agent i = 1 : 𝒩 do
    get initial observation s_1^i and compute φ_1^i
  end for
  for t = 1 : T do
    for agent i = 1 : 𝒩 do
      Select random action a_t^i with probability ε
      Otherwise, action a_t^i = max_a Q(φ(s_t^i), a; θ)
      Action probability, p_a = ρ_local / ρ_max
      if p_a ≥ X ~ U([0,1]) then
        Apply action a_t^i
        Get next state s_{t+1}^i and collect reward r_t^i
        compute φ_{t+1}
        Store the transition (φ_t, a_t, r_t, φ_{t+1}) in 𝒟
      end if
    end for
    Sample minibatch of b transitions (φ_j, a_j, r_j, φ_{j+1})
    at random from 𝒟
    for trainsition i = 1 : b do
      Update θ ← θ + α Δ_{θ_i} L_i(θ_i)
    end for
    Update weights θ⁻ = θ at every C gradient steps
  end for
end for
```



(a) MATRICS (b) NP-MARL

Fig. 3: Convergence of the reward functions.

where RL agents executed lane-changes. The speed limit for the road segment was fixed at 33.5m/s. Vehicles were injected following the Piosson distribution at a rate of 1800 veh/h/lane. The proportion of AVs was varied for evaluation purposes. The position and lane for each vehicle within the initial 250m of the road segment were randomly selected. To ensure a sufficient number of HVs were present, AVs were introduced into the network after a 60-second delay. The acceleration range of each vehicle was between $-2.6m/s^2$ and $2.6m/s^2$. The simulation leveraged the intelligent driver model (IDM) [33], [30] (using SUMO's default values) to mimic the driving behavior of HVs, alongside a widely-employed lane-change model [34] for simulating HVs' lane-change behavior. To enhance realism, four distinct types of HVs, characterized by varying body lengths, speed limits,

and driving imperfections, were incorporated. The maximum and minimum speed limits for HVs were set to 17.9m/s to 24.6m/s, respectively. In our simulation, 90% of HVs maintained speeds within 75% to 125% of the prescribed limits.

TABLE I: Hyper/parameter values for MATRICS.

Optimal Hyperparameters	
Minibatch size, b	64
Replay buffer size, M	500k
Discount factor, γ	0.999
Epsilon $(\epsilon, \epsilon_{min}, \epsilon_{decay})$	(1.0, 0.001, 0.999985)
Learning rate, α	1e-4
Target network update freq., C	20000 gradient steps
State & action space size	(46, 5)
Activation function (hidden/output)	ReLU/linear
Loss function & optimizer	Huber loss [35], AdamW
Network size	4 hidden layers with 256, 512, 256, 128 hidden units
Simulation Parameters	
Maximum accleration, a_{max}	$2.6m/s^2$
Comfortable deceleration, b	$-2.6m/s^2$
Acceleration exponent, δ	2
Time headway, T	$0.9s$
(Min gap, time step), $(s_o, \Delta t)$	$(2.5m, 0.1s)$
(w_c, w_u, w_l, w_{ge})	$(0.1, 0.08, 1, 0.06)$
$(w_{le}, w_{lon}, w_{lat}, w_{col})$	$(0.08, 1.5, 1.5, 1.5)$
Speed limits, (v_m, v_M, v_m, v_M)	$(20.56, 23.69, 20.11, 33.5)m/s$

Following the methodology in the literature [18], traffic efficiency was gauged by the average speed of all vehicles on a road segment; traffic safety was assessed based on the collision rate recorded during the simulation, and driver comfort was evaluated by analyzing the variations in vehicle acceleration and deceleration.

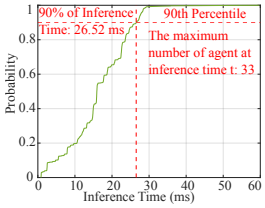


Fig. 4: Inference time.

A. Traffic Efficiency

Fig. 5 presents an overview of the temporal changes of traffic patterns for MATRICS and NP-MARL. NP-MARL led to reduced vehicle speeds across the highway segment compared to MATRICS, partly because it failed to consider the cumulative impact of lane-changing actions on overall traffic flow. In contrast, the result demonstrates that the coordinated lane-change decisions by MATRICS improved the speed of vehicles throughout the highway.

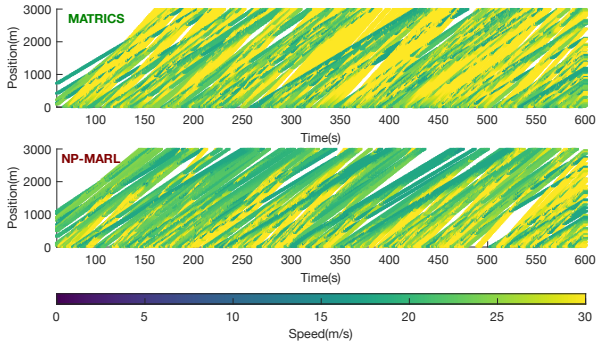


Fig. 5: A space-time diagram of traffic efficiency.

We further conducted a comprehensive analysis by tracking the average speed of all vehicles over time for MATRICS and NP-MARL. The results are depicted in Fig. 6. Initially, during the phase of traffic injection, the difference in average speed between MATRICS and NP-MARL was minimal. However, the result shows that MATRICS's ability to execute highly coordinated lane changes became increasingly evident as the highway segment became more congested. Specifically, MATRICS managed to sustain an average speed that was 8.9% higher than NP-MARL throughout the simulation period. Interestingly, it was observed that a significant number of agents opted to refrain from lane changes when MATRICS was used, which in turn facilitated higher speeds for surrounding vehicles, thereby enhancing the overall traffic flow on the segment. Additionally, as depicted in Fig. 6, the implementation of our local density-aware MARL DDQN algorithm within MATRICS enhanced traffic efficiency by 3.7% compared to a standard MARL DDQN algorithm.

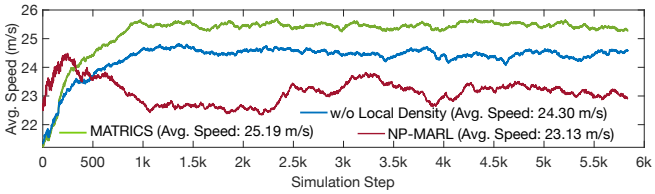
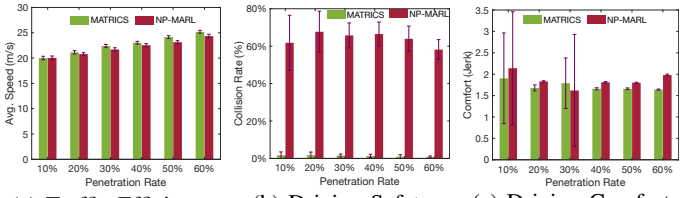


Fig. 6: The average speed changes over time for MATRICS, NP-MARL, and MATRICS w/o the local density module.

We also assessed the impact of agent penetration rate, defined as the proportion of agents to the total number of vehicles introduced into the road segment, on traffic efficiency. The outcomes are presented in Fig. 7a. The result indicates an overall enhancement in traffic efficiency for both systems as agent penetration rate increased. Two principal observations were made. Firstly, MATRICS consistently surpassed NP-MARL in performance across all levels of agent penetration rate, with MATRICS achieving an average speed that was 8.65% greater than NP-MARL's. Remarkably, MATRICS demonstrated a more pronounced improvement in performance at higher agent densities. For instance, at an agent penetration rate of 60%, MATRICS recorded a performance enhancement of 8.9% over NP-MARL, whereas at a lower agent density of 10%, the performance advantage was modestly higher at 4.7% compared to NP-MARL.



(a) Traffic Efficiency (b) Driving Safety (c) Driving Comfort

Fig. 7: Traffic efficiency, safety, and driver comfort for MATRICS and NP-MARL with varying penetration rates.

B. Driving Safety

MATRICS is engineered not only to optimize traffic efficiency but also to elevate driving safety and comfort.

In this subsection, we concentrate on assessing MATRICS' contributions to driving safety enhancement in comparison with NP-MARL. By default, SUMO ensures safe longitudinal and lateral maneuvers. To assess the capability of MATRICS, we disable the SUMO's default safety check and lane change mode and gauge driving safety for both MATRICS and NP-MARL by monitoring collision rates across various agent penetration rates. The findings are illustrated in Fig. 7b, indicating a general improvement in driving safety for both MATRICS and NP-MARL as agent density increases. In particular, a comparative examination reveals MATRICS outperforms NP-MARL in driving safety across all agent penetration levels, with the performance gap widening at higher densities. Specifically, MATRICS exhibits a noteworthy enhancement in driving safety, achieving an 98.69% greater safety margin compared to NP-MARL.

C. Driver Comfort

To evaluate the impact of lane-change maneuvers on driver comfort, we quantified driver comfort using jerk, defined as the derivative of acceleration with respect to time, *i.e.,* higher jerk values mean lower driving comfort. The results are shown in Fig. 7c. As shown, the agent density increases, driving comfort improves. This is potentially due to a more coordinated lane changes of the agents leading to improved driving comfort. It is also observed that the average jerk values for both MATRICS and NP-MARL mostly remain within the comfortable level (jerk values smaller than 2.94 are considered comfortable [36]). The results also show that MATRICS outperforms NP-MARL, delivering up to a 17.2% improvement in jerk reduction. This improvement underscores MATRICS's effectiveness in optimizing driving comfort through better-coordinated lane-changing strategies.

D. Ablation Study

An ablation study was conducted to assess the impact of key components of MATRICS: the lane-change utility reward r_u and the local density module of our MARL DDQN algorithm. Table II presents traffic efficiency, safety, and driver comfort for MATRICS and the three methods with varying penetration rates.

Efficiency: We observed that MATRICS, equipped with the local-density module and lane-change utility reward, outperforms all counterparts across various penetration rates. The advantages of MATRICS are more pronounced at higher penetration rates. Specifically, MATRICS achieves a 3.7% higher efficiency than MATRICS w/o local-density, a 10.6% improvement over NP-MARL, and a 2% increase in traffic efficiency compared to MATRICS w/o lane-change utility at penetration rates between 50% and 60%.

Safety: We also observed a distinct improvement in traffic safety due to the local-density module and lane-change utility reward. As shown in Table II, MATRICS consistently enhances driving safety across all penetration rates, maintaining a higher margin compared to other methods. Notably, at a penetration rate of 60%, MATRICS demonstrates significantly enhanced driving safety, with improvements of 67%, 98.7%, and 71.4% compared to MATRICS w/o local-density, NP-MARL, and MATRICS w/o lane-change utility, respectively.

Driving Comfort: Similar to its improvements in traffic efficiency and driving safety, MATRICS also excels in enhancing driving comfort. Specifically, MATRICS maintains a 3% greater driving comfort compared to MATRICS w/o local-density, while outperforming NP-MARL and MATRICS w/o lane-change utility by up to 17.2% and 1.8%, respectively. Overall, our ablation study highlights the importance of integrating the local-density module and lane-change utility reward to facilitate safe, efficient, and informed lane-change decisions.

Another critical component of MATRICS is its safety reward function, which contrasts sharply with NP-MARL. NP-MARL exhibits poor performance in traffic safety, as illustrated in Fig. 7b, primarily due to its simplistic safety reward function design and heavy dependence on the low-level controller for safety assurance. Table III shows the collision rates with and without the safety reward across different penetration rates. Without the safety reward, MATRICS also displays a higher collision rate similar to NP-MARL. However, with the safety module, MATRICS achieves over 95% reduction in collision rates.

Finally, we analyze the number of invalid lane changes, as detailed in Section III-D, with a specific focus on the impact of the lane-change utility reward.A well designed lane-change utility function can guide an agent when and where to perform lane-changes effectively to improve traffic safety, efficiency, and driving comfort. Table IV presents the average number of invalid lane changes by AVs over 20 different runs. A notable observation was that MATRICS with the lane-change utility recorded 81.6% fewer invalid lane changes compared to MATRICS without the lane-change utility function, and 98% fewer compared to NP-MARL. These results highlight the crucial role of the lane-change utility reward in reducing invalid lane changes.

V. Conclusion

MATRICS represents a MARL-based intelligent lane-change system that prioritizes the performance of individual vehicle while enhancing overall traffic dynamics. By integrating both local and global traffic data into a novel state space and developing a comprehensive reward function, MATRICS facilitates cooperative lane-change decisions that improve traffic efficiency, driving safety, and driver comfort. Our comparative simulations confirm that MATRICS outperforms existing state-of-the-art MARL models, underscoring its potential to optimize traffic flow and vehicle operation on busy roadways.

References

[1] G. Wang, J. Hu, Z. Li, and L. Li, "Harmonious lane changing via deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 5, pp. 4642–4650, 2021.

[2] W. Zhou, D. Chen, J. Yan, Z. Li, H. Yin, and W. Ge, "Multi-agent reinforcement learning for cooperative lane changing of connected and autonomous vehicles in mixed traffic," *Autonomous Intelligent Systems*, vol. 2, no. 1, pp. 1–11, 2022.

TABLE II: Comparison of traffic efficiency, safety, and comfort of MATRICS with MATRICS w/o local density, MATRICS w/o lane-change utility, and NP-MARL.

-	Traffic Efficiency (Avg. Speed in m/s)						Driving Safety (Rate of Collision in%)						Driving Comfort (Jerk)					
	AV Penetration Rate						AV Penetration Rate						AV Penetration Rate					
	10%	20%	30%	40%	50%	60%	10%	20%	30%	40%	50%	60%	10%	20%	30%	40%	50%	60%
MATRICS	**20.01 ±0.35**	**21.14 ±0.34**	**22.39 ±0.32**	**23.03 ±0.28**	**24.14 ±0.28**	**25.19 ±0.29**	**1.69 ±1.76**	**1.72 ±1.66**	**1.49 ±0.08**	**1.32 ±0.09**	**0.95 ±0.1**	**0.76 ±0.5**	**1.9 ±1.06**	**1.68 ±0.069**	1.79 ±0.59	**1.66 ±0.02**	**1.66 ±0.02**	**1.64 ±0.01**
w/o Local Density	19.97 ±0.26	20.89 ±0.34	21.91 ±0.31	22.50 ±0.32	23.34 ±0.26	24.30 ±0.36	2.04 ±2.1	2.43 ±1.78	2.8 ±1.4	2.24 ±1.4	2.35 ±1.29	2.3 ±1.04	1.96 ±1.04	1.72 ±0.032	1.83 ±0.59	1.71 ±0.02	1.70 ±0.01	1.69 ±0.01
w/o LC Utility	19.90 ±0.34	20.83 ±0.32	22.02 ±0.37	22.64 ±0.32	23.66 ±0.34	24.73 ±0.38	1.77 ±2.39	1.9 ±2.2	2.3 ±1.4	2.81 ±2	2.97 ±1.62	2.66 ±1	1.96 ±1.06	1.72 ±0.02	**1.55 ±1.39**	1.70 ±0.014	1.69 ±0.017	1.67 ±0.017
NP-MARL	19.11 ±0.26	19.60 ±0.29	20.32 ±0.25	21.0 ±0.37	21.82 ±0.45	23.13 ±0.81	61.82 ±14.71	67.71 ±10.97	65.75 ±6.67	66.48 ±6.38	63.93 ±6.78	58.17 ±5.33	2.14 ±1.32	1.83 ±0.017	1.62 ±1.31	1.81 ±0.018	1.80 ±0.013	1.98 ±0.021

TABLE III: Collision rate (%) w/o the safety module.

Methods/PR	10%	20%	30%	40%	50%	60%
MATRICS	**1.69 ±1.76**	**1.72 ±1.66**	**1.49 ±0.08**	**1.32 ±0.09**	**0.95 ±0.1**	**0.76 ±0.5**
w/o Safety	42.81 ±14.0	37.57 ±15.67	31.99 ±5.4	26.87 ±5.4	22.15 ±9.04	16.89 ±3.26

TABLE IV: Number of invalid lane changes at 60% PR.

	MATRICS	w/o local Density	w/o LC Utility	NP-MARL
Number of Invalid LC	**3568.2 ±767.93**	4640.95 ±966.55	19440.15 ±2924.94	180218.4 ±11583

[3] J. Nilsson, M. Brännström, E. Coelingh, and J. Fredriksson, "Lane change maneuvers for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1087–1096, 2016.

[4] Y. Zheng, B. Ran, X. Qu, J. Zhang, and Y. Lin, "Cooperative lane changing strategies to improve traffic operation and safety nearby freeway off-ramps in a connected and automated vehicles environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4605–4614, 2019.

[5] G. Li, Y. Yang, S. Li, X. Qu, N. Lyu, and S. E. Li, "Decision making of autonomous vehicles in lane change scenarios: Deep reinforcement learning approaches with risk awareness," *Transportation Research Part C: Emerging Technologies*, vol. 134, p. 103452, 2022.

[6] D. Chen, L. Jiang, Y. Wang, and Z. Li, "Autonomous driving using safe reinforcement learning by incorporating a regret-based human lane-changing decision model," in *Proc. of ACC*, 2020.

[7] K. Ahmed, M. Ben-Akiva, H. Koutsopoulos, and R. Mishalani, "Models of freeway lane changing and gap acceptance behavior," *Transportation and Traffic Theory*, vol. 13, pp. 501–515, 1996.

[8] D. Sun and L. Elefteriadou, "Lane-changing behavior on urban streets: An "in-vehicle" field experiment-based study," *Computer-Aided Civil and Infrastructure Engineering*, vol. 27, no. 7, pp. 525–542, 2012.

[9] Y. Zhang, Q. Xu, J. Wang, K. Wu, Z. Zheng, and K. Lu, "A learning-based discretionary lane-change decision-making model with driving style awareness," *IEEE TITS*, vol. 24, no. 1, pp. 68–78, 2023.

[10] R. Du, S. Chen, Y. Li, J. Dong, P. Y. J. Ha, and S. Labi, "A cooperative control framework for CAV lane change in a mixed traffic environment," *arXiv preprint arXiv:2010.05439*, 2020.

[11] S. Li, C. Wei, and Y. Wang, "Combining Decision Making and Trajectory Planning for Lane Changing Using Deep Reinforcement Learning," *IEEE TITS*, vol. 23, no. 9, pp. 16 110–16 136, 2022.

[12] P. Wang, C.-Y. Chan, and A. de La Fortelle, "A reinforcement learning based approach for automated lane change," in *Proc. of IV*, 2018.

[13] F. Ye, X. Cheng, P. Wang, C.-Y. Chan, and J. Zhang, "Automated lane change strategy using proximal policy optimization-based deep reinforcement learning," in *IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 1746–1752.

[14] X. He, H. Yang, Z. Hu, and C. Lv, "Robust Lane Change Decision Making for Autonomous Vehicles: An Observation Adversarial Reinforcement Learning Approach," *IEEE TIV*, vol. 8, no. 1, pp. 184–193, 2022.

[15] P. Y. J. Ha, S. Chen, J. Dong, R. Du, Y. Li, and S. Labi, "Leveraging the capabilities of connected and autonomous vehicles and multi-agent reinforcement learning to mitigate highway bottleneck congestion," *arXiv preprint arXiv:2010.05436*, 2020.

[16] S. Chen, J. Dong, P. Ha, Y. Li, and S. Labi, "Graph neural network and reinforcement learning for multi-agent cooperative control of connected autonomous vehicles," *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 7, pp. 838–857, 2021.

[17] Y. Hou and P. Graf, "Decentralized cooperative lane changing at freeway weaving areas using multi-agent deep reinforcement learning," *arXiv preprint arXiv:2110.08124*, 2021.

[18] J. Zhang, C. Chang, X. Zeng, and L. Li, "Multi-Agent DRL-Based Lane Change With Right-of-Way Collaboration Awareness," *IEEE TITS*, vol. 24, no. 1, pp. 854–869, 2022.

[19] C.-J. Hoel, K. Wolff, and L. Laine, "Automated speed and lane change decision making using deep reinforcement learning," in *Proc. of ITSC*, 2018.

[20] G. E. Uhlenbeck and L. S. Ornstein, "On the theory of the brownian motion," *Physical review*, vol. 36, no. 5, p. 823, 1930.

[21] D. Salles, S. Kaufmann, and H.-C. Reuss, "Extending the intelligent driver model in sumo and verifying the drive off trajectories with aerial measurements," in *SUMO Conference Proceedings*, vol. 1, 2020.

[22] S. Das and A. K. Maurya, "Defining time-to-collision thresholds by the type of lead vehicle in non-lane-based traffic environments," *IEEE TITS*, vol. 21, no. 12, pp. 4972–4982, 2019.

[23] M. Zhu, Y. Wang, Z. Pu, J. Hu, X. Wang, and R. Ke, "Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving," *Transportation Research Part C: Emerging Technologies*, vol. 117, p. 102662, 2020.

[24] I. Jacobson, L. Richards, and A. Kuhlthau, "Models of human comfort in vehicle environments," *Human Factors in Transport Research Edited by DJ Oborne, JA Levis*, vol. 2, no. 431103, pp. 24–32, 1980.

[25] L. C. Das and M. Won, "Saint-acc: Safety-aware intelligent adaptive cruise control for autonomous vehicles using deep reinforcement learning," in *International Conference on Machine Learning*, 2021.

[26] B. Chen, M. Xu, Z. Liu, L. Li, and D. Zhao, "Delay-aware multi-agent reinforcement learning for cooperative and competitive environments," *arXiv preprint arXiv:2005.05441*, 2020.

[27] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *Autonomous Agents and Multiagent Systems*, 2017, pp. 66–83.

[28] J. Foerster, I. A. Assael, N. De Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," *Advances in neural information processing systems*, vol. 29, 2016.

[29] M. Wunder, M. L. Littman, and M. Babes, "Classes of multiagent q-learning dynamics with epsilon-greedy exploration," in *ICML*, 2010.

[30] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *Proc. of ITSC*, 2018.

[31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[32] P. Liashchynskyi and P. Liashchynskyi, "Grid search, random search, genetic algorithm: a big comparison for NAS," *arXiv preprint arXiv:1912.06059*, 2019.

[33] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, p. 1805, 2000.

[34] M. Semrau and J. Erdmann, "Simulation framework for testing ADAS in Chinese traffic situations," *SUMO 2016–Traffic, Mobility, and Logistics*, vol. 30, pp. 103–115, 2016.

[35] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in statistics: Methodology and distribution*, 1992, pp. 492–518.

[36] Y. Du, C. Liu, and Y. Li, "Velocity control strategies to improve automated vehicle driving comfort," *IEEE Intelligent transportation systems magazine*, vol. 10, no. 1, pp. 8–18, 2018.