AN ADAPTIVE LONGITUDINAL DRIVING ASSISTANCE SYSTEM WITH REINFORCEMENT LEARNING

Xiangyu Li
Transportation Center, Northwestern University
600 Foster Street, Evanston, IL 60208, USA
Email: xiangyuli2027@u.northwestern.edu

Pengyuan Liu
Department of Civil and Environmental Engineering, Northwestern University
2145 Sheridan Road, Evanston, IL 60208, USA
Email: pengyuanliu2024@u.northwestern.edu

Hani S. Mahmassani Ph.D., Corresponding Author
William A. Patterson Distinguished Chair in Transportation
Transportation Center, Northwestern University
600 Foster Street, Evanston, IL 60208, USA
Email: masmah@northwestern.edu

Ying Chen Ph.D.
Transportation Center, Northwestern University
600 Foster Street, Evanston, IL 60208, USA
Email: y-chen@northwestern.edu

Word count: 7,240 words text + 1 table(s) × 250 words (each) = 7,440 words

This manuscript is being submitted to ACP80 for presentation to the 104th Annual Meeting of the Transportation Research Board, and publication in Transportation Research Record

Submission Date: September 18, 2024

ABSTRACT

This study introduces an Adaptive Longitudinal Driving Assistance System (ALDAS) based on Reinforcement Learning (RL) to enhance autonomous driving performance and accommodate the expected car-following distance of different drivers. The proposed system employs the Deep Deterministic Policy Gradient (DDPG) algorithm to optimize car-following behavior, which integrates a state space comprising speed, acceleration, distance between vehicles, and traffic conditions, with a continuous action space focusing on real-time acceleration adjustments. The reward function balances safety, efficiency, comfort, and fuel consumption, ensuring an overall improvement in driving quality. The results demonstrate substantial performance improvements of the DDPG model over long short-term memory (LSTM), recurrent neural network (RNN) and cooperative adaptive cruise control (CACC) model when tested under varying traffic flows and penetration rate of autonomous vehicles (AVs). By integrating the desired following distances of various drivers into the DDPG car-following model and analyzing changes in the human take-over rate, this study demonstrates the model's ability to effectively capture human car-following behavior. Additionally, human-machine co-driving outperforms human driving in critical aspects of the simulation experiment. This study is the first to utilize a cyber-physical system (CPS) combined with driving simulator, Simulation of Urban Mobility (SUMO) and Carla to train an autonomous driving system that better aligns with human preferences. The work makes a substantial contribution to improving driver acceptance of autonomous driving systems and provides guidance for the design of future autonomous driving modes.

Keywords: Driving Assistance System, DDPG, Autonomous Driving, Car-following Behavior, Cyber-physical system, Human-Machine Co-Driving

INTRODUCTION

The advancement of autonomous driving technology has garnered significant attention in recent years, driven by its potential to revolutionize transportation systems through enhancing safety, efficiency, and user comfort. Central to this evolution is the development of sophisticated driving assistance systems that can intelligently manage vehicle behavior in different traffic environments. Traditional approaches to car-following models, such as the Krauss model (*1*) and CACC (*2*), have laid the groundwork for these systems. However, they often fall short in accommodating the varying preferences and driving styles of different individuals, thereby limiting their effectiveness in real-world applications.

Recent developments in artificial intelligence, particularly in the realm of Reinforcement Learning (RL), have opened new avenues for addressing these limitations. RL algorithms are well-suited for developing adaptive systems that can learn and optimize behavior based on continuous feedback from the environment. This study introduces an innovative framework, the Adaptive Longitudinal Driving Assistance System (ALDAS), which leverages the DDPG algorithm (*3*) and is grounded in a comprehensive state space that includes parameters such as speed, acceleration, inter-vehicle distance, and prevailing traffic conditions. This state space is paired with a continuous action space that facilitates real-time acceleration adjustments, enabling the system to respond swiftly to changing driving conditions. The reward function integrated within ALDAS is designed to balance the goals of safety, efficiency, comfort, and fuel consumption, thereby ensuring an overall enhancement in driving quality. Additionally, the desired following distance of driver is integrated into the system during the training process.

Empirical evaluations of the DDPG car-following model in ALDAS, conducted under various traffic flows and with different penetration rates of AVs in SUMO (*4*) and Carla (*5*), have demonstrated significant performance improvements over other car-following models in terms of average speed, time loss and waiting time. Additionally, desired following distance of various drivers are analyzed, relationship between desired following distance and driver statistics are illustrated. Furthermore, the decreasing human take-over rate across simulation iterations indicates the capability of the proposed method to capture human driving behaviors. Finally, the superiority of the proposed method is shown through comparison with human driving scenarios.

LITERATURE REVIEW

Safety improvement

Autonomous driving systems have the potential to enhance road safety by using advanced sensors, real-time data processing, and decision-making algorithms to detect and respond to hazards more effectively than human drivers. This review covers four areas: the impact of CAV penetration rates, intersection and ramp safety, public perception and pedestrian safety, specific safety concerns.

Papadoulis et al. (*6*) investigate the effect of CAVs on motorway safety using traffic microsimulation. Finding suggests increased CAV penetration significantly reduces traffic conflicts, with a 12–47% decrease at 25% penetration and a 90–94% at 100% penetration, demonstrating substantial safety benefits. Similarly, Guériau and Dusparic et al. (*7*) examine how varying CAV penetration rates impact road safety and traffic efficiency. They observe that lower penetration rates increase traffic conflicts due to interactions between human-driven vehicles (HDVs) and CAVs. The optimal CAV penetration rates, which typically fall between 20% to 40%, vary depending on the network type and traffic conditions.

Niroumand et al. (*8*) introduce the 'white phase' in traffic signals, allowing CAVs to guide connected human-driven vehicles (CHVs) through intersections safely, reducing speed variance and rear-end near-collision observations as CAV penetration increases. Using VISSIM traffic microsimulation and the Surrogate Safety Assessment Model (SSAM), Morando et al. (*9*) find that AVs reduce conflicts by 20% to 65% at intersections and by 29% to 64% at roundabouts with high AV penetration. Yang et al. (*10*) show that CAVs significantly improve safety at on-ramp scenarios by reducing time-to-collision metrics, with safety improvements becoming more notable at higher CAV penetration rates.

Brar and Caulfield et al. (*11*) conduct an online survey revealing mixed public perception and concerns regarding AVs: optimism about safety benefits balanced by concerns over technology failures and interactions with non-autonomous elements, highlighting the need for extensive real-world testing to ensure pedestrian safety. Khashayarfard and Nassiri et al. (*12*) demonstrate that AVs can significantly reduce potential traffic accidents, with up to 93% decrease in accidents at 100% AV penetration, especially improving safety at interactions and addressing issues like distracted driving.

Enhanced driving quality

By optimizing routes, maintaining consistent speeds, and minimizing abrupt maneuvers, AVs can improve passenger comfort, ensure smoother traffic flow, and reduce emissions.

Sever et al. (*13*) develop a controller that optimally adjusts vehicle dynamics to reduce motion sickness by integrating a mathematical model of the human vestibular system. Simulations show significant reductions in perceived acceleration, jerk, and motions that cause carsickness, enhancing passenger comfort without compromising safety or path-following accuracy. Qin et al. (*14*) find that the vehicle-to-vehicle (V2V) communications lead to smoother traffic flows, reducing speed and acceleration fluctuations, resulting in a more comfortable ride, especially at higher CAV penetration rates under high-density traffic conditions.

Niroumand et al. (*8*) show that AVs improve traffic flow and reduce congestion. Their study indicated that aggressive driving behaviors reduce delays and smooth traffic. They also suggest that a 'white phase' at intersections, which adjusts timing based on real-time conditions, reduces waiting times and performs better at higher AV penetration rates. Talebpour and Mahmassani (*15*) demonstrate that CAVs make traffic flow more stable and increase throughput, preventing traffic disruptions and leading to smoother, more predictable driving with fewer stops and less variation in speed. Guériau and Dusparic (*7*) show that higher CAV penetration rates significantly reduce congestion, especially on motorways and major roadways. The continuous flow of CAVs minimizes stops and starts, leading to more stable and efficient traffic flow, increased speeds, and reduced travel times across various road types.

Stogios et al. (*16*) find that AVs, particularly those with aggressive driving programs, improve traffic flow on highways and reduce greenhouse gas emissions by up to 26% on expressways. Makridis et al. (*17*) show that V2V communications in CAVs further reduce emissions during peak and off-peak hours by smoothing traffic flows and reducing stop-and-go situations. Their study also finds that CAVs are more effective at reducing emissions compared to AVs without connectivity.

Human factors

This section highlights the critical role of human factors in enhancing the performance, safety, and effectiveness of AVs and CAVs. Considering human factors in system design, driving behavior, and control transitions is essential for developing algorithms that effectively complement human drivers and address their cognitive capabilities and limitations.

Cárdenas et al. (*18*) identify key human factors, such as age, focus, multitasking capabilities, intelligence, and learning speed, that affect performance and safety in Level 3 automation, where drivers must take over in certain situations. Their study applies quality function deployment to ensure autonomous driving systems user needs and consider human factors. Sarker et al. (*19*) discuss how integrating human factors into CAV design can improve performance. They highlight the importance of responsive systems that adapt to user feedback in real-time and the role of human factors in enhancing Vehicle-to-Everything (V2X) communications.

Liu and Zhang (*20*) explore how human cognitive factors impact driver performance during critical transitions in autonomous driving. They find that emotional states and the environment affect a driver's ability to take control from the autonomous system. They emphasize the need for clear and varied takeover cues (visual, auditory, tactile) to keep drivers aware and ready without overwhelming them. Li et al. (*21*) investigate the use of Model Predictive Control (MPC) to optimize the switch between human driving and autonomous control. Their MPC-based strategy adjusted control dynamically, incorporating human reaction times, distance perception, and adaptive cruise control (ACC) system. The approach effectively mimics human driving behavior. Raiyn and Weidl (*22*) delve into integrating human factors into autonomous driving algorithms to improve predictions and manage driving behavior in safety-critical scenarios. Their method, which accounts for factors like age and driving experience, reduce predictive errors by 20-30% and achieves up to 85% accuracy. Personalized cognitive agents tailored to individual driver profiles, also improve system response rates by about 15%.

Xin et al. (*23*) investigate the integration of human factors in developing autonomous driving technologies, emphasizing human-vehicle collaboration. They highlight trust and situational awareness as crucial factors for successful interaction. The study also address challenges like ensuring vehicle actions are predictable and understandable to human drivers and calls for more research into adaptive systems that can dynamically adjust to individual drivers' needs and behaviors.

Reinforcement learning

Some autonomous driving algorithms utilize RL techniques to improve performance and adaptability. RL allows vehicles to learn from experience by interacting with their environment, making decisions, and receiving feedback. These studies cover deep reinforcement learning frameworks, multi-agent systems, lateral control systems, and probabilistic guarantees and safety.

Sallab et al. (*24*) present a deep RL framework incorporating attention models and recurrent neural networks to handle partial information, showing effective learning and adaptation with reduced computational load. Palanisamy et al. (*25*) discuss the complexities of modeling CAV systems as multi-agent environments using Partially Observable Markov Games (POMG), improving navigation in urban intersections based on learned policies.

Bouton et al. (*26*) integrate probabilistic guarantees into reinforcement learning for unsignalized intersections using Linear Temporal Logic (LTL), ensuring safety by avoiding collisions. Cao et al. (*27*) present Trustworthy Improvement Reinforcement Learning (TiRL), which consistently outperforms rule-based policies in simulation driving tests.

Li et al. (*28*) divide the lateral control system into two parts: a perception module that predicts track features using a neural network, and a control module that uses these features for steering decisions via RL. Results show the RL controller outperforms traditional methods on various tracks. Zhu et al. (*29*) enhance car-following velocity control using a DDPG algorithm, achieving better collision times, smoother acceleration, and more balanced headway compared to human drivers and adaptive cruise control systems.

Osiński et al. (*30*) employ RL to develop driving policies with synthetic and limited real-world data. Their policies, tested on real vehicles in various scenarios, show successful real-world applications and lower developmental costs by using simulated data. Cao et al. (*27*) use TiRL to ensure RL policies outperform integrated rule-based policies. TiRL demonstrated superior performance in extensive simulations, highlighting its potential for real-world application.

Cyber-physical System

Cyber-physical systems (CPS) are increasingly integrated into AVs to tackle challenges like traffic flow modeling, path tracking control, and cyber-attack resilience.

Chen et al. (*31*) propose using in-vehicle CPS sensors to improve traffic flow modeling for better route planning and driving decisions. Their model optimizes speed, braking, and safe distances while assisting neighboring AVs through Ad-Hoc networks or cloud systems. The result show accurate traffic signal influence and real-time adjustments based on velocity, density, and location. Zhang et al. (*32*) present a CPS-based approach for improving cruise control in CAVs. By combining real-time and historical data, their approach improves behavior prediction and safe following distances, resulting in better vehicle safety and comfort in complex driving scenarios.

Guo et al. (*33*) introduce a CPS-based approach to strengthen the robustness of AVs' path-tracking control against cyber-attacks. Their model addresses Denial of Service (DOSA) and False Data Injection Attacks (FDIA) using nonlinear equations, sensor redundancy, and cyber-attach detection with extended Kalman filters. A Model Predictive Control (MPC) framework maintains stability, accuracy and effectiveness in simulations.

Ding (*34*) proposes a framework to enhance AV resilience against cyber-attacks using nonlinear state, measurement equations and sensor redundancy. Extended Kalman filters and sensor switching rules help isolate disturbances while MPC ensures stability under attack conditions. Simulations validate minimal lateral and heading errors. Chattopadhyay and Lam (*35*) analyze the security challenges in AVs modeled as CPS. They emphasize increased security risks at higher autonomy levels due to interconnected systems and external communication.

METHODOLOGY

CACC Car-following Model

The car-following model optimizes the driving process of the following vehicle by determining the optimal acceleration, deceleration rate, and following distance. The CACC model is designed to enhance traffic flow and safety through V2V communication. VanderWerf et al. (*2*) propose a CACC car-following model based on a constant desired headway distance. The model structure is as follows:

$$\dot{v}_i(t) = k_1 \dot{v}_{i+1}(t) + k_2 \left[x_{i+1}(t) - x_i(t) - t_g v_i(t) - l - s_0 \right] + k_3 \left[v_{i+1}(t) - v_i(t) \right] \quad (4)$$

In this equation, $\dot{v}_i(t)$ and $\dot{v}_{i+1}(t)$ are the acceleration of the following and leading vehicle respectively. $v_i(t)$ represents speed, and $x_i(t)$ represents horizontal location. s_0 is the safe stop distance, t_g is the CACC desired headway time, and k_1, k_2, k_3 are coefficients. The first term of the equation is the response term to the acceleration of the leading vehicle, the second term is the error term between the actual headway distance $x_{i+1}(t)-x_i(t)$ and the desired headway distance $t_g v_i(t)+l+s_0$, and the third term is the relative speed.

This proposed CACC model is a widely used car-following model based on a constant desired headway time. The CACC model implemented in SUMO is developed based on four papers (*36-39*). It operates in four modes:

- Speed Control Mode: the vehicle maintains the driver's desired speed by adjusting its speed.
- Gap Control Mode: the vehicle maintains a fixed headway distance from the leading vehicle.
- Gap-Closing Control Mode: ensures a smooth transition between Speed Control Mode and Gap Control Mode.
- Collision Avoidance Control Mode: the vehicle decelerates at its maximum rate to prevent rear-end collisions.

The car-following model discussed provides a digital representative of the car-following system, focusing on maintaining a safe distance and optimizing following speed. However, in real mixed-traffic environments, traditional models must also consider factors like driving efficiency, fuel consumption, comfort, etc. To address these needs, this paper introduces a car-following model based on the DDPG algorithm. This model optimizes car-following behavior from multiple perspectives and adapts to complex driving conditions by incorporating human driving behaviors, making it more compatible with real-world scenarios and acceptable to human drivers.

DDPG Car-following Model

FIGURE 1 shows the flow chart of DDPG algorithm (*3*). It begins by initializing the Critic evaluation network $Q(s,a|\theta^Q)$ and Actor policy network structures $\mu(s|\theta^\mu)$, along with their weights θ^Q and θ^μ. Next, it initializes the Critic target evaluation network $Q'(s',a'|\theta^{Q'})$ and Actor target policy network structures $\mu'(s'|\theta^{\mu'})$, along with their weights $\theta^{Q'}$ and $\theta^{\mu'}$. The final initialization step involves setting up the replay experience pool. The Output of DDPG algorithm includes Critic evaluation network parameters θ^Q, Actor policy network parameters θ^μ, Critic target evaluation network parameters $\theta^{Q'}$, and Actor target policy network parameters $\theta^{\mu'}$.

As shown in **Figure 3**, the DDPG car-following model updates acceleration actions iteratively and learns from the environment through experience replay. This approach allows the vehicle to adapt efficiently to varying traffic conditions.

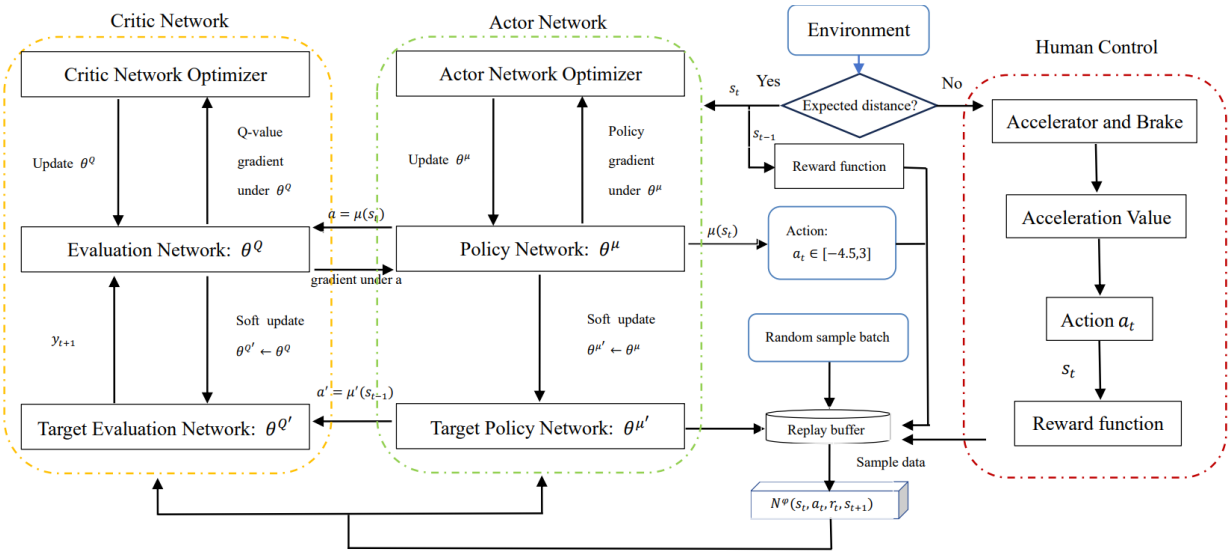

FIGURE 1 Flow chart of DDPG algorithm

Algorithm 1 DDPG Car-Following Model

for episode from 1 to M **do**
  1) Initialize a random process K for action exploration.
  2) Initialize s_1 as the first state of the current state sequence and obtain its feature vector $\phi(S)$.
  for t from 1 to T **do**
    if expected following distance met
      a) Select a car-following acceleration action $a_t = \mu(s_t|\theta^\mu) + K_t$ based on the current policy and exploration noise.
      b) The vehicle executes action a_t, observes reward r_t and new state s_{t+1}.
    else
      c) The driver takes control and executes action a_t, observes reward r_t and new state s_{t+1}.
    end if
    d) Store the transition (s_t, a_t, r_t, s_{t+1}) in the replay buffer.
    e) Sample a random minibatch of N transitions (s_t, a_t, r_t, s_{t+1}) from the replay buffer.
    f) Critic evaluation network calculates the current Q-value:

$$y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}).$$

    g) Update the Critic evaluation network parameters by minimizing the loss:

$$L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^Q))^2.$$

    h) Update the Actor policy network gradient using the sampled policy gradient:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N}\sum_i \left(\nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s_i}\right).$$

    i) Update the parameters of the Critic target evaluation network and Actor target policy network:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'},$$
$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}.$$

  end for
end for

FIGURE 2 DDPG algorithm

State space

In the practical training of DDPG car-following model, each AV acts as an individual agent using the algorithm framework outlined in FIGURE 2. This framework includes two pairs of Advantage Actor-Critic (A2C) networks that update weights mutually. The Actor policy network

receives input from the state space defined for the car-following model, and its continuous action output determines the following vehicle's acceleration. The state space is divided into the following components:

- The current following speed of the agent vehicle $V_f(t)$
- The current acceleration or deceleration of the agent vehicle $a_f(t)$
- The current following distance of the agent vehicle $D_f(t)$
- The relative speed between the agent following vehicle and the leading vehicle $\Delta V(t)$
- The current trip time loss for the agent vehicle $T_{loss}(t)$

Thus, the overall state space can be represented as a six-dimensional set:

$$s(t) = \left\{ V_f(t), a_f(t), D_f(t), V_r(t), T_{loss}(t) \right\} \tag{5}$$

The setup of this state space considers both microscopic traffic parameters, such as following speed and following distance, and macroscopic traffic conditions such as the trip time loss. This approach allows the agent's learning process to account for the vehicle's individual following pattern and the overall traffic efficiency of all vehicles.

Action space

Since the DDPG algorithm studies the car-following model of agent vehicles, involving real-time adjustments to following acceleration based on road traffic conditions, this design utilizes the continuous nature of the actions output by the DDPG's Actor policy network. The action space is directly set as the agent vehicle's acceleration/deceleration, and the suggested following distance is calculated based on the following acceleration. This setup allows the agent to maximize traffic efficiency while ensuring a safe following distance. The following acceleration in the action space can be expressed as:

$$a(t) \in [-4.5, 3](m/s^2) \tag{6}$$

And the following distance set by the agent vehicle to ensure safe following is established as:

$$D_f(t) = \begin{cases} |a(t)| \times 15 & 80 \le D_f(t-1) \\ |a(t)| \times 5 & 40 \le D_f(t-1) < 80 \\ |a(t)| \times 2 & 0 \le D_f(t-1) < 40 \end{cases} \tag{7}$$

Reward function

The reward function is designed to provide rewards or penalties to the agent at each timestep, which considers four aspects: safety, efficiency, comfort, and fuel consumption. At each timestep, the agent makes two observations: one at the start of the action (time t) and another at the end of the action (time t+1). The total reward function R_{total} is a weighted sum of these four aspects. The specific definitions of the four reward functions are as follows:

Safety Reward: Time-to-Collision (TTC)

For the safety reward of the car-following model, Time-to-Collision (TTC) is primarily considered, rather than the actual number of collisions. This is because actual collisions in real traffic systems are extremely dangerous, and the agent's decisions must prevent real collisions before they occur. TTC is defined as the time until a collision occurs if the vehicle does not change its speed during the car-following process. A large TTC value indicates that even with acceleration, a collision is unlikely; a small TTC value under a certain threshold (*40*) indicates danger. The definition of TTC (*41*) is expressed as follows:

$$TTC(t)=\begin{cases}\dfrac{X_{i+1}(t)-X_i(t)-l_i}{V_i(t)-V_{i+1}(t)} & \forall V_i(t)>V_{i+1}(t)\\ \infty & \forall V_i(t)\le V_{i+1}(t)\end{cases} \tag{8}$$

Where $X_{i+1}(t)$ is the position of the leading vehicle, $X_i(t)$ is the position of the agent vehicle, and l_i is the length of the agent vehicle, assumed to be 4.5 meters, the typical length of a passenger vehicle. $V_i(t)$ and $V_{i+1}(t)$ are the speeds of the agent and leading vehicle, respectively. Assuming the agent makes two observations at each time step, one at the start of the action (time t), and then another at the end of the action (time $t+1$), the safety reward can be expressed as:

$$R_{safety}(t)=TTC(t+1)-TTC(t)\quad \forall TTC<10 \tag{9}$$

If $TTC(t+1)$ is larger than $TTC(t)$, a reward is given. Conversely, if the TTC decreases, a penalty is given. If there is no change in TTC, no reward or penalty is applied.

Efficiency reward

Efficiency reward is determined by the headway and speed between the leading vehicle and the following vehicle during the car-following process, and it is defined as follows:

$$R_{effi}(t)=0.5\left(V_i(t)-V_{i+1}(t)\right)^2+\left(S_i(t)-S^*\right)^2+0.5a_i(t)^2+0.1\left(\frac{da_i(t)}{dt}\right)^2 \tag{10}$$

Where $(V_i-V_{i+1})^2$ represents the difference in speed between two vehicles, S_i is the actual spacing between the following vehicle and the leading vehicle, S^* is the spacing desired by the driver (determined by collecting data from drivers without any driving assistance), and a_i is the acceleration of the following vehicle. The term $(\frac{da_i}{dt})^2$ penalizes rapid changes in acceleration, i.e., high jerk, ensuring smoother driving by discouraging sudden changes in acceleration.

A common approach to defining the desired spacing S^* is to use a combination of a fixed minimum spacing and a term proportional to the speed of the following vehicle. This paper uses the "intelligent driver model" (IDM) (*42*) to define the desired spacing, S^* can be formulated as follows:

$$S^* = S_0 + V_i T + \frac{V_i (V_i - V_{i+1})}{2\sqrt{ab}} \tag{11}$$

This study uses the following parameters for calculating S^*:

- S_0 =2 meters: Minimum safe distance when stationary to avoid collisions
- T =1.5 seconds: Safe time headway
- a =2 m/s^2: Maximum acceleration of the following vehicle
- b =2 m/s^2: Comfortable deceleration rate (positive value, as deceleration is negative)

The above S^* is only used before training since the purpose of this paper is to allow the algorithm to capture the real behavior of human drivers, the desired spacing S^* will then be updated using empirical data, and the regression relationship with the highest R-square is calculated as:

$$S^* = a \cdot V_i + b \cdot V_{i+1} + c \cdot V_i^2 + d \cdot V_{i+1}{}^2 + e \cdot (V_i \times V_{i+1}) + f \tag{12}$$

Comfort reward

The comfort reward measures the degree of vehicle bumpiness before and after the agent selects an action, focusing on the comfort parameter known as Jerk, which is the rate of change of vehicle acceleration. High Jerk values indicate abrupt braking or rapid acceleration, leading to poor passenger comfort. Therefore, the agent should prioritize driving comfort when selecting actions. The comfort reward value can be calculated as follows:

$$R_{comf}(t) = \frac{jerk_i(t+1)^2}{j_{\max}} - \frac{jerk_i(t)^2}{j_{\max}} \tag{13}$$

$$jerk_i(t) = \frac{a_i(t) - a_i(t-1)}{\Delta t} \tag{14}$$

In the above equations, squaring the jerk makes the reward more sensitive to large changes in jerk. This approach helps to effectively reduce high jerk values, improving passenger comfort. The variable $j_{\max} = 3m/s^3$ represents the maximum allowed jerk value for maintaining comfort (*43*).

Fuel consumption reward

The fuel consumption reward measures how the agent vehicle's fuel consumption rate changes before and after an action, reflecting the impact on energy efficiency. Although the SUMO traffic simulator uses the VT-Micro model for energy calculations, it is not very accurate. Instead, the Hesham energy consumption model is used, with factors in both speed and acceleration. This model estimates instantaneous fuel consumption using a regression-based approach that considers the vehicle's kinematic properties and empirical data. The energy consumption reward (*44*) can be expressed as follows:

$$R_{fuel} = \sum_{t=0}^{t=end} e_i^t (V_i^t, a_i^t) \tag{15}$$

$$e_i^t(V_i^t, a_i^t) = \exp\left[\sum_{j=0}^{3}\sum_{k=0}^{3} K_{jk}(a_i^t)\left(\left\lfloor V_i^t \right\rfloor_0^{120\text{ km/h}}\right)^j \left(\left\lfloor a_i^t \right\rfloor_{\text{-5 km/h/s}}^{13\text{ km/h/s}}\right)^k\right] \quad (16)$$

Where K_{ij} are coefficients derived from empirical data collected under different driving conditions, specific to the vehicle type and the measure of effectiveness (MOE), such as fuel consumption or emissions. These coefficients depend on the vehicle's acceleration. This study uses $[K_{jk}(a_i^t)]_{j,k=0,1,2,3}$ values obtained from test results of eight light-duty vehicles collected by the Oak Ridge National Laboratory (*44*).

The term $\left\lfloor V_i^t \right\rfloor_0^{120\text{ km/h}}$ and $\left\lfloor a_i^t \right\rfloor_{\text{-5 km/h/s}}^{13\text{ km/h/s}}$ represents bounds for the speed and acceleration values in the model, ensuring that it operates within realistic and safe limits. The indices j and k, ranging from 0 to 3, are used to denote polynomial terms in the regression equation, indicating the model includes terms up to the third degree for both speed and acceleration.

In summary, the total reward function for the DDPG car-following model algorithm can be expressed as:

$$R_{total} = w_{safety} \times R_{safety} + w_{effi} \times R_{effi} + w_{comf} \times R_{comf} + w_{fuel} \times R_{fuel} \quad (17)$$

Where the weights of $w_{safety}, w_{effi}, w_{comf}, w_{fuel}$ are 0.5, 0.3, 0.1, 0.1 respectively.

EXPERIMENT

Simulation Environment

To conduct driving simulation experiments and train autonomous driving models that align with various drivers' car-following habits, the framework of the driving simulation and autonomous driving training system is illustrated in **FIGURE 3** Cyber-physical System for Autonomous Driving Simulatio.

The software component includes a combined virtual simulation environment using SUMO and Carla. SUMO (*4*) is an open-source traffic simulation that handles large road networks and detailed traffic systems, including vehicle movements, traffic lights, and public transportation. Carla (*5*) is an open-source autonomous driving simulator offering high-fidelity urban environments and realistic vehicle dynamics. Integrating SUMO's traffic modeling with Carla's vehicle simulation enables accurate simulation of complex urban driving scenarios. Town 4 built into Carla is used for its diverse road types, making it ideal for studying driver-following distances at various speeds.

The hardware component features a driving simulator that mimics a real vehicle, equipped with a steering wheel, accelerator, brake, clutch, and gear shifter. This setup, combined with SUMO and Carla, allows real-time monitoring of vehicle status and data collection on driving habits, acceleration, braking frequency, and shifting frequency. As shown in **FIGURE 3**, the button closest to the driver's right hand on the steering wheel is designated for transferring vehicle control, allowing the driver to switch between manual and autonomous driving modes as needed.

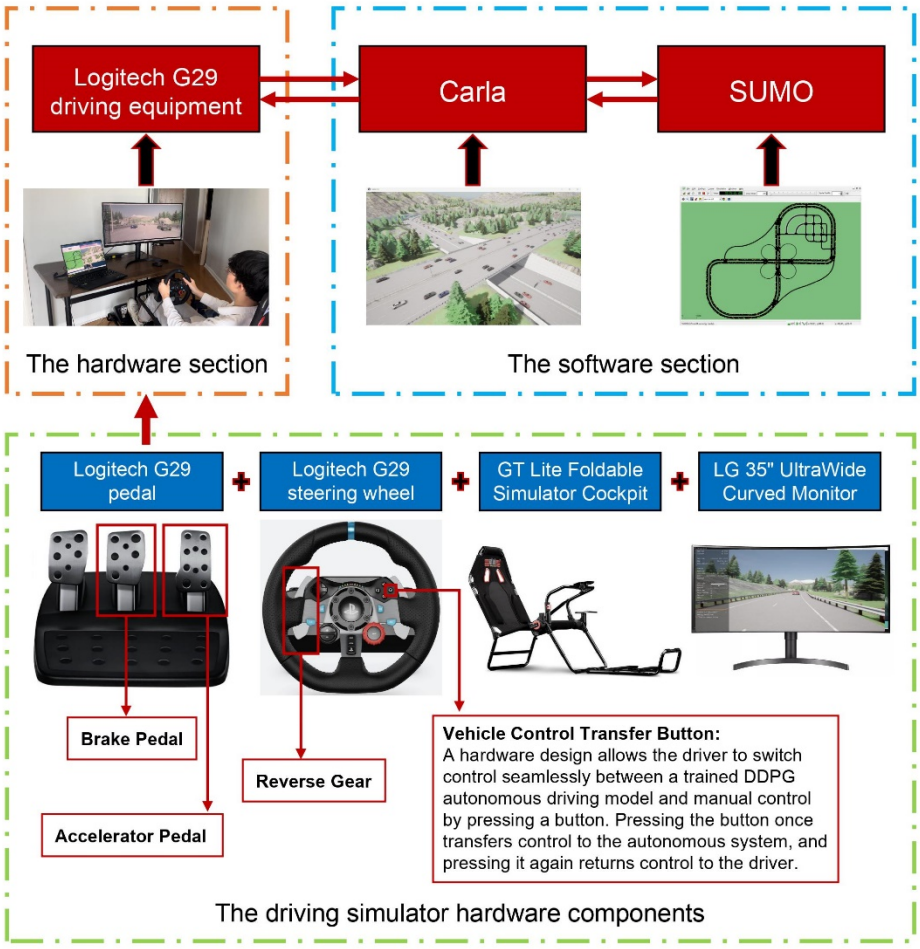

FIGURE 3 Cyber-physical System for Autonomous Driving Simulation

Experiment Procedure

Before the experiment, participants are asked to drive without driving assistance system, desired following distance are then identified using collected data. In the first simulation, a basic autonomous driving model with the driver's desired following distance is trained and a leading vehicle to follow is assigned. As shown in **FIGURE 4,** participants are instructed that if the following distance in autonomous mode does not meet their expectations (e.g., too close or too far), they should switch to manual control to adjust the following distance. In this case, the DDPG output will be replaced by the driver's control inputs, i.e., acceleration rate. The data collected throughout the driving process will be used to fit a new curve representing the driver's desired following distance and to train and update the autonomous driving model for the next simulation. By conducting multiple training sessions and comparing the vehicle takeover rates in autonomous mode, this system can evaluate the efficiency of the DDPG algorithm in learning different drivers' following behavior. The simulation configuration is shown in **TABLE 1**. To compare the performance fairly, besides the CACC model, this study also implements two learning-based car-following models, a RNN based car-following model (*45*) and a LSTM based car-following model (*46*).

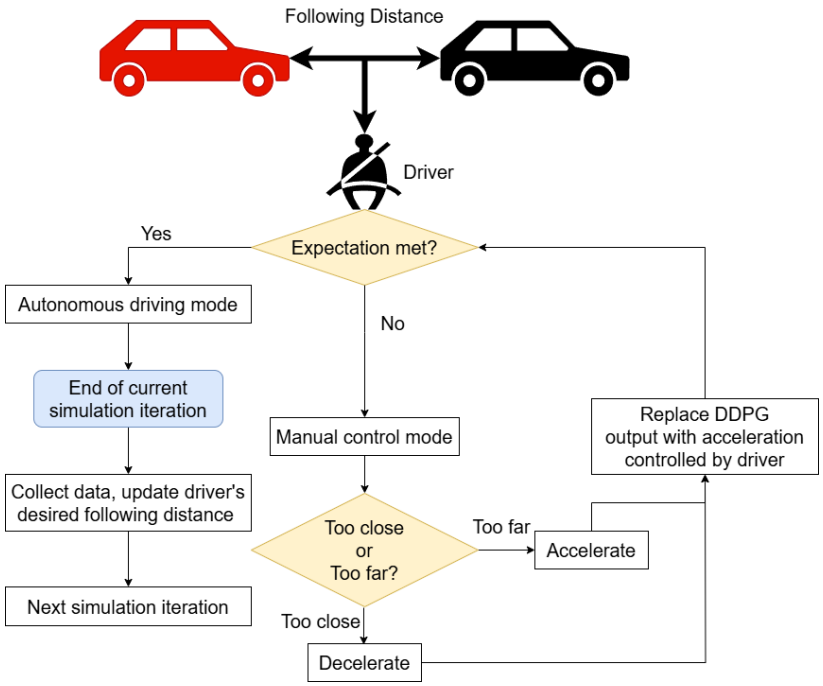

FIGURE 4 Experiment procedure

RESULT

DDPG Car-Following Model Performance

FIGURE 5 shows the overall trend of the average cumulative reward. Initially, the reward increases significantly around 260 epochs. Between 600 and 1000 epochs, the reward fluctuates and declines. Afterward, it gradually rises and stabilizes, converging to around 53 at 2100 epochs. This indicates that the DDPG agent has effectively learned the car-following model.

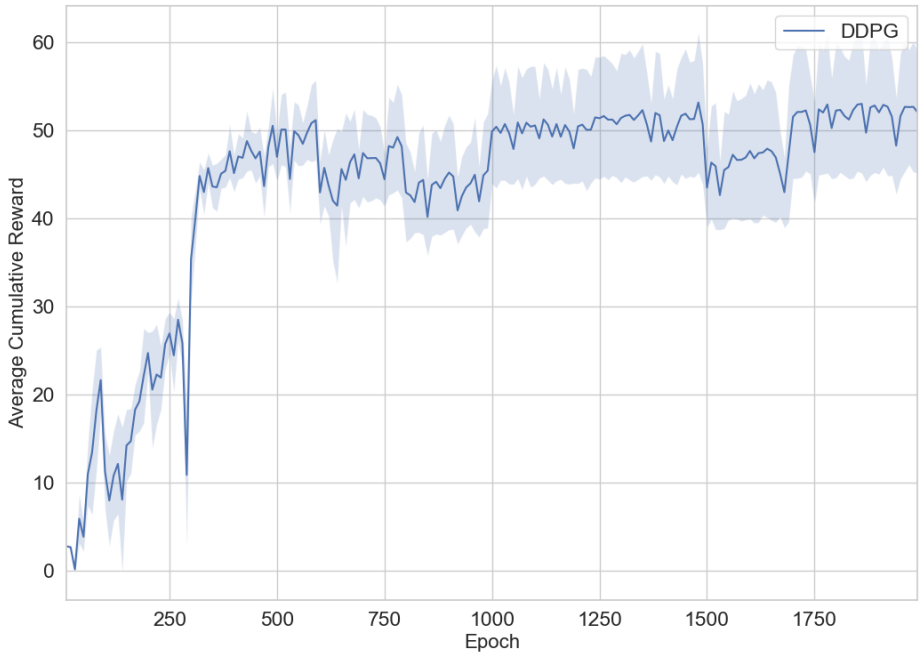

FIGURE 5 Average cumulative reward for the DDPG car-following model

TABLE 1 Simulation Configuration

No.	Parameter	Setting	Unit
0	Traffic Simulation Platform	SUMO, Carla	-
1	Simulation Duration	3600	seconds
2	Traffic Scenario	Carla Town 4 Map	-
3	Average Trip Length	10	km
4	Traffic Flow Settings	1000, 1500, 2000, 2500, 3000	vehicle/h
5	Maximum Vehicle Speed	90	km/h
6	Benchmark Car-Following Models	CACC, RNN, LSTM	-
7	Vehicle Lane-Change Model	LC2013 in SUMO	-
8	AV Penetration Rate	0,25,50,75,100	%
9	Human-Driven Car-Following Model	Krauss	-
10	Autonomous Car-Following Model	DDPG	-
11	DDPG Framework	2 Actor-Critic Networks	-
12	Actor/Critic Network Layers	5	layers
13	Actor/Critic Network Architecture	Input, 3 Hidden, Output Layer	-
14	Number of Neurons in Neural Network	6, 128, 256, 128, 1	units
15	Activation Function	ReLU	-
16	Optimizer	Adam	-
17	Learning Rate	0.01	-
18	Training Epochs	2000	epochs
19	Operating System	Windows 11	-
20	CPU	Intel Core i9-13900H CPU	-
21	GPU	NVIDIA RTX 4080	-
22	RAM	16 GB DDR4	-
23	SSD	1 TB	-
24	Training Environment	TensorFlow, Keras	-
Participant Statistics			
25	Participant Driver	13	person
26	Age	23-57	years
27	Sex	7 males, 6 females	years
28	Driving Experience	1-24	-
29	Accidents Involved	0-5	-
30	Use of Driver Assistance System	6 yes, 7 no	-
31	Aggressive Driving Behavior Index (*47*)	1.9-3.55	-

The following analysis compares the performance of the DDPG car-following model in comparison to the LSTM, RNN and CACC models under various traffic flow conditions and AV penetration rates. 错误!未找到引用源。 shows that at 75% AV penetration rate, the DDPG model consistently achieves the highest average speeds across all traffic flows. For example, under 3000 vehicles per hour, it results in a 41% increase in average speed compared to the CACC model. Additionally, the DDPG model has the lowest average travel and waiting times, across all traffic conditions, demonstrating its efficiency in optimizing routes and reducing delays.

FIGURE 6 compares the models across different AV penetration rates with 2000 vehicles per hour. The DDPG model demonstrates superior performance in average speed, which increases

with higher AV penetration rate, reaching the highest at 100% AVs. At 100% AVs, it outperforms the LSTM, RNN, and CACC models by 19%, 30% and 38%, respectively. The DDPG model also maintains the lowest average waiting times, reducing them by 9%, 12% and 22% compared to the LSTM, RNN and CACC models, respectively. These results highlight DDPG model's effectiveness in integrating AVs into traffic systems for optimal performance.

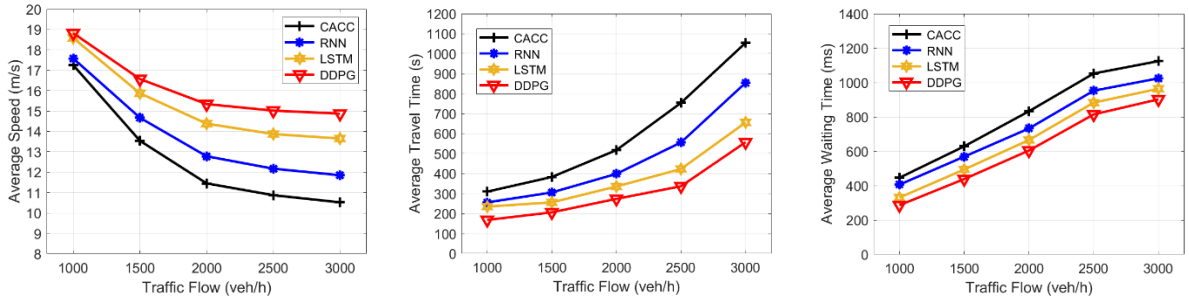

FIGURE 7 Performance of the DDPG car-following model compared to the LSTM, RNN, and CACC models under different traffic flows

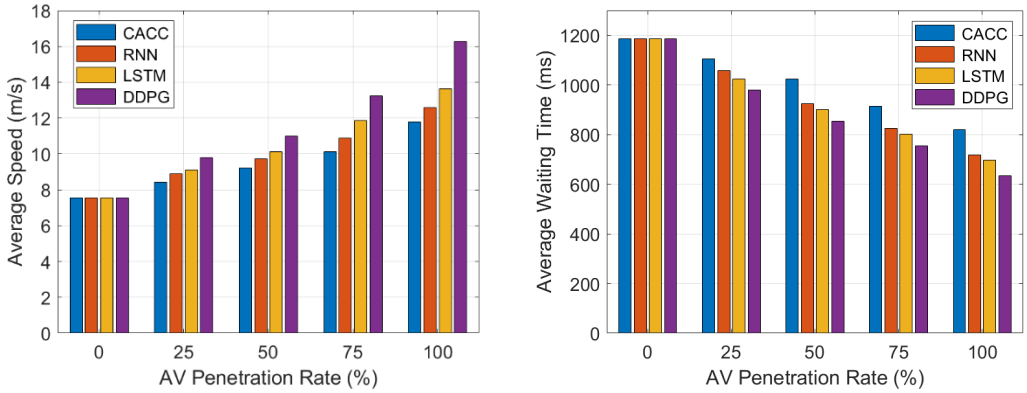

FIGURE 6 Performance of the DDPG car-following model compared to the CACC, RNN and LSTM models under different AV penetration rates

Desired Following Distance

This section examines desired following distance curves based on human drivers' expectations. Using the Ordinary Least Squares (OLS) method, curves (in **Equation 18**) are fitted from drivers' data collected their driving sessions.

$$S^* = a \cdot V_i + b \cdot V_{i+1} + c \cdot V_i^2 + d \cdot V_{i+1}^{\ 2} + e \cdot (V_i \times V_{i+1}) + f \tag{18}$$

FIGURE 7 visualizes the desired following distance of 13 drivers when the leading vehicle 's speed is 10 m/s. Generally, when the following vehicle's speed is less than the leading vehicle's speed is less than 10 m/s, drivers typically accelerate to close the gap. As the speed increases, drivers maintain a smaller following distance but increase it again when their speed surpasses the leading vehicles' speed to avoid collisions. The curves vary, showing different driving behaviors, with some drivers preferring smaller gaps and other larger ones

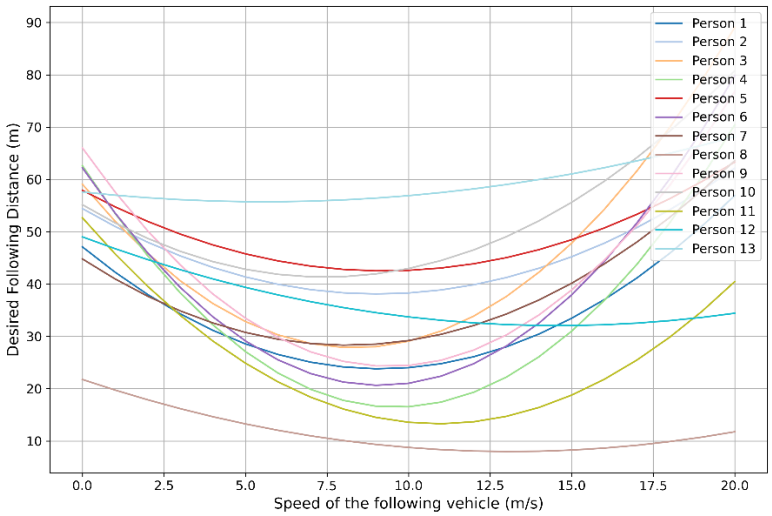

FIGURE 7 Desired following distance of different drivers with a leading vehicle speed of 10 m/s

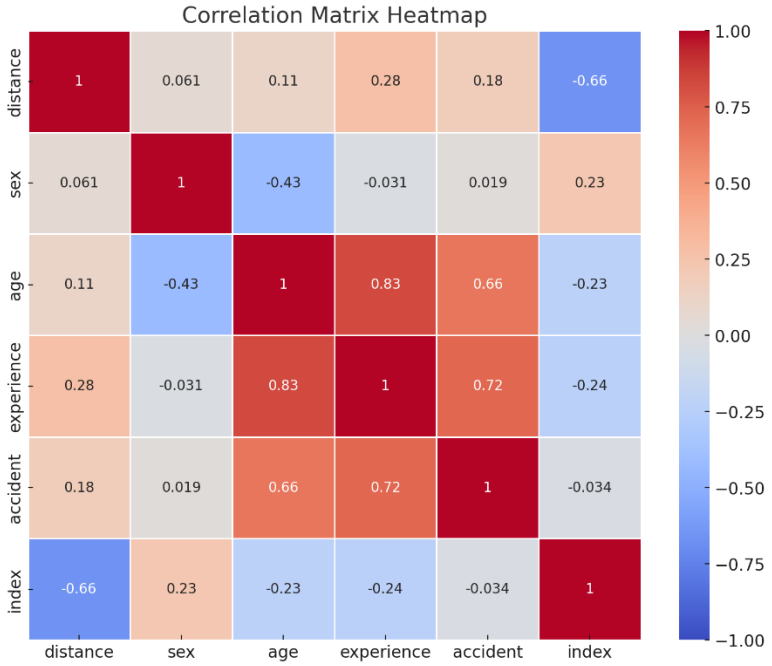

FIGURE 8 Correlation heatmap between desired following distance and various driver statistics

FIGURE 8 reveals a strong negative correlation (-0.66) between desired following distance and aggressive driving behavior. Higher aggressive driving indices results in smaller desired following distance. Age, driving experience, and the number of accidents positively impact correlate with larger following distances, highlighting the complex nature of car-following behavior.

Human Take-Over Rates

Following the desired following distances established earlier, the autonomous driving mode was updated to match driver's preferences for the next simulation. Participant then experienced this updated mode, with the option to take control if they felt the following distance did not meet their expectations. The human take-over rate, indicating the fraction of time that the system was manually controlled, was recorded from this point.

After each simulation, the mode was updated based on new data, and participants repeated the process for iterations (approximately 50 minutes total). **Figure 11** shows a decreasing trend in human take-over rates across iterations 1, 3 and 5. This indicates that the algorithm effectively captures everyone's driving behaviors, helping the autonomous driving mode progressively aligns with individual driving preferences.

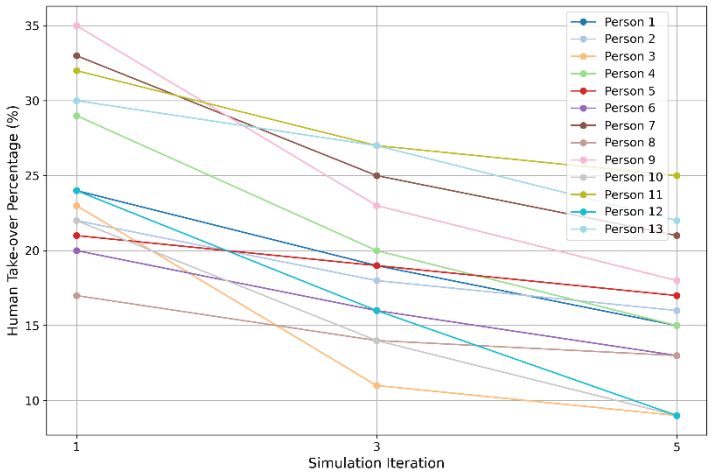

FIGURE 9 Human Take-over Rate in iteration 1, 3 and 5

FIGURE 10 breaks down the take-over rate changes by participants' experience with driver assistance systems. Experienced participants (red) in driver assistance systems (e.g., adaptive cruise control, navigation on autopilot (NOA)) began with a lower take-over and it decreased faster and stabilized lower compared to inexperienced participants (blue). For participants without such experience, in subsequent interviews, they expressed initial distrust in the autonomous driving mode, but this distrust diminished over iterations.

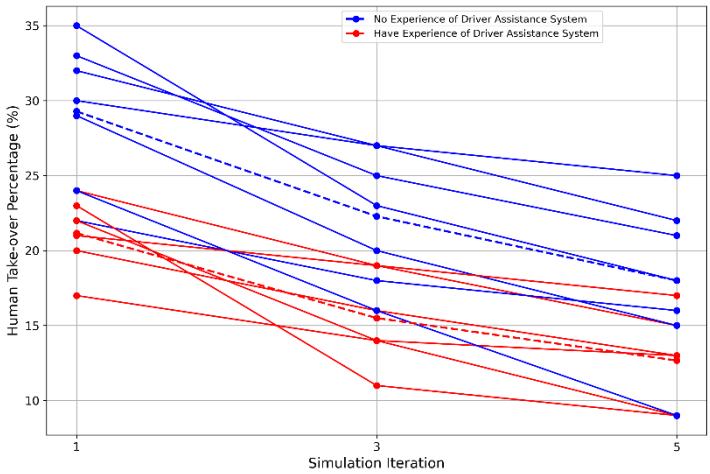

FIGURE 10 Human take-over rate in iteration 1, 3 and 5 differentiated by experience with driver assistance systems

DDPG-Autonomous Driving System

After five simulation iterations, the data from the 5th simulation iteration is compared with human driving data to evaluate the DDPG-autonomous driving system. Key aspects considered are:

- Safety: measured by Time to Collision (TTC), which represents the time remaining before a collision would occur if the current speeds and trajectories are maintained
- System Efficiency: assessed through average speed, time loss, and waiting time
- Comfort: evaluated using the absolute jerk value (rate of change of acceleration rate)
- Energy consumption: measured by the fuel consumption per second per vehicle

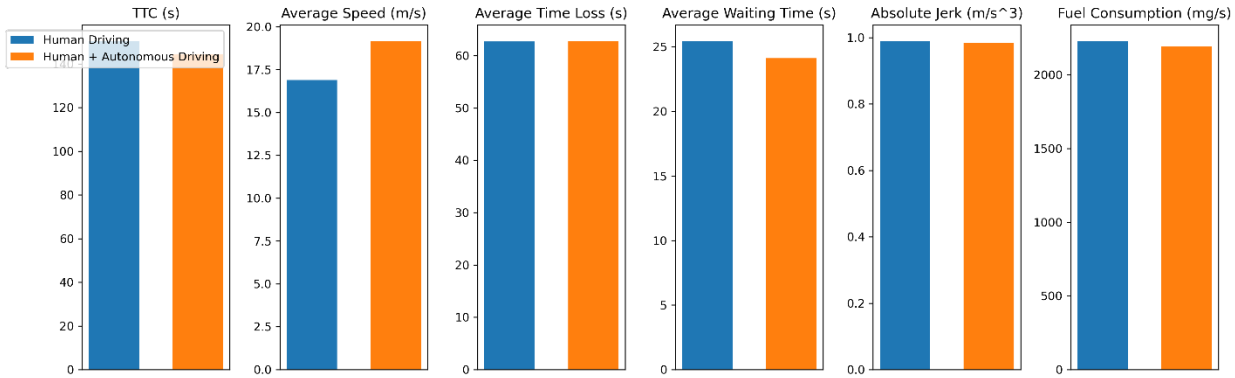

FIGURE 11 Comparison of DDPG-autonomous driving system with human driving

The proposed DDPG system results in lower average TTC, indicating maintained safety, as values were above 140 seconds within a safe threshold (*40*). It also improves average speed of the system by 3m/s, reduces both time loss and waiting time, making the transportation system more efficient. Additionally, the absolute jerk value is slightly lower than human driving model, indicating that the model provided a smoother ride with fewer abrupt changes in acceleration. It

also results in marginally better fuel efficiency, contributing to lower energy consumption and an environmentally friendly transportation system. By examining these aspects, this section assessed the effectiveness of the DDPG-autonomous driving system in replicating and potentially enhancing human driving behavior across critical performance metrics.

CONCLUSIONS AND FUTURE WORK

This study develops and evaluates a DDPG-based ALDAS designed to capture and enhance human car-following behaviors. The model is rigorously evaluated under various traffic flow conditions and different AV penetration rates, and compared with LSTM, RNN, and CACC models. To achieve this, the desired following distances for various drivers are integrated into the DDPG car-following model. Human take-over rates are used to assess the model's capability in effectively capturing human car-following behavior.

The findings reveal that compared to LSTM, RNN, and CACC models, the DDPG model consistently achieves higher average speed, lower time loss and waiting time, highlighting its capability to maintain efficient traffic flow and minimize congestion as traffic density increases. Further analysis shows that the DDPG model effectively captures individual driving preferences and behaviors. This is evidenced by the decrease in human take-over rates over successive simulation iterations, particularly among participants having experience with driver assistance systems. Compared to human driving, the human-machine co-driving system ensures safety by maintaining TTC values above the safe threshold, while also improving system efficiency and providing smoother rides with less fuel consumption.

By enhancing safety, building trust in autonomous systems, and capturing individual driving preference to provide a more personalized driving experience, these factors are crucial for widespread acceptance and adoption of autonomous driving vehicles. Considering the reduced fuel consumption provided by the human-machine co-driving system too, autonomous driving offers an attractive option for eco-conscious consumers.

This research bridges technological advancements with human-centered perspectives in autonomous driving, offering valuable insights for the design of future autonomous driving systems. Future work will involve expanding the driver base for system testing and collecting extensive data for training. Additionally, different driving test scenarios, including varied weather conditions, will be set up to enhance the system's applicability across diverse traffic environments. Furthermore, additional driving behaviors, such as lane changing and turning, will be considered to capture a broader range of driving habits and improve the system's adaptability throughout the entire driving process.

AUTHOR CONTRIBUTIONS

All authors contributed to development methodology, experiment design, interpretation of results, and manuscript preparation. All authors reviewed the results and approved the submission of the manuscript.

REFERENCES

[1] Krauß, S. Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics. 1998.

[2] VanderWerf, J., S. Shladover, N. Kourjanskaia, M. Miller, and H. Krishnan. Modeling effects of driver control assistance systems on traffic. *Transportation Research Record,* Vol. 1748, No. 1, 2001, pp. 167-174.

[3] Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[4] Behrisch, M., L. Bieker, J. Erdmann, and D. Krajzewicz. SUMO–simulation of urban mobility: an overview.In *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*, ThinkMind, 2011.

[5] Dosovitskiy, A., G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator.In *Conference on robot learning*, PMLR, 2017. pp. 1-16.

[6] Papadoulis, A., M. Quddus, and M. Imprialou. Evaluating the safety impact of connected and autonomous vehicles on motorways. *Accident Analysis & Prevention,* Vol. 124, 2019, pp. 12-22.

[7] Guériau, M., and I. Dusparic. Quantifying the impact of connected and autonomous vehicles on traffic efficiency and safety in mixed traffic.In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2020. pp. 1-8.

[8] Niroumand, R., L. Hajibabai, A. Hajbabaie, and M. Tajalli. Effects of Autonomous Driving Behavior on Intersection Performance and Safety in the Presence of White Phase for Mixed-Autonomy Traffic Stream. *Transportation Research Record,* Vol. 2676, No. 8, 2022, pp. 112-130.

[9] Morando, M. M., Q. Tian, L. T. Truong, and H. L. Vu. Studying the safety impact of autonomous vehicles using simulation-based surrogate safety measures. *Journal of advanced transportation,* Vol. 2018, No. 1, 2018, p. 6135183.

[10] Yang, S., M. Du, and Q. Chen. Impact of connected and autonomous vehicles on traffic efficiency and safety of an on-ramp. *Simulation Modelling Practice and Theory,* Vol. 113, 2021, p. 102374.

[11] Brar, J. S., and B. Caulfield. Impact of autonomous vehicles on pedestrians' safety.In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2017. pp. 714-719.

[12] Khashayarfard, M., and H. Nassiri. Studying the simultaneous effect of autonomous vehicles and distracted driving on safety at unsignalized intersections. *Journal of advanced transportation,* Vol. 2021, No. 1, 2021, p. 6677010.

[13] Sever, M., N. Zengin, A. Kirli, and M. S. Arslan. Carsickness-based design and development of a controller for autonomous vehicles to improve the comfort of occupants. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering,* Vol. 235, No. 1, 2021, pp. 162-176.

[14] Qin, Y., H. Wang, and B. Ran. Impact of connected and automated vehicles on passenger comfort of traffic flow with vehicle-to-vehicle communications. *KSCE journal of civil engineering,* Vol. 23, 2019, pp. 821-832.

[15] Talebpour, A., and H. S. Mahmassani. Influence of connected and autonomous vehicles on traffic flow stability and throughput. *Transportation research part C: emerging technologies,* Vol. 71, 2016, pp. 143-163.

[16] Stogios, C., D. Kasraian, M. J. Roorda, and M. Hatzopoulou. Simulating impacts of automated driving behavior and traffic conditions on vehicle emissions. *Transportation Research Part D: Transport and Environment,* Vol. 76, 2019, pp. 176-192.
[17] Makridis, M., K. Mattas, C. Mogno, B. Ciuffo, and G. Fontaras. The impact of automation and connectivity on traffic flow and CO2 emissions. A detailed microsimulation study. *Atmospheric Environment,* Vol. 226, 2020, p. 117399.
[18] Cárdenas, J. F. S., J. G. Shin, and S. H. Kim. A few critical human factors for developing sustainable autonomous driving technology. *Sustainability,* Vol. 12, No. 7, 2020, p. 3030.
[19] Sarker, A., H. Shen, M. Rahman, M. Chowdhury, K. Dey, F. Li, Y. Wang, and H. S. Narman. A review of sensing and communication, human factors, and controller aspects for information-aware connected and automated vehicles. *IEEE transactions on intelligent transportation systems,* Vol. 21, No. 1, 2019, pp. 7-29.
[20] Liu, Z., and J. Zhang. Exploring factors in Human Cognition that affect Human Performance during Takeover in Autonomous Driving. *Highlights in Science, Engineering and Technology,* Vol. 85, 2024, pp. 1182-1193.
[21] Li, Y., D. Sun, M. Zhao, J. Chen, Z. Liu, S. Cheng, and T. Chen. MPC-based switched driving model for human vehicle co-piloting considering human factors. *Transportation research part C: emerging technologies,* Vol. 115, 2020, p. 102612.
[22] Raiyn, J., and G. Weidl. Predicting autonomous driving behavior through human factor considerations in safety-critical events. *Smart Cities,* Vol. 7, No. 1, 2024, pp. 460-474.
[23] Xing, Y., C. Lv, D. Cao, and P. Hang. Toward human-vehicle collaboration: Review and perspectives on human-centered collaborative automated driving. *Transportation research part C: emerging technologies,* Vol. 128, 2021, p. 103199.
[24] Sallab, A. E., M. Abdou, E. Perot, and S. Yogamani. Deep reinforcement learning framework for autonomous driving. *arXiv preprint arXiv:1704.02532*, 2017.
[25] Palanisamy, P. Multi-agent connected autonomous driving using deep reinforcement learning.In *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020. pp. 1-7.
[26] Bouton, M., J. Karlsson, A. Nakhaei, K. Fujimura, M. J. Kochenderfer, and J. Tumova. Reinforcement learning with probabilistic guarantees for autonomous driving. *arXiv preprint arXiv:1904.07189*, 2019.
[27] Cao, Z., S. Xu, X. Jiao, H. Peng, and D. Yang. Trustworthy safety improvement for autonomous driving using reinforcement learning. *Transportation research part C: emerging technologies,* Vol. 138, 2022, p. 103656.
[28] Li, D., D. Zhao, Q. Zhang, and Y. Chen. Reinforcement learning and deep learning based lateral control for autonomous driving [application notes]. *IEEE Computational Intelligence Magazine,* Vol. 14, No. 2, 2019, pp. 83-98.
[29] Zhu, M., 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, 2020, p. 102662.
[30] Osiński, B., A. Jakubowski, P. Zięcina, P. Miłoś, C. Galias, S. Homoceanu, and H. Michalewski. Simulation-based reinforcement learning for real-world autonomous driving.In *2020 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2020. pp. 6411-6418.
[31] Chen, B., Z. Yang, S. Huang, X. Du, Z. Cui, J. Bhimani, X. Xie, and N. Mi. Cyber-physical system enabled nearby traffic flow modelling for autonomous vehicles.In *2017 IEEE 36th*

international performance computing and communications conference (IPCCC), IEEE, 2017. pp. 1-6.
[32] Zhang, T., Y. Zou, X. Zhang, N. Guo, and W. Wang. Data-driven based cruise control of connected and automated vehicles under cyber-physical system framework. *IEEE transactions on intelligent transportation systems,* Vol. 22, No. 10, 2020, pp. 6307-6319.
[33] Guo, J., L. Li, J. Wang, and K. Li. Cyber-physical system-based path tracking control of autonomous vehicles under cyber-attacks. *IEEE Transactions on Industrial Informatics,* Vol. 19, No. 5, 2022, pp. 6624-6635.
[34] Ding, C. Efficient firmware distribution in vehicular networks. 2021.
[35] Chattopadhyay, A., and K.-Y. Lam. Security of autonomous vehicle as a cyber-physical system.In *2017 7th International Symposium on Embedded Computing and System Design (ISED)*, IEEE, 2017. pp. 1-6.
[36] Milanés, V., and S. E. Shladover. Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. *Transportation research part C: emerging technologies,* Vol. 48, 2014, pp. 285-300.
[37] Xiao, L., M. Wang, and B. Van Arem. Realistic car-following models for microscopic simulation of adaptive and cooperative adaptive cruise control vehicles. *Transportation Research Record,* Vol. 2623, No. 1, 2017, pp. 1-9.
[38] Xiao, L., M. Wang, W. Schakel, and B. van Arem. Unravelling effects of cooperative adaptive cruise control deactivation on traffic flow characteristics at merging bottlenecks. *Transportation research part C: emerging technologies,* Vol. 96, 2018, pp. 380-397.
[39] Porfyri, K. N., E. Mintsis, and E. Mitsakis. Assessment of ACC and CACC systems using SUMO. *EPiC Series in Engineering,* Vol. 2, 2018, pp. 82-93.
[40] Nie, X., Y. Liang, and K. Ohkura. Autonomous highway driving using reinforcement learning with safety check system based on time-to-collision. *Artificial Life and Robotics,* Vol. 28, No. 1, 2023, pp. 158-165.
[41] Hayward, J. *Near misses as a measure of safety at urban intersections*. Pennsylvania Transportation and Traffic Safety Center, 1971.
[42] Treiber, M., A. Hennecke, and D. Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical review E,* Vol. 62, No. 2, 2000, p. 1805.
[43] Batra, M., J. McPhee, and N. L. Azad. Real-time model predictive control of connected electric vehicles. *Vehicle System Dynamics,* Vol. 57, No. 11, 2019, pp. 1720-1743.
[44] Ma, J., X. Li, F. Zhou, J. Hu, and B. B. Park. Parsimonious shooting heuristic for trajectory design of connected automated traffic part II: Computational issues and optimization. *Transportation Research Part B: Methodological,* Vol. 95, 2017, pp. 421-441.
[45] Zhou, M., X. Qu, and X. Li. A recurrent neural network based microscopic car following model to predict traffic oscillation. *Transportation research part C: emerging technologies,* Vol. 84, 2017, pp. 245-264.
[46] Long, X., H. Zhao, L. Li, L. Teng, and S. Galland. Modeling Safe and Smooth Car-Following Behavior: An LSTM-Attention Based TD3 Approach.In *2023 China Automation Congress (CAC)*, IEEE, 2023. pp. 4919-4924.
[47] Gurda, A. Evaluating the psychometric properties of the aggressive driving behavior questionnaire (ADBQ). 2012.