Enhancing Autonomous Driving Decision: A Hybrid Deep Reinforcement Learning-Kinematic-Based Autopilot Framework for Complex Motorway Scenes

Yongqiang Lu, Hongjie Ma, Edward Smart, *Member, IEEE,* and Hui Yu, *Senior Member, IEEE*

***Abstract*—Autonomous vehicles (AVs) still pose challenges in improving intelligence, safety, and reliability in complex motorway scenarios. Recently, deep reinforcement learning (DRL) has demonstrated superior decision-making capabilities in dynamic environments compared to rule-based methods. However, it requires considerable training resources due to a lack of innovative DRL component design (e.g., state space and reward) to link observation and action accurately. Its opaque nature may also result in hazardous driving conditions. In this paper, we introduce a hybrid autopilot framework that amalgamates three modules: (i) DRL is employed to build a smart, learnable, and scalable driving policy across various motorway scenarios; (ii) a kinematic-based co-pilot strategy is devised to bolster training efficiency and provide flexible decision-making guidance; and (iii) a rule-based system assesses and determines the final action outputs in real-time between itself and the DRL policy to further enhance safety. Extensive simulations are conducted under different complex motorway scenarios. The results indicate that the proposed framework surpasses the baseline DRL policy in terms of training efficiency, intelligence, safety, and reliability.**

***Index Terms*—Autonomous vehicle, deep reinforcement learning, kinematic model, co-pilot strategy, training efficiency, hybrid autopilot framework.**

I. Introduction

Human error contributes to approximately 95% of all road accidents, underscoring its crucial role in road safety [1]. Autonomous vehicles (AVs) are recognised as a significant element in diminishing traffic accidents due to their capacity to minimise human error [2]-[4]. The decision-making algorithm is crucial to the functionality of AVs, functioning similarly to the human brain. This algorithm generates intelligent driving commands (e.g., *obstacle avoidance*) based on environmental perception outputs, subsequently providing the control module with targeted actions such as throttle, brake, and steering [5]. Of the numerous objectives in decision-making, safety and intelligence present significant challenges to the broad implementation of autonomous driving technologies.

Yongqiang Lu, Hongjie Ma, and Edward Smart are with the School of Energy and Electronic Engineering, University of Portsmouth, Portsmouth PO1 3HF, UK. (e-mail: yongqiang.lu@port.ac.uk; hongjie.ma@port.ac.uk; edward.smart@port.ac.uk).

Hui Yu is with the cSCAN Centre, University of Glasgow, Glasgow G12 8QB, UK. (e-mail: hui.yu@glasgow.ac.uk).

Currently, the AV industry has successfully deployed decision-making strategies based on explainable algorithms, specifically rule-based and dynamic/kinematic vehicle model-based systems. Rule-based systems, such as finite state machines (FSM) [6], [7] and fuzzy rule-based control algorithms [7], establish rigorous logical connections between surrounding environments and driving behaviours, providing robust interpretability and reasoning. Conversely, dynamic or kinematic vehicle model-based methods rely on explicit and precise analytical equations to perform reliable driving tasks, e.g., motion planners [9], [9]. These explainable algorithms ensure the safe driving of AVs in known environments by adhering to predefined rules. However, their lack of learning capabilities makes them prone to failure in undefined scenarios, underscoring their limitations in practical real-world applications. In contrast, learning-based algorithms, such as deep learning (DL)- and DRL-based methods [11], effectively address these challenges due to their superior adaptability and scalability. Examples include DL-based approaches for trajectory planning through imitation learning [12] and DRL-based strategies for path planning in motorway scenarios [13]. Additionally, hybrid policies that combine rule- and DRL-based approaches have been proposed for roundabout applications [14] and connected and autonomous vehicles (CAVs) [15]. Nevertheless, DL-based methods necessitate extensive training data and exhibit limited scalability in uncertain environments. DRL-based approaches achieve remarkable accuracy and adaptability through traffic interactions, proving highly suitable for AVs' decision-making. However, they demand considerable learning resources, which may compromise training efficiency. Additionally, the safety and reliability concerns regarding the 'black-box' nature of these systems, alongside their capability to function intelligently in diverse environments, still require further enhancement.

To improve training efficiency, one effective approach involves decomposing and optimising the DRL's state space [16]. For example, selecting specific raw data, such as vehicle speed and steering angle, and abstracting relevant raw perceptual details (e.g., the distance between vehicles and their velocities) into a compact indirect representation, such as time-to-collision (TTC). However, state-space abstraction entails the discretisation of continuous environmental information, resulting in reduced precision during the training process compared to a full state-space model. Moreover, given

the complex and opaque nature of the DRL algorithm, sole reliance on this method to independently explore decision-making correlations may yield an unintelligent and unsafe model. This model may demonstrate irrational following distances and inflexible or illogical lane-changing behaviours. Furthermore, the safety and reliability may be compromised in varied environments because of the lack of comprehensive scenario datasets. In summary, the DRL-based decision-making approach still presents several significant limitations:

- **Suboptimal Training Efficiency**: DRL demands extensive learning resources, impacting training efficiency.
- **Inadequate Intelligence, Safety, and Reliability**: The DRL's opaque nature and diverse driving scenarios complicate the assurance of intelligence, safety, and reliability.

To address the challenges outlined, a hybrid DRL-kinematic-based autopilot framework is proposed. This framework aims to train an intelligent, safe, and reliable AV driving decision model applicable across various motorway scenarios. The approach tackles the bottleneck concerning training data requirements and shortens the learning duration. Additionally, potential high-risk DRL action outputs are actively monitored and constrained in real-time through the deployment of a rule-based policy, further enhancing safety and reliability. The principal contributions include:

- **Effective State-Reward Design**: An expertly crafted DRL state space with a corresponding reward function ensures swift and accurate correlation between perception and decision-making.
- **Innovative Co-Pilot Strategy**: A kinematic-based co-pilot strategy is developed to effectively guide the DRL algorithm in exploring and establishing safe, intelligent decision-making processes. It also enhances adaptability by utilising representative scenario training data.
- **Rule-Based Policy as Guardian**: A rule-based policy is employed to further improve safety and reliability.

The remainder of this paper is organised as follows: Section II discusses relevant work. Section III elaborates on the simulated driving system. Section IV introduces the methodology of the decision-making framework, including the kinematic-based co-pilot system, the DRL-based driving decision algorithm, the rule-based policy, and the autopilot framework integration. Section V presents an analysis of the results. Finally, Section VI concludes the paper and discusses potential future research.

II. Related Work

In this section, research on autonomous driving is surveyed, focusing on the domains of DRL-based AV decision-making, as well as training efficiency and driving performance.

A. DRL-based Decision-Making

Recently, DRL algorithms have emerged as powerful tools for long sequential decision-making challenges, attracting significant attention in the field [17]. Chae *et al*. [17] proposed a deep Q-network (DQN)-based braking strategy to minimise collision risks in dynamic environments, predicated on a well-designed reward function. Similarly, Fu *et al*. [19] developed a deep deterministic policy gradient (DDPG)-based autopilot braking system aimed at enhancing CAVs' safety in emergency situations, incorporating a multi-objective reward function to balance considerations of braking timing, accident scenarios, and passenger experiences. A Modularized RL model, as discussed in [20], decomposes complex multi-objective problems into distinct DDPG-based driving strategies—free driving and car following—to ensure desired speeds and safe following distances, respectively. This model enables AVs to drive safely, smoothly, and comfortably. Li and Okhrina [21] proposed a DRL framework to optimise car-following behaviours based on human driving experience, designing a two-stage agent aimed at enhancing safety and efficiency in human-robot interaction traffic. Additionally, [22] introduced a supervised RL-based car-following model for adaptive cruise control (ACC), trained with an actual human-driving dataset. References [23] and [24] developed a RL strategy to devise an autonomous longitudinal vehicle controller in the cooperative ACC system, demonstrating efficiency and flexibility in driving behaviours, including reduced delay, speed and safe distance adaptation across diverse experimental setups. Furthermore, Liao *et al*. [2] introduced a duelling DQN algorithm to enhance AVs' lane-changing behaviours, enabling more flexible overtaking manoeuvres on motorways. Concurrently, Wang *et al*. [2] proposed a DQN-based lane-changing approach for motorway scenarios, optimising individual vehicle efficiency and overall traffic flow by incorporating traffic indicators and flow rates into the reward function. Despite these advancements achieving notable accuracy in driving decisions, the risks of fully autonomous driving persist and necessitate continuous improvement [26]. Thus, there remains a need to further refine the safety and intelligence of decision-making processes while considering training efficiency.

B. Training Efficiency and Driving Performance

One of the prominent challenges training efficiency of traditional DRL grapples with is state space optimisation. Theoretically, with a defined state space $\mathcal{S}$ and action space $\mathcal{A}$, DRL must explore the optimal strategy from a maximum of $\|\mathcal{A}\|^{|\mathcal{S}|}$ potential strategies at each decision step, implying that training costs will increase exponentially with each newly introduced state feature [27], [28]. Consequently, in cases of complex state spaces, the efficiency of DRL training is compromised, necessitating a well-designed state space to boost training efficiency. Ferns *et al*. [28] analysed and experimented with similarity measures across diverse states, accomplishing a trade-off between computational time, memory usage, and performance. Asadi *et al*. [30] developed a partitioned state space containing fewer states than the original Markov decision process by extending the $\varepsilon-reduction$ model. Consequently, model training within this partitioned space proceeds more rapidly than in the full state. However,

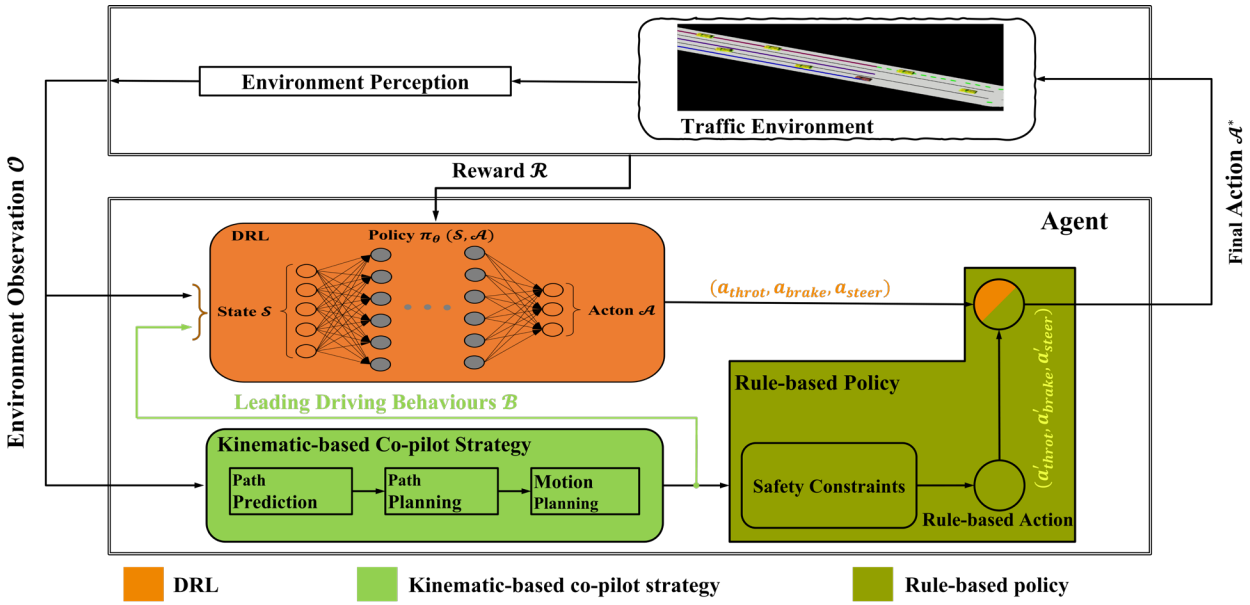


Fig. 1. The proposed hybrid DRL-kinematic-based autopilot framework.

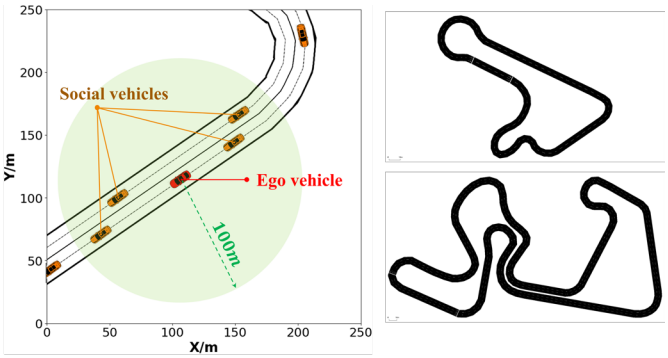
this efficiency comes at the cost of reduced cumulative long-term rewards. An iterative dimensionality reduced RL framework, as introduced in [30], accelerates the RL algorithm convergence towards an optimal policy by simplifying state representations with minimal variable expressions. Similarly, Curran *et al*. [32] utilised the principal component analysis method to accelerate the DRL training process by mapping a large state space to a lower dimensionality representation. Nevertheless, [31] and [32] underscore the vital trade-off between convergence and performance, noting that low-dimensional manifolds may inevitably omit significant data. To address the performance degradation from reduced state spaces, Fu *et al*. [33] proposed a hierarchical RL strategy. This approach minimises the state space and optimises the reward function by concentrating on action sub-rewards, thereby enhancing advantageous action selection and improving convergence speed.

Safety and intelligence in decision-making are essential for integrating AVs into intelligent transport systems. Bouton *et al*. [34] introduced a safe RL-based model checker, enabling intelligent navigation of AVs through complex intersections with various traffic participants. Moreover, [35] combined planning and DRL to construct a driving decision model suited for diverse motorway scenarios, employing an expert system to prevent collisions while increasing computational time to enhance precision. Similarly, Fu *et al*. [15] developed a decision-making framework that integrates expert knowledge with DRL to overcome the limitations of a standalone policy, thus enhancing safety and intelligence. However, it still requires substantial data to train an essential model. Chen *et al*. [36] introduced a motion planning strategy that employs a DQN combined with fuzzy logic to manage the uncertain action outputs, but the learning duration required to enhance driving stability in a single scenario remains high. Furthermore, [14] implemented a rule-based policy to address the opaque issues associated with RL. Additionally, Kherroubi *et al*. [37] employed an artificial neural network to predict the behaviours of surrounding human-operated vehicles, thereby facilitating the generation of expected acceleration through these behaviours. While the aforementioned decision-making strategies exhibit notable intelligence and safety, they fail to adequately address the training efficiency challenges of DRL, requiring substantial resources such as prolonged learning periods ([35], [36]) or significant training data ([14, 15, 37]) for effective model training. Additionally, safety, reliability, and intelligence across diverse motorway scenarios still necessitate further improvements.

In summary, the challenge involves achieving a balance between training efficiency and driving performance in practical applicability. Simplifying information to decompose the state space could reduce DRL capability, while prioritising driving performance might worsen training efficiency. Consequently, this paper introduces a hybrid DRL-kinematic-based autopilot framework designed to optimise the DRL in terms of state space, action space, and reward function. This aims to lessen reliance on extensive training resources and to enable robust navigation across complex motorway scenarios.

III. Simulated System Description

The proposed hybrid autopilot framework is developed and assessed on Huawei's SMARTS platform [38], the foundation for the 2019 DriveML Huawei Autonomous Driving Challenge in the UK. This platform can currently simulate a broad range of real traffic scenarios, fulfilling diverse interaction demands through both rule-based and DRL-based agents. As depicted in Fig. 2 (1), the traffic setup involves solely one ego vehicle (controlled by DRL algorithm) and a different number of social vehicles (SVs—vehicles sharing the environment with the AV, managed by a rule-based planner). The ego vehicle's perception range spans 100 meters, encompassing data on its and surrounding SVs' states, a 2D RGB BEV, and an occupancy grid map. Additionally, various complex one-way, closed-loop



(1). Interaction of the ego vehicle (red) with social vehicles (yellow) on a one-way motorway.

(2). Complex closed-loop motorway tracks utilised in the study.

Fig. 2. Simulation system of the study.

motorway maps featuring straight lanes, U-, S-, hairpin-, and winding sharp curves have been collated for training and evaluation, as demonstrated in Fig. 2 (2). The essential platform settings for the study are described below:

a) The motorways are devoid of slip roads, intersections, and traffic lights, featuring single, dual, and triple lanes, with the potential to extend to a five-lane configuration.
b) There is no natural or man-made environment beyond the road edge due to the absence of perception requirements.
c) We utilise single-agent learning, interacting solely with a variable number of rule-based vehicles exhibiting diverse random behaviours; no other participants, such as pedestrians, bicycles, and motorcycles, are involved.
d) At the start of each epoch, vehicles are spawned at random positions.
e) Crash events are monitored in each iteration, including vehicle-vehicle collisions, road-offs, and rollovers.

IV. Methodology

This section delineates the proposed hybrid DRL-kinematic-based autopilot framework, as illustrated in Fig. 1. Initially, using environmental perception as input, the kinematic-based co-pilot strategy generates leading driving behaviours through path prediction, dynamic path planning, and motion planning calculations. Subsequently, DRL-based driving actions are inferred from the leading reference and other perceptions. Ultimately, the rule-based policy identifies the final actions, ensuring adherence to safety constraints.

A. Kinematic-based Co-pilot Strategy

1) Path Prediction

Path prediction focuses on forecasting a vehicle's future position based on current driving behaviour to infer subsequent driving intentions, such as lane-keeping or changing lanes. Typically, a vehicle's movement is constrained by a kinematic model, enabling driving predictions through the constant yaw rate and acceleration model. Thus, the kinematic model-based trajectory prediction $\mathcal{T}_{kml}$ for the subsequent cycle T at location $[x(t), y(t)]$ can be derived from the following:

$$\mathcal{T}_{kml}(t+T)=\begin{cases}x(t)+\Delta x(T)\\ y(t)+\Delta y(T)\end{cases} \tag{1}$$

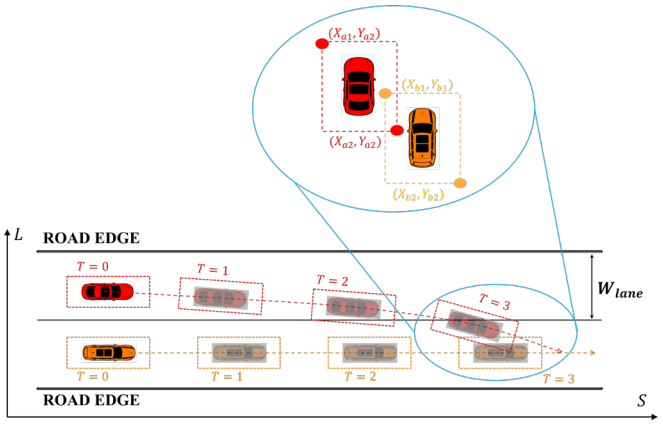


Fig. 3. Collision conditions between the ego vehicle and SV.

To ensure the predicted trajectory aligns closely with actual driving as the timestep increases, the waypoint $wp_{closest}(t)$ nearest to the ego vehicle on path $\mathcal{T}_{kml}$ and the weight function $w(t)=1-1/(1+e^{-6\cdot(t-1)})$ are integrated to determine the final trajectory prediction as follows:

$$\mathcal{T}_{\text{targt}}=w(t)\cdot\mathcal{T}_{kml}(t)+(1-w(t))\cdot wp_{closest}(t) \tag{2}$$

Then, the trajectory predictions are limited to the ego vehicle and the nearest SVs from lanes in the front, middle, and rear to minimise computational load, ensuring that the number of path calculations does not exceed $N_{\mathcal{T}_{\text{targt}}}=\begin{cases}3l, l<3\\ 9, l\geq 3\end{cases}$ where l represents the number of lanes. In this study, the time domain for trajectory prediction is confined to 20 timesteps.

Subsequently, we calculate all potential overlap cases using the vehicle's rectangular outline, which includes its actual dimensions plus a safety margin, as illustrated in Fig. 3. By analysing the enlarged view in Fig. 3, the collision criterion is established according to (3) based on the coordinates (X,Y) of the rectangle outline's left diagonal.

$$\begin{cases}|\sum_{i=1,2}(X_{b_i}-X_{a_i})|\leq\sum_{k=a,b}|X_{k_2}-X_{k_1}|\\ |\sum_{i=1,2}(Y_{b_i}-Y_{a_i})|\leq\sum_{k=a,b}|Y_{k_2}-Y_{k_1}|\end{cases} \tag{3}$$

Upon collision, we simplify the problem by estimating the minimum Euclidean distance $D_{collisn}$ and TTC from the collision location to determine potential hazards associated with the ego vehicle's current manoeuvre in the short term, that is:

$$\begin{aligned}C_{risk}=\{&\min_{\substack{1\leq m\leq 9\\ 1\leq k\leq n}} D_{collisn}\parallel P_{ego}-P^{p}_{m_k}\parallel_2<10 \ \ or\\ &\min_{\substack{1\leq m\leq 9\\ 1\leq k\leq n}}(D_{collisn}\parallel P_{ego}-P^{p}_{m_k}\parallel_2<12 \ and \ TTC_{ego2p_{m_k}}<3)\}\end{aligned} \tag{4}$$

where P_{ego} and P^{p} represent the position of the ego vehicle and the waypoint position in the trajectory predictions of SVs, respectively. Dynamic path planning is initiated when (4) is satisfied.

2) Dynamic Path Planning

Dynamic path planning seeks to identify the optimal route for AV navigation in subsequent steps and provides short-term guidance. Initially, potential drivable zones are delineated by

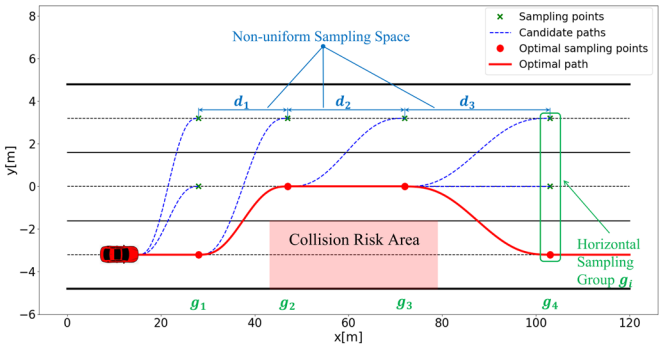


Fig. 4. Dynamic path planning for the AV.

excluding areas deemed high-risk for collisions as determined by (4). Subsequently, a longitudinal non-uniform lateral equidistant spatial sampling method is applied to generate feasible destination groups for future timesteps, effectively reducing computational demands in path planning, as depicted in Fig. 4. Optimal spatial sampling points are then selected for preferred path planning based on predefined penalty criteria:

a) ***Deviation from lane centre***, defined by the deviation $D(s)$ and lane width L:

$$P_{centre} = -0.1 \cdot \int \frac{2D(s)}{L} ds \tag{5}$$

b) ***Steering change***, where excessive adjustment heightens rollover risk at high speeds:

$$P_{steering} = \begin{cases} 0, & v \le 60km/h \\ -(\frac{v-60}{20} \cdot \frac{steering}{4})^2, & v > 60km/h \end{cases} \tag{6}$$

c) ***Crash risk***, evaluating crashes caused by following distance on current and candidate trajectories:

$$P_{crash} = sum(P_{curr}, P_{planig}) \begin{cases} P_{curr} = -5 \\ P_{planig} = -5 \end{cases} \tag{7}$$

For $\mathcal{T}_k \,|\, 1 \le k \le n$, the optimal trajectory is yielded by:

$$P(\mathcal{T}) = \frac{P_{centre}(\mathcal{T}) + P_{steering}(\mathcal{T}) + P_{crash}(\mathcal{T})}{200}$$
$$\text{The optimal trajectory: } \mathcal{T}^* = \min_{1 \le k \le n}(|\, P(\mathcal{T}_k)\,|) \tag{8}$$

In this study, each optimal sampling point is achieved subsequent to the determination of its predecessor (The ego vehicle's current location is set at the first optimal sampling point). Subsequently, the optimal sampling point set is fitted with a spline curve to generate the desired driving path, as the bold red line in Fig. 4. This method significantly reduces the time complexity from $O(2^n)$ to $O(n)$ compared to traditional global traversal optimisation algorithms, thereby enhancing the real-time decision-making performance.

3) **Motion Planning**

We propose a leading driving behaviour planning as an observation $\mathcal{B} = [\mathcal{B}_l, \mathcal{B}_s, \mathcal{B}_a]$, which is coupled with an associated reward design to aid the DRL agent in swiftly and accurately achieving an efficient decision-making model. Rather than offering step-by-step guidance, this motion planning strategy provides the agent with definitive directional co-pilot cues towards the optimal planning path's endpoint. Nevertheless, the agent is required to autonomously navigate the intermediate actions from departure to destination. Consequently, this strategy resolves issues inherent in rule-based models, such as unending *if...else...* loops or decision-making failures arising from varying traffic conditions on identical routes, thus bolstering environmental adaptability and robustness.

The next driving lane is first inferred and represented using the relative lane index to improve generalisation as follows:

$$\mathcal{B}_l = l^i_{planig} - l^i_{curr}, \mathcal{B}_l \in [-l^i_{\max}, l^i_{\max}] \tag{9}$$

Subsequently, the next step of relative steering angle γ^p_{ego} is deduced through the optimal path's start and target waypoint and transformed into the discrete value in the following:

$$\mathcal{B}_s = \text{sign}(\gamma^p_{ego}) = \begin{cases} +1, \text{ Left turn} \\ 0, \text{ Maintain} \\ -1, \text{ Right turn} \end{cases} \tag{10}$$

The acceleration estimation a_p based on the trajectory prediction is assigned to acceleration planning a^p_{ego} for multi-lane scenes. However, if there is no predicted collision in the planning period under single-lane situations, the value assignment for acceleration planning will be $a^p_{ego} = a_p$. Despite potential future collisions during a car-following distance of $d_{ego2sv} > d_{safe}$, the ego vehicle is allowed to continue accelerating at a value of $a^p_{ego} = 1m/s^2$, thereby prompting the vehicle to drive near the limit of d_{safe} for traffic efficiency. However, deceleration is imperative to lengthen the safety distance using (11) under $d_{ego2sv} \le d_{safe}$.

$$a^p_{ego} = \min[a_p, \frac{v_{ego}(0.025 \cdot d_{ego2sv} - 2)}{3.6}] \tag{11}$$

Finally, the acceleration planning is as follows:

$$\mathcal{B}_a = \text{sign}(a^p_{ego}) = \begin{cases} +1, \text{ Acceleration} \\ 0, \text{ Maintain} \\ -1, \text{ Deceleration} \end{cases} \tag{12}$$

Specifically, trajectory prediction is not initiated, and motion planning solely provides safe lane-keeping manipulation to avoid unnecessary lane changes by the ego vehicle in the absence of surrounding SVs.

B. Deep Reinforcement Learning

The proximal policy optimisation (PPO) algorithm [39] is utilised to train the DRL agent for motorway scenarios. Owing to its ability to balance tuning complexity, learning efficiency, and accuracy, PPO is widely applied in AV research. The new policy is updated using (13) during the iteration process.

$$\theta_{k+1} = \arg\max_{\theta} \mathcal{L}^{CLIP}_{\theta_k}(\theta) \tag{13}$$

by taking K steps of minibatch stochastic gradient descent (SGD) to maximise the objective. Its loss function is described as follows:

$$\mathcal{L}^{TOTAL}_t = -\mathcal{L}^{CLIP}_t + c_1 \mathcal{L}^{VF}_t - c_2 \mathcal{L}^{ENTROPY}_t \tag{14}$$

1) **State Space**

TABLE I
THE EXPLANATION OF THE STATE SPACE

VARIABLES	EXPLANATION
d_{ego2c}	The lateral distance between the AV and the lane centre
$\theta_{wpsHdgErrList}$	A list of heading errors of waypoints
v_{ego}	The speed of ego vehicle
$\gamma_{egoSteer}$	The steering in radians
$d_{ego2svList}$	Distance list from the ego vehicle to the closest SVs ahead in each lane
$t_{ttcList}$	TTC
$v_{ego2svClost}$	The relative speed between the ego vehicle and the closest SVs ahead in each lane
$\boldsymbol{s}_{svTrffSigList}$	Driving intentions of the nearest SVs
$\boldsymbol{\mathcal{B}}$	The leading driving behaviour planning

The explanation and mathematical expression of the state space are detailed in TABLE I and presented as (15), respectively.

$$\mathcal{S}=\{d_{ego2c},\theta_{wpsHdgErrList},v_{ego},\gamma_{egoSteer},d_{ego2svList},t_{ttcList},v_{ego2svClost},\mathrm{s}_{\mathrm{svTrffSigList}},\boldsymbol{\mathcal{B}}\}\in\boldsymbol{\mathcal{S}} \tag{15}$$

The proposed strategy utilises a leading driving behaviour planning $\boldsymbol{\mathcal{B}}$ instead of a simplified proximity map P_{map} that still encompasses raw perception, which reduces the time required to learn the decision-making association. Additionally, the extracted $\boldsymbol{\mathcal{B}}$ considers the precise states of both the front and rear SVs, enabling the DRL agent to effectively comprehend the surrounding global environment. Furthermore, the nearest SVs' driving intentions $\mathrm{s}_{\mathrm{svTrffSigList}}$ are estimated based on their future driving predictions. Analogous to the turn signal in practical driving, the SV is to signal left $(+1)$ when driving away to the left, while for signal right (-1), lane keeping is (0).

2) **Action Space**

The mathematical expression of the action space at timestep t is expressed below:

$$\mathcal{A}=\{a_a(t),a_s(t)\}\in\boldsymbol{\mathcal{A}} \tag{16}$$

where $a_a(t)\in[-1,1]$ represents the ego vehicle's acceleration or deceleration value, and $a_s(t)\in[-1,1]$ denotes the residual steering. As the SMARTS provides throttle, brake, and steering to control vehicle driving, action $\mathcal{A}$ can be further refined to control actions of $(throttle,brake,steering)$. During practical driving, $(throttle,brake)$ represents a pair of opposing actions. Consequently, $a_a(t)$ should be decomposed into distinct actions as follows:

$$\begin{cases}throttle=a_a\\brake=0\end{cases},\ \text{if } a_a\geq 0,\qquad \begin{cases}throttle=0\\brake=a_a\end{cases},\ \text{else.} \tag{17}$$

To mitigate uncontrollable driving that may result from excessive steering changes output by the DRL algorithm, reference is made to the steering representation used in the drive++ model, as follows:

Algorithm 1: Rule-Based Policy for Improving Safety

Input: Post-processing DRL action $(throttle,brake,steering)$

$(throttle^*,brake^*,steering^*)\leftarrow(throttle,brake,steering)$;
if Lateral distance $d_{ego2c}>1$ **then**
  $steering^*\leftarrow CLIP(\gamma^p_{ego},-25^\circ,+25^\circ)$;
end if
if $\varepsilon_{\min}\leq d_{safe}<\varepsilon_{\max}$ **and** number of lanes ≥ 1 **and** there is no SV behind within d_{safeLC} in the target lane **and** $v_{ego}<10km/h$ **then**
  if $a^p_{ego}>0$ **then**
    Update the safe acceleration value: $a^s_{ego}=0.3\cdot[\kappa\cdot(1-\gamma^p_{ego}/45)^2\cdot 0.99+0.1]$;
    $throttle^*\leftarrow\min(a^p_{ego},a^s_{ego})$, $brake^*\leftarrow 0$;
  else
    $throttle^*\leftarrow 0$, $brake^*\leftarrow -a^p_{ego}$;
  end if
  $steering^*\leftarrow\gamma^p_{ego}$;
end if
if $d_{safe}<\varepsilon_{\min}$ **then**
  $throttle^*\leftarrow 0$, $brake^*\leftarrow 1$;
end if

Output: Final action $(throttle^*, brake^*, steering^*)$

$$steering=CLIP(steering_{old}+25\cdot a_s,-45^\circ,+45^\circ) \tag{18}$$

3) **Reward Function**

To establish a correlation with the proposed leading driving behaviour planning $\boldsymbol{\mathcal{B}}$ and enhance training efficiency, a comprehensive set of task-oriented rewards is designed based on criteria for optimal path selection. Compared to the baseline model, significant emphasis is placed on lane change rewards in (19) and (20). These rewards are specifically crafted to guide the agent in learning safe and efficient lane change policies by following precise co-pilot instructions.

$$r=t_{cl}\cdot\max_{i=1,2,3}(|\,P^i{}_{steering}\,|) \tag{19}$$

where the smoothing factor $\tau_{cl}=3$ is utilised to moderate lane changes within the simulator. Additionally, (19) compensates for steering penalties by applying the maximum penalty from the previous three cycles if the lane change is completed.

Equation (20) is designed to impose penalties on the ego vehicle for failing to adhere to lane change guidance in high-speed scenarios. Additionally, penalties associated with deviation from the lane centreline and steering changes are nullified under these conditions.

$$p=-\,\text{Number of lanes}\times|\,\boldsymbol{\mathcal{B}}_l\,| \tag{20}$$

Furthermore, (21) introduces a penalty for excessive driving speed on curves to mitigate the risk of rollover. Additionally, the crash penalty, compared to (7), accounts for potential collisions caused by insufficient TTC.

$$p=-\Big(\big(v_{ego}\,|_{v_{ego}>70km/h}-70\big)\cdot\big(\mathrm{std}\big(\theta_{wpsHdgErrList}\big)-0.35\big)\Big)^2 \tag{21}$$

Consequently, the reward formulation fed back into the DRL model is defined as follows:

Algorithm 2: The Hybrid DRL-Kinematic-Based Autopilot Framework

Input: All initial PPO parameters, observation $\mathcal{O}$;
For $k \leftarrow 1$ **to** $timesteps\,N$ **, do**
  Initialise the crash risk list $C_{risk} = empty$;
  Compute *path prediction function*, **then** estimate the future ego's crash risk C_{risk};
  Initialise the optimal trajectory waypoint queue $q = empty$;
  If C_{risk} is not empty, **then**
    Execute *dynamic path planning function*, **then** get the optimal trajectory $\mathcal{T}^*$;
    Store waypoints $ws_i\,|_{i=1,2,3,\ldots,n} \in \mathcal{T}^*$ to q;
  End if
  If q is not empty, **then**
    Compute *motion planning function* via q;
  else
    Compute *motion planning function* via lane-keeping rules;
  End if
  Get motion planning results $(l^i_{planig}, \gamma^p_{ego}, a^p_{ego})$ **and** driving behaviour planning $\mathcal{B}$;
  Extract other state vectors, e.g., v_{ego}, $\tau_{ttcList}$, **then** construct the state space $\mathcal{S}$;
  $\mathcal{A}^* \leftarrow \mathcal{A} \leftarrow$ PPO;
  If the ego vehicle encounters crash risk or stop (waiting) conditions, **then**
    Use the rule-based action: $\mathcal{A}^* \leftarrow \mathcal{A}^p$
  End if
  Compute corresponding reward $\mathcal{R}$ from $(\mathcal{S}, \mathcal{A})$, **and** get next observation $\mathcal{O}'$;
  Calculate PPO update: $\theta_{new} = \arg\max_{\theta} \mathcal{L}^{CLIP}_{\theta_{old}}(\theta)$;
  $\theta_{old} \leftarrow \theta_{new}$;
End for
Output: Final action $\mathcal{A}^*$

$$\mathcal{R} = \frac{1}{200} \cdot \begin{cases} -R + P, & \text{if crashed} \\ R + P, & \text{else} \end{cases} \in \mathbb{R} \tag{22}$$

where R, P are the total bonus and penalties, respectively.

C. Rule-based Policy

The rule-based policy is formulated to validate the opaque DRL-based action outputs in real-time, thus enhancing safety further, as depicted in Algorithm 1. Initially, steering is adapted, and the current speed is maintained if the ego vehicle may drive off the road (i.e., $d_{ego2c} > 1$), thus enabling the AV to align with and follow the lane centreline smoothly. Subsequently, we consider the slow car-following scenarios wherein the safe distance between the ego vehicle and the leading SV is $\varepsilon_{\min} \le d_{safe} < \varepsilon_{\max}$. Specifically, $\varepsilon_{\min}$ and $\varepsilon_{\max}$ are the permissible minimum and maximum safety distances for collision avoidance, respectively. Also, the SV is considered to stop ahead where the safety distance maintains $d_{safe} < \varepsilon_{\min}$. Thus, the ego vehicle must halt behind and wait in line.

TABLE II
THE SIMULATION SETTING FOR TRAINING AND VALIDATION

STAGE	PARAMETERS	VALUES
Training	Tracks (*S-single, D-dual, T-triple*)	[S, S, S, S, D, D, D, D, D, T, T, T, T, T, T]
	Number of SVs for each track	[10, 15, 20, 25, 10, 15, 30, 35, 25, 10, 25, 30, 45, 25, 55]
	Timestep duration	100 ms
	Maximum allowable speed	120 km/h
	Number of SGD iterations	10
	SGD minibatch size	4,096
	Training batch size	131,072
	GAE lambda	0.95
	PPO epsilon	0.2
	Learning rate schedule (*m - million*)	{[0, 1e-3], [3m, 5e-4], [6m, 1e-4], [9m, 5e-5], , [12m, 1e-5], }
	Maximum timestep length	1,000
Validation	Tracks	[S, S, D, D, T, T, T]
	Number of SVs for each track	[15, 25, 25, 35, 35, 45, 55]
	Seeds per track	30 non-training seeds
	Epochs per seed	10

D. Decision-making Framework Integration

Building on the methodologies delineated above, the exploration of the proposed framework is illustrated in Algorithm 2 and can be better comprehended in conjunction with Fig. 1. Notably, an epoch will end at the ego vehicle crashes or attains the maximum number of timesteps.

V. RESULTS AND DISCUSSION

The performance of the proposed framework is assessed across various motorway scenarios, focusing on training efficiency, intelligence, safety, reliability, and real-time capability. This work represents an advancement over our entry submitted to the 2019 DriveML challenge (UK) and innovates upon the drive++ model, as outlined in [40], which secured first place[1] in the challenge. The work is benchmarked against the drive++ method. Simulation settings for training and validation are depicted in TABLE II. Experiments are conducted on an Intel Xeon W-2145 @ 3.70GHz CPU using a Linux system.

A. Training Efficiency

The training efficiency of the proposal is discussed using the following metrics: (i) the speed of reaching the inflexion point and (ii) the total number of training timesteps.

Fig. 5 illustrates the total loss of all models across various scenarios and random seeds. Analysis of total loss variations reveals that the proposed approach takes approximately 8 million timesteps to reach the inflection point, after which it maintains stability. Despite variations in seeds and track quantities, training performance remains consistent. In contrast, the drive++ model attains the inflection point at around 15 million timesteps. This disparity highlights the enhanced convergence speed of the proposed method compared to the drive++ method, attributed primarily to the effective identification of the optimal policy during training through coordinated guidance of the co-pilot strategy and

[1] The challenge leaderboard can be found at https://competitions.codalab.org/competitions/21639

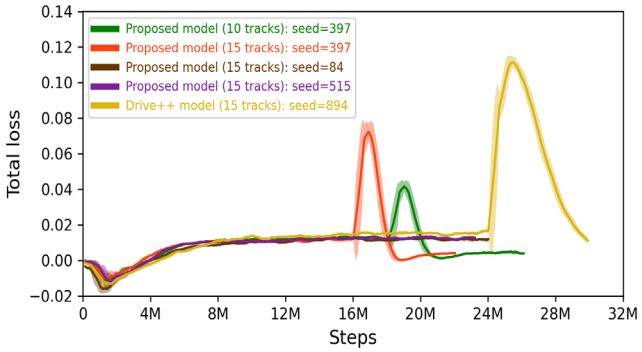


Fig. 5. The total loss of different models. The green, orange, brown, and purple lines are proposed models trained by different seeds and different numbers of tracks. In comparison, the gold line indicates the drive++ model trained by 15 tracks and a random seed.

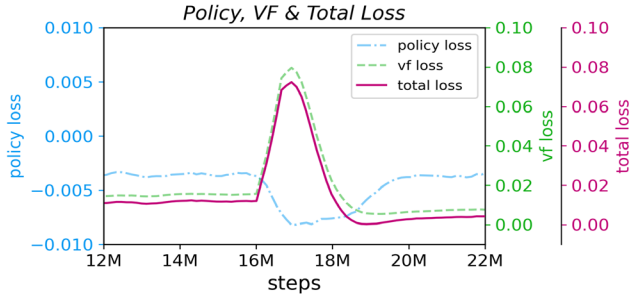


Fig. 6. Variations in policy loss, VF loss, and total loss around the peak area.

corresponding rewards. Given the robustness of the proposed approach across various seeds, random seed 397 is selected for the first-stage checkpoint training. This checkpoint forms the foundation for subsequent fine-tuning stages of the final model. TABLE III summarises the training timesteps necessary for different models.

As shown in Fig. 5 and TABLE III, the total loss of the proposed strategy and the drive++ model exhibits considerable fluctuation at the beginning of the fine-tuning phase. This is attributable to the DRL agent re-adjusting network weights to explore a new optimal policy after adding steering penalty and lane-change reward (i.e., the proposed model). This process instigates significant oscillation in policy loss. Moreover, the supplemental rewards bring a distinct deviation between the expected returns under two adjacent observation states, inflating the value function (VF) loss. The variations in policy loss and VF loss at this stage are observable in Fig. 6, leading to a peak in total loss according to (14). Nevertheless, the total loss of the proposed model declines rapidly and stabilises at a small value as training progresses. Because the proposed co-pilot strategy can provide flexible guidance to the agent by holistically assessing traffic conditions and the optimal timing for lane changes, which precludes negative lane-change returns and maintains driving safety. Furthermore, driven by the co-pilot strategy, an intelligent policy can be swiftly explored. Conversely, the drive++ model relies on a rudimentary map indicating the surrounding SVs. This necessitates a protracted learning time for the assimilation of useful information to identify the final policy, leading to a higher number of training steps required for convergence.

TABLE III
TRAINING TIMESTEP DISTRIBUTION OF DIFFERENT MODELS

STRATEGIES	PROPOSED MODELS		THE DRIVE++
	10 tracks (timesteps)	*15 tracks* (timesteps)	*15 tracks* (timesteps)
First-stage training	~18 million	~16 million	~24 million
Fine-tuning	~8 million	~6 million	~6 million
Total training	~26 million	~22 million	~30 million

TABLE IV
AVERAGE COLLISION RATE / COMPLETION RATE OF THREE MODELS ON TASK-ORIENTED TASKS IN THREE SCENARIOS

STRATEGIES	PROPOSED MODELS		THE DRIVE++
	10 tracks	*15 tracks*	*15 tracks*
Single-lane track	**0.02/0.98**	0.03/0.97	0.38/0.62
Dual-lane track	0.09/0.91	**0.04/0.96**	0.18/0.82
Triple-lane track	0.12/0.88	**0.07/0.93**	0.19/0.81

As evidenced by TABLE III, the proposed methodology converges to the optimal policies at 22 million and 26 million timesteps, trained on 15-track and 10-track maps, respectively. Significantly, the learning speed increased by 26.67% compared to the drive++ method when the training resources are adequate, e.g., on 15-track maps. Even under more resource-constrained conditions, such as on 10-track maps, the proposed method still achieves a faster learning speed by 13.33%. Additionally, total loss in the proposed model stabilises at a smaller value, indicating a reduction in model uncertainty and an increase in decision-making precision. Furthermore, the proposed methodology demonstrates a significant reduction in learning time given the same computational resources and better convergence performance even when training scenarios are limited.

B. Driving Intelligence, Safety, and Reliability

In this section, seven distinct one-way motorway tracks are designed for validation, featuring straight lanes and complex curves, including U-, S-, hairpin-, and winding sharp curves. Each scenario includes a varying number of SVs. To further assess the robustness of the proposed decision-making strategy, 30 non-training seeds are randomly selected from a range of 0 to 1,000. For each track, these seeds are tested across ten episodes, with each episode set to a maximum timestep length of 1,000. Subsequently, the proposed approach is compared with the drive++ method using performance metrics of (i) *Completion rate*, (ii) *Collision rate*, (iii) *Driving distance*, and (iv) *Driving score*, as delineated in [41] and [42].

TABLE IV depicts the agent population's average collision rate and completion rate across 600, 600, and 1,200 episodes in three scenarios with various background traffic settings. It can be discerned from TABLE IV that both of proposed models outperform the drive++ model, particularly under the single-lane track. In this context, the ego vehicle persists in car-following without lane change. However, the co-pilot strategy still provides acceleration and steering instructions for retaining in the lane centre. Additionally, the rule-based policy is capable of identifying and substituting the DRL uncertainty actions, e.g., possible off-road driving, thereby guaranteeing safe driving. Moreover, the DRL agent tends to follow the

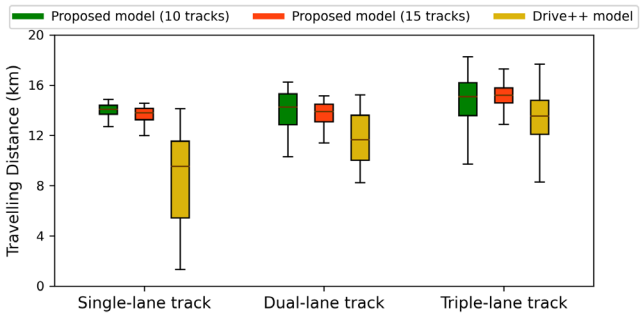


Fig. 7. The travelling distance of three models under three scenarios.

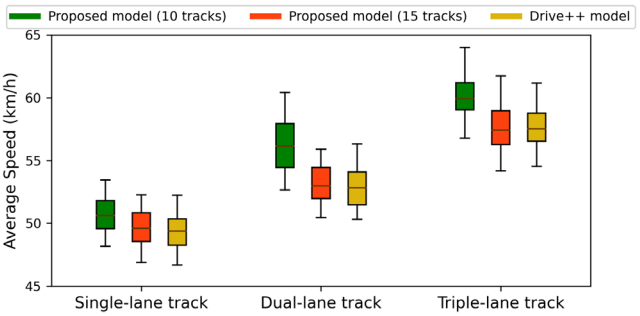


Fig. 8. The average speed of three models under three scenarios.

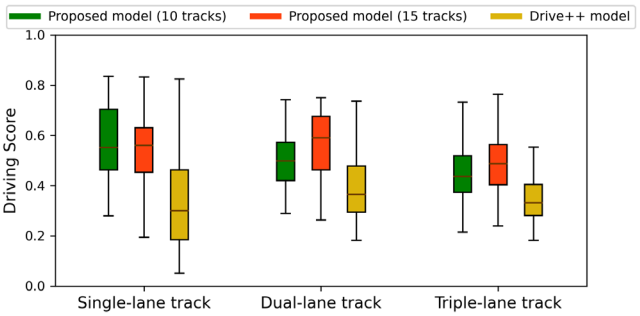


Fig. 9. The driving score of three models under three scenarios.

leading SV closely due to its greedy propensity to maximise driving distance return. For instance, the ego vehicle may aggressively accelerate to approach the front SV or car-following at a distance lesser than the safe threshold. In such instances, the rule-based policy modulates the acceleration and leaves a safe gap from the leading SV. In contrast, the drive++ model cannot effectively regulate these driving behaviours as its low-level control relies solely on DRL actions. This results in a higher average collision rate than the proposal. Furthermore, the proposed models' completion rate outperforms the drive++ model under the dual-lane and triple-lane track scenarios. This efficiency stems from the proposed method's capability to predict imminent traffic changes accurately and provide the ego vehicle with effective navigation—primarily for lane-change guidance. With the assistance of the rule-based policy, the proposed agent can navigate efficiently and safely under intricate environments with high traffic flows, e.g., fifty-five SVs involved in the test track. Compared to the drive++ agent without precise navigation, it needs to estimate the accurate lane-change timing from various observations containing irrelevant information. However, errors in the time window may be substantial in certain dynamic situations, increasing the likelihood of collisions. Additionally, the decision-making performance of the 15-track model surpasses the 10-track model due to enhanced training scenarios. However, the 10-track model can still achieve a more commendable average completion rate than the drive++ model.

Fig. 7 illustrates the driving distances achieved by three models under three scenarios, evaluated using 30 non-training seeds. The results demonstrate that the two proposed models perform comparably and outperform the drive++ model in terms of higher mileage and reduced variance. This improved performance is attributed to the robustness and low uncertainty of the proposed strategy, which enables safe and efficient driving in complex conditions characterised by diverse random behaviours. The strategy integrates an explainable kinematic model with an opaque DRL algorithm, providing precise discrete risk forecasts based on predefined rules. These discrete events are then tackled through the DRL's powerful learning capabilities, enhancing decision-making success rate and ensuring seamless action transitions between events. Additionally, the integration of the rule-based policy enhances the strategy's potential to adapt to various unknown and changing environments, significantly improving reliability and adaptability. Conversely, the drive++ model, lacking specific guidance and constraints on DRL outputs, exhibits considerable variability in decision-making performance and achieves shorter driving distances in dynamic conditions.

Fig. 8 presents the average speeds of three models under three scenarios. The analysis reveals that the 10-track model achieves higher speeds compared to the 15-track and drive++ models, whose average speeds are comparable. The 10-track model, limited by fewer training resources, exhibits more competitive driving behaviours. This ego vehicle prefers driving at a higher speed to maximise cumulative rewards. The 10-track model's competitive driving behaviour, however, leads to a higher collision rate and greater variability in driving mileage across multi-way scenarios than the 15-track and drive++ models. Despite similar average speeds achieved by 15-track and drive++ models, the drive++ model shows substantial variation in driving mileage across different seeds, reflecting its lower decision-making accuracy.

We further evaluate the proposal's comprehensive decision-making performance across various seeds using the driving score metric [42]. The definition of this metric is as follows:

$$\text{Driving Score} = CR \cdot IP \tag{23}$$

where CR represents the task completion ratio, while IP indicates the infraction penalty, which encompasses only the penalties for deviation from the lane centre and driving on the road edge. Other types of crashes are excluded from this penalty as they result in shutdown events.

As depicted in Fig. 9, both proposed models achieve higher driving scores than the drive++ model across various scenarios. This improvement is attributed not only to penalties

for deviation from the lane centreline and unnecessary steering changes but also to the implementation of real-time directional feedback that aligns the DRL agent with the lane centre during lane-keeping manoeuvres. Consequently, this approach effectively curtails unwarranted lane changes in multi-lane scenarios, thus minimising IP values and enhancing the CR bonus. Despite the drive++ model limiting deviations from the centreline and reducing steering changes, the agent continues to adopt risky driving practices when the potential gain from increased distance outweighs these penalties. Specifically, the drive++ agent is predisposed to maintaining proximity to the lane edges and actively seeking lane-changing opportunities, thereby heightening the risk of collisions. In single-lane scenarios, the proposed strategy dictates lane-following manoeuvres even when there is no need for changing lanes, enabling the ego vehicle to attain high driving scores. Conversely, the drive++ model's emphasis on competitive driving compromises its completion rate. Furthermore, the proposed strategy equips the ego vehicle with accurate guidance for car-following and lane changes across multi-lane scenarios, leading to superior driving scores compared to the drive++ model. Additionally, the driving performance of the 15-track model surpasses that of the 10-track model due to its less competitive driving behaviour, which minimises unnecessary lane changes and enhances CR rewards. While additional training resources could further improve decision-making accuracy, the disparity between the 10-track and 15-track models remains less pronounced than that between the 10-track and the drive++ models. Consequently, the proposed framework can achieve superior driving decisions with fewer training resources.

In summary, the co-pilot strategy substantially boosts training efficiency. Its efficient navigation facilitates intelligent lane changes and mitigates prolonged car-following, thereby enhancing decision-making intelligence. Cooperation between the co-pilot and rule-based strategies also significantly improves safety and reliability, resulting in extended cumulative travel distances and elevated driving scores. Additionally, the mean inference time for the proposed framework is 0.5 ms, compared to 0.48 ms for drive++, demonstrating that the additional computational demands of the co-pilot and rule-based strategy are negligible.

VI. Conclusion

In this paper, we introduce an innovative hybrid DRL-kinematic-based autopilot framework, designed to facilitate AV driving decisions across varying motorway environments. This framework integrates a kinematic-based co-pilot strategy, a DRL algorithm, and a rule-based system, thus enabling an AV to learn and update a reliable decision-making model with reduced training data. Notably, the co-pilot strategy provides traffic predictions and optimal path guidance, which supports the adoption of DRL for decision-making model learning and intelligent driving. Subsequently, task-oriented rewards are explored and optimised, compelling the DRL agent to adhere to safety instructions to the utmost degree. Furthermore, we implemented a rule-based policy to refine the final action outputs and further enhance safety in diverse driving scenes. The extensive simulation results demonstrate superior driving decision performance from the proposed framework compared to the baseline model in aspects of training efficiency, intelligence, safety, and reliability.

In future work, we aim to adapt this framework to practical AVs and further evaluate the decision-making performance in real-world driving scenarios.

References

[1] R. Zuraida, H. Iridiastadi and I. Z. Sutalaksana, "Indonesian Drivers' Characteristics Associated with Road Accidents," *International Journal of Technology,* vol. 8, no. 2, pp. 311-319, 2017.

[2] J. Liao, T. Liu, X. Tang, X. Mu, B. Huang, and D. Cao, "Decision-Making Strategy on Highway for Autonomous Vehicles Using Deep Reinforcement Learning," *IEEE Access,* vol. 8, pp. 177804-177814, 2020.

[3] National Highway Traffic Safety Administration. (2017). *Automated Driving Systems 2.0: A Vision for Safety.* [Online]. Available: https://www.nhtsa.gov/manufacturers/automated-driving-systems

[4] Y. Lu, H. Ma, E. Smart, and H. Yu, "Real-Time Performance-Focused Localization Techniques for Autonomous Vehicle: A Review," *Ieee T. Intell. Transp.,* vol. 23, no. 7, pp. 6082-6100, 2022.

[5] S. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. Eng, D. Rus, and M. Ang, "Perception, Planning, Control, and Coordination for Autonomous Vehicles," *Machines,* vol. 5, no. 1, p. 6, 2017.

[6] J. Leonard, J. How, S. Teller, M. Berger, S. Campbell, G. Fiore, L. Fletcher, E. Frazzoli, A. Huang, S. Karaman, O. Koch, Y. Kuwata, D. Moore, E. Olson, S. Peters, J. Teo, R. Truax, M. Walter, D. Barrett, A. Epstein, K. Maheloni, K. Moyer, T. Jones, R. Buckley, M. Antone, R. Galejs, S. Krishnamurthy, and J. Williams, "A perception-driven autonomous urban vehicle," *J. Field Robot.,* vol. 25, no. 10, pp. 727-774, 2008.

[7] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller, E. Kaus, R. G. Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M. Enzweiler, C. Kno¨ppel, J. Hipp, M. Haueis, M. Trepte, C. Brenk, A. Tamke, M. Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, and E. Zeeb, "Making Bertha Drive-An Autonomous Journey on a Historic Route,". vol. 6 New York, NY: IEEE, 2014, pp. 8-20.

[8] M. Gao and M. Zhou, "Control Strategy Selection for Autonomous Vehicles in a Dynamic Environment," in *2005 IEEE International Conference on Systems, Man and Cybernetics* Waikoloa, HI, USA, 2005.

[9] H. Fan, F. Zhu, C. Liu, L. Zhang, L. Zhuang, D. Li, W. Zhu, J. Hu, H. Li, and Q. Kong, "Baidu Apollo EM Motion Planner," 2018.

[10] J. Xu, Q. Luo, K. Xu, X. Xiao, S. Yu, J. Hu, J. Miao, and J. Wang, "An Automated Learning-Based Procedure for Large-scale Vehicle Dynamics Modeling on Baidu Apollo Platform,", 2019, pp. 5049-5056.

[11] Q. Liu, X. Li, S. Yuan, and Z. Li, "Decision-Making Technology for Autonomous Vehicles: Learning-Based Methods, Applications and Future Outlook," in *IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021.

[12] S. M. Grigorescu, B. Trasnea, L. Marina, A. Vasilcoi, and T. Cocias, "NeuroTrajectory: A Neuroevolutionary Approach to Local State Trajectory Learning for Autonomous Vehicles," *IEEE Robotics and Automation Letters,* vol. 4, no. 4, pp. 3441-3448, 2019.

[13] Z. Cao, D. Yang, S. Xu, H. Peng, B. Li, S. Feng, and D. Zhao, "Highway Exiting Planner for Automated Vehicles Using Reinforcement Learning," *Ieee T. Intell. Transp.,* vol. 22, no. 2, pp. 990-1000, 2021.

[14] Z. Cao, S. Xu, H. Peng, D. Yang, and R. Zidek, "Confidence-Aware Reinforcement Learning for Self-Driving Cars," *Ieee T. Intell. Transp.,* pp. 1-12, 2021.

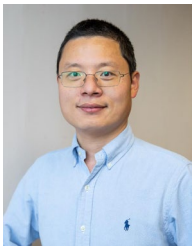
[15] Y. Fu, C. Li, F. R. Yu, T. H. Luan, and Y. Zhang, "Hybrid Autonomous Driving Guidance Strategy Combining Deep Reinforcement Learning and Expert System," *Ieee T. Intell. Transp.,* pp. 1-14, 2021.

[16] H. B. Suay and S. Chernova, "Effect of human guidance and state space size on Interactive Reinforcement Learning," in *2011 Ro-Man*, 2011, pp. 1-6.

[17] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S.

Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature,* vol. 518, no. 7540, pp. 529-533, 2015.
[18] H. Chae, C. M. Kang, B. Kim, J. Kim, C. C. Chung, and J. W. Choi, "Autonomous Braking System via Deep Reinforcement Learning," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* Yokohama, Japan, 2017.
[19] Y. Fu, C. Li, F. R. Yu, T. H. Luan, and Y. Zhang, "A Decision-Making Strategy for Vehicle Autonomous Braking in Emergency via Deep Reinforcement Learning," *Ieee T. Veh. Technol.,* vol. 69, no. 6, pp. 5876-5888, 2020.
[20] F. Hart, O. Okhrin and M. Treiber, "Formulation and validation of a car-following model based on deep reinforcement learning," 2021.
[21] D. Li and O. Okhrin, "DDPG car-following model with real-world human driving experience in CARLA," 2021.
[22] D. Zhao, B. Wang and D. Liu, "A supervised Actor–Critic approach for adaptive cruise control," *Soft Comput.,* vol. 17, no. 11, pp. 2089-2099, 2013.
[23] C. Desjardins and B. Chaib-draa, "Cooperative Adaptive Cruise Control: A Reinforcement Learning Approach," *Ieee T. Intell. Transp.,* vol. 12, no. 4, pp. 1248-1260, 2011.
[24] S. Wei, Y. Zou, T. Zhang, X. Zhang, and W. Wang, "Design and Experimental Validation of a Cooperative Adaptive Cruise Control System Based on Supervised Reinforcement Learning," *Applied Sciences,* vol. 8, no. 7, p. 1014, 2018.
[25] G. Wang, J. Hu, Z. Li, and L. Li, "Harmonious Lane Changing via Deep Reinforcement Learning," *Ieee T. Intell. Transp.,* pp. 1-9, 2021.
[26] C. Ryan, F. Murphy and M. Mullins, "End-to-End Autonomous Driving Risk Analysis: A Behavioural Anomaly Detection Approach," *Ieee T. Intell. Transp.,* vol. 22, no. 3, pp. 1650-1662, 2021.
[27] L. P. Kaelbling, M. L. Littman and A. W. Moore, "Reinforcement Learning: A Survey," *J. Artif. Intell. Res.,* vol. 4, pp. 237-285, 1996.
[28] E. Wong. (2021). *State-space decomposition for Reinforcement Learning*. [Online]. Available: https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/2021-ug-projects/State-space-decomposition-for-Reinforcement-Learning.pdf
[29] N. Ferns, P. S. Castro, D. Precup, and P. Panangaden, "Methods for Computing State Similarity in Markov Decision Processes," *arXiv:1206.6836,* 2012.
[30] M. Asadi and M. Huber, "State Space Reduction For Hierarchical Reinforcement Learning," in *FLAIRS Conference*, 2004, pp. pp. 509-514.
[31] W. Curran, T. Brys, D. Aha, M. Taylor, and W. D. Smart, "Dimensionality Reduced Reinforcement Learning for Assistive Robots," in *The 2016 AAAI Fall Symposium Series*, 2016.
[32] W. Curran, T. Brys, M. Taylor, and W. Smart, "Using PCA to Efficiently Represent State Spaces," 2015.
[33] Y. Fu, Q. Liu, X. Ling, and Z. Cui, "A Reward Optimization Method Based on Action Subrewards in Hierarchical Reinforcement Learning," *The Scientific World Journal,* vol. 2014, pp. 1-6, 2014.
[34] M. Bouton, A. Nakhaei, K. Fujimura, and M. J. Kochenderfer, "Safe Reinforcement Learning with Scene Decomposition for Navigating Complex Urban Environments,", 2019, pp. 1469-1476.
[35] C. Hoel, K. Driggs-Campbell, K. Wolff, L. Laine, and M. J. Kochenderfer, "Combining Planning and Deep Reinforcement Learning in Tactical Decision Making for Autonomous Driving," *IEEE Transactions on Intelligent Vehicles,* vol. 5, no. 2, pp. 294-305, 2020.
[36] L. Chen, X. Hu, B. Tang, and Y. Cheng, "Conditional DQN-Based Motion Planning With Fuzzy Logic for Autonomous Driving," *Ieee T. Intell. Transp.,* pp. 1-12, 2020.
[37] Z. El Abidine Kherroubi, S. Aknine and R. Bacha, "Novel Decision-Making Strategy for Connected and Autonomous Vehicles in Highway On-Ramp Merging," *Ieee T. Intell. Transp.,* pp. 1-13, 2021.
[38] M. Zhou, J. Luo, J. Villella, Y. Yang, D. Rusu, J. Miao, W. Zhang, M. Alban, I. Fadakar, Z. Chen, A. C. Huang, Y. Wen, K. Hassanzadeh, D. Graves, D. Chen, Z. Zhu, N. Nguyen, M. Elsayed, K. Shao, S. Ahilan, B. Zhang, J. Wu, Z. Fu, K. Rezaee, P. Yadmellat, M. Rohani, N. P. Nieves, Y. Ni, S. Banijamali, A. C. Rivers, Z. Tian, D. Palenicek, H. B. Ammar, H. Zhang, W. Liu, J. Hao, and J. Wang, "SMARTS: Scalable Multi-Agent Reinforcement Learning Training School for Autonomous Driving," 2020.
[39] S. John, W. Filip, D. Prafulla, R. Alec, and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv preprint,* 2017.
[40] P. Castro, B. Kolbeinsson and J. Sun. (2020). *Drive++ A Safe Autonomous Driving Algorithm*. [Online]. Available: https://github.com/PedroCastro/DriveML
[41] J. Bernhard, K. Esterle, P. Hart, and T. Kessler, "BARK: Open Behavior Benchmarking in Multi-Agent Environments,", Ithaca, 2020, pp. 6201-6208.
[42] The CARLA Team. (2022). *CARLA Autonomous Driving Leaderboard*. [Online]. Available: https://leaderboard.carla.org/

Yongqiang Lu received the B.S. degree in thermal energy and power engineering from Guangxi University, China, in 2013 and the M.S. degree in power machinery and engineering from Tianjin University, China, in 2017. He is currently pursuing a PhD degree in the Faculty of Technology, University of Portsmouth, U.K., with a focus on AV perception and decision-making using data fusion techniques and DRL. His current research interests also include AI, ML/DL, CV, and data mining.



Hongjie Ma received double B.S. degrees in thermal energy and power engineering and computer science and technology in 2009, and the M.S. and PhD degrees in power machinery and engineering from Tianjin University in 2015. He is currently serving as a Senior Research Fellow in the Innovative Industrial Research group, University of Portsmouth. With over a decade of experience, his research focuses on AI-based diagnostics, optimisation and embedded system development.

Edward Smart (M'11) received the M.Math. degree from University of Reading, Berkshire, U.K., in 2005 and the Ph.D. degree from University of Portsmouth, Portsmouth, U.K., in 2011. He was a Software Engineer with Clearswift, applying artificial intelligence to image analysis. He was also a Statistician with Flight Data Services Ltd., Hampshire, U.K. He is currently an Associate Professor with University of Portsmouth. His research interests include machine learning and industrial applications. He has been a member of the Institute of Mathematics and its Applications for seventeen years.

Hui Yu (Senior Member, IEEE) is a Professor of Visual and Cognitive Computing at the University of Glasgow, UK and leads the Visual Computing and Social Robot Group. His research interests lie in visual and cognitive computing as well as machine learning with applications to social signal analysis, social robot, human-machine interaction, intelligent vehicle, and video analysis. He has been awarded the Industrial Fellowship by the Royal Academy of Engineering. He serves as an Associate Editor for IEEE Transactions on Human-Machine Systems, IEEE Transactions on Intelligent Vehicles, and IEEE/CAA Journal of Automatica Sinica.