A Hybrid Deep Reinforcement Learning For Autonomous Vehicles Smart-Platooning

Sahaya Beni Prathiba, *Student Member, IEEE*, Gunasekaran Raja, *Senior Member, IEEE*, Kapal Dev, *Member, IEEE*, Neeraj Kumar, *Senior Member, IEEE*, and Mohsen Guizani, *Fellow, IEEE*

***Abstract*—The development of Autonomous Vehicles (AVs) envisions the promising technology of future Intelligent Transportation Systems (ITS). However, the complex road structures and increased vehicles cause traffic congestion and road safety, which eventually leads to horrible accidents. Cooperative driving of AVs, a groundbreaking initiative of vehicle platooning, epitomizes the next wave in vehicular technology through minimizing accident risks, transport times, costs, energy, and fuel consumption. However, the traditional machine learning-based platooning approaches fail to regulate the policy with the dynamic feature of AVs. This paper proposes a hybrid Deep Reinforcement learning and Genetic algorithm for Smart-Platooning (DRG-SP) the AVs. The leverage of the deep reinforcement learning mechanism addresses the computational complexity and accommodates the high dynamic platoon environments. Adopting the Genetic Algorithm in Deep Reinforcement learning overcomes the slow convergence problem and offers long-term performance. The simulation results reveal that the Smart-Platooning effectively forms and maintains the platoons by minimizing traffic congestion and fuel consumption.**

***Index Terms*—Autonomous vehicles platooning, traffic congestion, deep reinforcement learning, genetic algorithm, fuel economy.**

I. Introduction

Autonomous Vehicles (AVs), the major component of the Intelligent Transportation System (ITS), have moved from the realm of science fiction to a genuine possibility in this decade [1], [2]. Efficient and coordinated traffic management systems would hit a high spot in ITS regarding safety, fuel consumption, and traffic congestion [3], [4]. AV platooning is one of the most promising catalysts in the ITS services for increasing the road capacity and thereby averting traffic congestion [3], [5]. Platooning refers to car convoys and road trains, a coordinated group of connected vehicles that operate together to maintain a desired distance and speed.

Platooning the AVs enrolls numerous benefits [4]. The first and foremost benefit is that the platoon allows the AVs to move at a minimal distance. This emancipates previously engaged road spaces, which in turn minimizes traffic congestion. The second benefit is that driving in a platoon pattern minimizes fuel consumption considerably [6], [7]. Third, the platoon offers safer and more comfortable travel for passengers with advanced technologies. However, the unavoidable uncertainty in traditional technology causes latency in vehicular communication [8]. The 5th Generation Vehicle-to-Everything (5G-V2X) enables communication among vehicles, infrastructures, and other components via 5G technology with low latency [9], [10].

The characterization of the realistic environment and the coordinated management of platoons requires an intelligent solution. However, existing machine learning-based solutions consider one-shot optimization without maximizing the platoon features [11]–[14]. In such cases, the online deep reinforcement learning approach helps to achieve an optimal global maximum solution by maximizing the performance [15]. However, the approach's performance is immensely troubled by the slower convergence rate at the exploration stage.

This paper proposes a hybrid Deep Reinforcement learning and Genetic algorithm for Smart-Platooning (DRG-SP). The deep reinforcement learning approach in the DRG-SP algorithm performs online learning to observe the environment and maximize the optimal policy. Further, the adapted Genetic Algorithm (GA) accelerates the exploration stage of the deep reinforcement learning approach. Thus, the DRG-SP algorithm controls the decision-making strategy of the captain AV, which is the first AV in each Smart-Platoon that acts as a leader and takes decisions over creating and managing the Smart-Platoon structure. Based on the captain AV decisions, the Smart-Platoons are formed, controlled, and organized.

The key contributions are as follows,

1) To effectively form platoons, the DRG-SP algorithm is proposed. The deep reinforcement learning approach explores the high dimensional dynamic environments and results in effective Smart-Platooning.
2) The DRG-SP algorithm regularly updates the policy through a rank-based replay memory. Then the captain AV

Manuscript received August 28, 2020; revised December 30, 2020, May 22, 2021, July 24, 2021, and October 6, 2021; accepted October 7, 2021. Date of publication October 26, 2021; date of current version December 17, 2021. The work of Sahaya Beni Prathiba and Gunasekaran Raja was supported by the NGNLab, Department of Computer Technology, Anna University, Chennai, India. The review of this article was coordinated by Prof. Moussa Boukhnifer. *(Corresponding author: Sahaya Beni Prathiba.)*

Sahaya Beni Prathiba and Gunasekaran Raja are with the NGNLab, Department of Computer Technology, Anna University, Chennai 600025, India (e-mail: sahayabeni@ieee.org; dr.r.gunasekaran@ieee.org).

Kapal Dev is with the Department of Institute of Intelligent Systems, University of Johannesburg, Auckland Park 2006, South Africa (e-mail: kapal.dev@ieee.org).

Neeraj Kumar is with the Thapar Institute of Engineering and Technology, Patiala 147004, India, with the Department of Computer Science and Information Engineering, Asia University, Taichung city 40704, Taiwan, and also with the School of Computer Science, University of Petroleum and Energy Studies, Dehradun 248007, Uttarakhand, India (e-mail: neeraj.kumar@thapar.edu).

Mohsen Guizani is with the Qatar University, Doha 122104, Qatar (e-mail: mguizani@ieee.org).

Digital Object Identifier 10.1109/TVT.2021.3122257

is aided to make highly optimal decisions, which reflects on the Smart-Platoon in a coordinated manner.

3) The Smart-Platoon features such as the dynamics, balanced state, Inter-Platoon Distance (IPD), Inter-Vehicle Distance (IVD), and size of the platoon are formulated.

The rest of the paper is organized as follows. Section II studies the literature on platoon mechanisms. Following this, Section III describes the system model of Smart-Platoon architecture. The proposed DRG-SP algorithm for Smart-Platooning the AVs is then described in Section IV. Section V delivers the features of the Smart-Platoon. Section VI covers the simulation setup, obtained results, and its corresponding discussion. The conclusion of this work is given in Section VII.

II. Related Work

Numerous works examine the importance of vehicle platooning [16], via utilizing 5G-V2X communication [17], and incorporating intelligent techniques [18]. A platoon merging approach is introduced in [19] for reducing fuel consumption using a distributed model predictive control algorithm. The algorithm initially creates spaces for the second platoon to merge with the first platoon and later fills the spaces with the AVs of the second platoon. A platooning approach in [20] utilizes distributed control protocols for effectively reducing fuel consumption and greenhouse gas emissions. Like [21], considering the IVD and velocity, the longitudinal controllers are designed in the four-layer framework to deal with multiple platoon strings [22].

A cooperative spacing control in [23] forms a platoon without any communication between the vehicles. The sensational onboard measurements are the reason behind the spacing control strategy. Similarly, in [24], a cooperative power split-up method is proposed, utilizing the multiple and dynamic populations generating particle swarm optimization algorithm to group the electric vehicles. A Composite Platoon Trajectory Planning Strategy (CPTPS) proposed in [25], develops a trajectory plan for the leader vehicles and applies reinforcement learning mechanism for the followers. Further, CPTPS introduces a flexible platoon management protocol for managing the platoons efficiently. In addition to the platoon controlling strategies, potential communication plays a significant role in effective platooning.

The 5G-V2X connectivity for AVs enhances the reliability of the platoons. In [26], the 5G-V2X enabled vehicles collect the perceptions of the surroundings as crowdsensing to improve safety, save energy, and optimize the traffic. In [13], a cooperative autonomous driving strategy is proposed in Mobile Edge Computing enabled 5G-V2X environment. The system adopts the permutation operator for the deep learning technique to reduce the implementation cost and solve complex optimization issues. Thus, to adapt to the dynamic AV environment, intelligent techniques were introduced [27]. The Convolutional Neural Networks in [12] collects and shares the videos of extensive infrastructure features captured by the sensors of the AV with the other AVs in the platoon. The Q-network reinforcement learning technique in [28] finds the optimal locations at which the base stations can be fixed to provide better platoon features to the AVs. In [29], a path-planning scheme is presented via the Deep Reinforcement learning method (ϵ-Q-learning) for AV platooning.

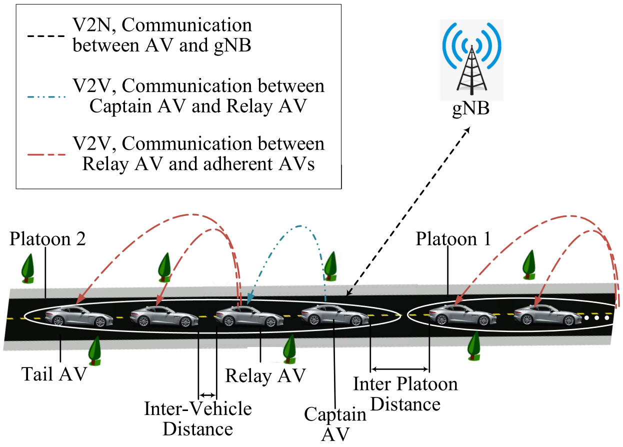

V2N, Communication between AV and gNB
V2V, Communication between Captain AV and Relay AV
V2V, Communication between Relay AV and adherent AVs
gNB
Platoon 2
Platoon 1
Tail AV
Relay AV
Inter-Vehicle Distance
Captain AV
Inter Platoon Distance

Fig. 1. Smart-Platoon Architecture.

Still, the existing schemes suffers with high complexity and high exploration cost, making the AVs unadaptable to the real-time environment. To overcome this, the deep reinforcement learning algorithm in the proposed Smart-Platooning embraces GA for regulating the policy as per the environment with improved performance. Thus, the captain AV makes intelligent decisions, reflecting on the adherent AVs and forming a high-performance Smart-Platoon.

III. System Model

In Smart-Platooning, the 5G base station gNodeB (gNB) synchronizes the platoon information to other AVs, helping them to join and acquire beneficiaries. Each AV possesses a unique ID, which authenticates the model and increases its credibility. Further, they are equipped with a Global Navigation Satellite System (GNSS) and proximal sensors to grab the AVs' position accurately. Each AV accesses the V2V (Vehicle-to-Vehicle) communication mode of 5G-V2X technology. In contrast, the captain AV of each Smart-Platoon access V2N (Vehicle-to-Network) and V2V communication modes transmitting platoon information to the gNB and platoon driving strategies to adherent AVs, respectively. Smart-Platooning uses 5G NR (New Radio)-based V2X communication and the model-free deep reinforcement learning methodology as the communication and computation model, respectively. Fig. 1 shows the architecture model of Smart-Platooning via hybrid deep learning algorithms.

The Smart-Platoon follows the typical predecessor-follower topology, in which all the adherent AVs follows the captain AV. The responsibilities of captain AV include transmission of the velocity, acceleration, and deceleration to be followed in the Smart-Platoon (SP_{i-1}). Besides, captain AV maintains an appropriate distance with the preceding individual AV or the tail AV of the preceding Smart-Platoon. Thus, the IPD (IPD_i) (distance between Smart-Platoons SP_{i-1} and SP_i) and the IVD (IVD), (distance between the two AVs AV^i_{x-1} and AV^i_x of SP_i) plays a significant role in Smart-Platooning. Since the

TABLE I
MATHEMATICAL NOTATIONS

Variable	**Explanation**
SP_i	ith Smart-Platoon
IPD_i	Inter-Platoon Distance
IVD	Inter-Vehicle Distance
IPD_i	Inter-Platoon Distance
$L(SP_i)$	Length of the Smart-Platoon
R_{min}	Minimum transmission range
V_x	Velocity of AV_x
ΔV	Change in velocity
Pos_x	Position of AV_x
IVD_x^{des}	Desired distance of AV_x
IVD_{min}	Minimum IVD between two AVs (at standstill)
b	Balanced state
acc_x	Acceleration of AV_x
IVD_b	IVD at balanced state b
acc_{max}	Maximum acceleration of an AV
T_h	Time headway
α	Damping Ratio
β	Natural Frequency of an AV
$IVD_x^{dev}(t)$	Deviation in IVD
V_μ	Critical Velocity
S_t	State of the environment
A_t	Action at time t
π	Policy of the Actor component
$\Re_t$	Reward at time t
g^μ	Parameter of Actor component
g^Q	Parameter of Critic component
$\Re_{min}$	Minimum Reward
$P\,(cr)$	Probability of crossover
$P\,(mu)$	Probability of mutation
$\mathcal{M}$	Rank-based replay memory

AVs follow the mode of V2V communication, the length of the Smart-Platoon ($L(SP_i)$) depends on the minimum transmission range ($\mathbb{R}_{min}$) of the relay AV of each SP_i. Relay AV is the intermediate AV, transmits the information obtained from the captain AV to the adherent AVs. Table I encapsulates the necessary mathematical notations.

The function of IVD, velocity (V), and change in velocity (ΔV) defines the typical car-following strategy as $f(IVD,\ V, \Delta V)$. This function estimates the acceleration of an AV in the platoon for a specific time t. Thus, the acceleration of the x-th AV (AV_x) is described as,

$$acc_x(t) = \frac{dV_x(t)}{dt} \tag{1}$$

where $V_x(t)$ is the velocity of the AV, AV_x at time t. Similarly, the function for the car-following strategy at time t is,

$$f\,(IVD,\ V, \Delta V)\,|_t = f\,(IVD_x\,(t)\,,\ V_x\,(t)\,,\ \Delta V_x\,(t))$$

such that,

$$\Delta V_x(t) = V_x(t) - V_{x-1}(t) \tag{2}$$

where $IVD_x(t)$ is the IVD of AV_x. $V_{x-1}(t)$ is the velocity of the preceding AV, AV_{x-1} at time t. To follow the captain AV, the adherent AVs should update their velocity and position, making the car-following strategy as a platoon [30]. Thus, each AV updates the velocity and position by,

$$V_x\,(t + \Delta t) = V_x(t) + acc_x(t)\Delta t$$

$$Pos_x\,(t + \Delta t) = Pos_x(t) + V_x(t)\Delta t + 1/2acc_x(t)\Delta t^2$$

where Pos_x is the position of AV_x, Δt is the renewing timestamp within which each AV should update the velocity and position (typically, $\Delta t \cong 0.1$ s).

In Smart-Platoon, the desired distance of AV_x (IVD_x^{des}) to the precedent AV is estimated by,

$$IVD_x^{des}(t) = IVD_{min} + V_x(t)T_h + \frac{V_x(t) * \Delta V_x(t)}{2\sqrt{acc_{max} * dec_{max}}} \tag{3}$$

where IVD_{min} is the minimum IVD between two AVs (at standstill), T_h is the time headway (Here, $T_h = 1.5$ s), acc_{max} is the maximum acceleration, and dec_{max} is the maximum deceleration allowed for AV_x at the time (t) respectively. Time headway is the time taken by a trailing AV to reach the position of the precedent AV without changing its speed. The acceleration of AV_x is,

$$acc_x(t) = acc_{max}\left[1 - \left(\frac{V_x(t)}{V_{max}}\right)^4 - \left(\frac{IVD_{min}(t)}{IVD_x(t)}\right)^2\right] \tag{4}$$

where $acc_x(t)$ is the acceleration of AV_x at time t, and V_{max} is the maximal velocity allowed for each AV. However, $acc_x(t)$ can vary if there is no precedent AVs. That is if AV_x does not have any AVs ahead then, $IVD_{min}(t) = 0$. In this case, the acceleration of AV_x from (4) becomes,

$$acc_x(t) = \ acc\left[1 - \left(\frac{V_x(t)}{V_{max}}\right)^4\right]$$

Similarly, the deceleration of AV_x concerning the precedent AV ($dec_x(t)$) can be described as,

$$dec_x(t) = -acc\left(\frac{IVD_{min}(t)}{IVD_x(t)}\right)^2$$

IV. Intelligent Driving Strategy Via Deep Reinforcement Learning and Genetic Algorithm for Smart-Platooning

DRG-SP algorithm equips the captain AV with an intelligent driving strategy. The objective of the deep reinforcement algorithm of DRG-SP is to improvize the long-term rewards for enhancing the driving strategy to a greater extent. However, the deep learning algorithm causes slow convergence when they act on the high dimensional action spaces. To overcome this issue, the DRG-SP algorithm relies on GA. Further, the adoption of GA scales down the computation complexity considerably.

A. State and Action Formulation

The system state of the Smart-Platooning environment for a specific time-period t is,

$$S_t^i = \left(S_{SP}^i,\ S_{AV}^i, S_c^i\right) \tag{5}$$

where S_t is the state of the environment, which holds other states such as, the state of current Smart-Platoon i (S_{SP}^i), the state of

AVs in the i-th Smart-Platoon (S_{AV}^{i}), and the current state of the environment (S_{c}^{i}).

B. Overview of Hybrid Deep Reinforcement Learning and Genetic Algorithm for Smart-Platooning (DRG-SP) Algorithm

The two important components of DRG-SP are the actor component and the critic component. The actor learns a policy by mapping S_t^i with the action A_t at a specific time t. The policy of the DRG-SP algorithm is,

$$\pi :\ S_t^i \to A_t$$

where π is the policy of the actor component.

The critic component evaluates the mapping made by the actor component towards the reward $\Re$, which is,

$$Q^{\pi} : \left(S_t^i \to A_t\right) \to \Re_t$$

The captain AV receives the request from the other AVs to join in the platoon, which is collected periodically for S_t^i in SP_i. The DRG-SP algorithm gets triggered whenever the captain AV receives a request from the other AVs and changes in the velocity of IPD for the particular Smart-Platoon. At the time of decision-making t, the actor component makes an action A_t by exploring or exploiting the Smart-Platoon environment for S_t^i. Once the captain AV makes an action, it transmits the action to the other AVs of the same platoon. In this spot, the critic component evaluates the action given by the captain AV by a reward $\Re_t$. Consequently, the actor component receives the next state S_{t+1}^i along with $\Re_t$ for the previous action made ($S_t^i \to A_t$).

The DRG-SP algorithm stores S_t^i, A_t, and $\Re_t$ along with the next state S_{t+1}^i as a single component ($S_t^i, A_t, \Re_t,\ S_{t+1}^i$) in the replay memory. The replay memory helps the actor to learn from its previous experiences, which speeds up the learning and helps to break undesirable temporal correlations. Hence, to update the parameter of the actor (g^{μ}) and the critic (g^{Q}) components, the DRG-SP algorithm selects the highly ranked samples from the replay memory. This augments the critic component to maximize the long-term rewards for the Smart-Platoon. Accordingly, the objective function of the DRG-SP algorithm is to identify the optimal policy π^*, which maximizes the reward of each decision made by the captain AV, which is as follows,

$$\pi^{*}= \max \varUpsilon_{S,\pi} \left[\sum_{t=1}^{T} \Re\left(S_{t+1}^i | S_t^i,\ \pi\left(A_t\right)\right) \right] \tag{6}$$

where π^* is the optimal policy for the Smart-Platooning environment $\varUpsilon_{S,\pi}$ with state S and policy π.

C. Genetic Algorithm in DRG-SP

The reinforcement algorithm conventionally operates by adding an entropy bonus to maximize expected rewards, thereby ensuring a comprehensive exploration. In the high dimensional action space, the AVs require long-term platooning service without reducing the performance. The existing optimization algorithms suffer from slow convergence, less accuracy, non-adaptability towards the exponential expansion of action space [31]. Therefore, the DRG-SP algorithm uses the GA algorithm for enhancing the performance. The GA in DRG-SP algorithm prompts the convergence in high dimensional action space of AV environment and thereby reduces the computational complexity vastly. Hence, the DRG-SP algorithm obtains optimal policy based on the past platooning experiences, and thus, a considerable number of unnecessary iterations is reduced. In DRG-SP, the loss function of the critic component evaluates the actor component's erroneous portion of each action taken. If the loss value obtained (ω_t) for the action A_t is higher than the threshold loss value (ω_{th}), the GA in DRG-SP guarantees sufficient exploration. The critic component's feature learning while pre-training yields ω_{th}, which is a value at which the critic component converges the most. Thus, to provide promising explorations without time and computation complexity, the DRG-SP algorithm uses GA to explore the state spaces efficiently.

The difference between average reward $\Re_{avg}$ and minimum reward $\Re_{min}$ evaluates the convergence property of crossover and mutation operators of GA [32]. This is expressed as,

$$C_{cr} = C_{mu} = \Re_{avg} - \Re_{min}$$

When the population converges to an optimal solution (either local optimum or global optimum), the convergence of crossover and mutation tends to be lower. Thus, based on the convergence, the probability of crossover and mutation has to be adjusted. If the convergence pauses in the local optimal solution, then the probability of crossover $P(cr)$ and mutation $P(mu)$ increases, else they decrease. Accordingly,

$$P\left(cr\right) = \frac{\ell_1}{\Re_{avg} -\ \Re_{min}}$$

$$P\left(mu\right) = \frac{\ell_2}{\Re_{avg} -\ \Re_{min}}$$

However, increasing $P(cr)$ and $P(mu)$ may not be suitable when the population heads towards the global maximum of the state space. To solve this issue in the DRG-SP algorithm, the value of $P(cr)$ depends on the parent's reward ($\Re_{t-1}$), meanwhile $P(mu)$ depends on its reward ($\Re_t$). Therefore, $P(cr)$ increases when the difference between the minimum reward and parent's reward ($\Re_{min} -\ \Re_{t-1}$) increases, and $P(mu)$ increases when the difference between its reward and minimum reward ($\Re_t -\ \Re_{min}$) increases. But, when the values of $P(cr)$ and $P(mu)$ exceeds the maximum probability 1,

$$P^{*}(cr) = \begin{cases} \frac{(\Re_{min} -\ \Re_{t-1}) * \ell_1}{\Re_{t-1} - \Re_{min}},\ \ \Re_{min} \geq\ \Re_{t-1} \\ \ell_3,\ \ \Re_{min} <\ \Re_{t-1} \end{cases} \tag{7}$$

$$P^{*}(mu) = \begin{cases} \frac{(\Re_{t} -\ \Re_{min}) * \ell_2}{\Re_{t-1} - \Re_{min}},\ \ \Re_{min} \leq\ \Re_{t} \\ \ell_4,\ \ \Re_{min} >\ \Re_{t} \end{cases} \tag{8}$$

where ℓ_1, ℓ_2, ℓ_3 and ℓ_4 are the random numbers whose range is from 0 to 1.

D. Determining Optimal Q-Function by Critic Component

In the DRG-SP algorithm, the policy π updates iteratively for the Smart-Platooning environment. Thus for a given state, action, reward, and the next state, the update rule for the critic component in DRG-SP is as follows,

$$\begin{aligned} Q\left(S_t, A_t\right) = {} & Q\left(S_t, A_t\right) + \varphi\left(S_t, A_t\right) \\ & \times \left[\Re_t + \theta * \max_{j \in \mathbb{A}} Q\left(S_{t+1}, j\right) \ - Q\left(S_t, A_t\right)\right] \end{aligned} \tag{9}$$

When subtracting $Q^{+}(S_t, A_t)$ on both sides of (9),

$$\Delta\left(S_t, A_t\right) = Q\left(S_t, A_t\right) - Q^{+}\left(S_t, A_t\right)$$

On deriving,

$$\Delta\left(S_t, A_t\right) = (1-\varphi) * \Delta\left(S_t, A_t\right) + \ \varphi\left(S_t, A_t\right) * f(S_t, A_t)$$

where the function $f(S_t, A_t)$ is given as,

$$f\left(S_t, A_t\right) = \left[\Re_t + \theta * \max_{j \in \mathbb{A}} Q\left(S_{t+1}, j\right)\right] - Q^{+}\left(S_t, A_t\right)$$

When t = (1,∞), $\sum_{t=1}^{\infty} \varphi = \infty$ and $\sum_{t=1}^{\infty} \varphi^2 = m$, where m is a finite number [33]. When the critic function $\Delta(S_t, A_t)$ converges to zero, the critic component of the DRG-SP algorithm determines the optimal Q-function. The criteria at which the critic function converges to zero are as follows,

1) $\|\mathcal{E}[f(S_t, A_t)|f]\|_{\infty} \leq \varphi \|\Delta(S_t, A_t)\|_{\infty}$, with $\varphi < 1$
2) var$[f(S_t, A_t)|f] \leq c(1 + \|\Delta(S_t, A_t)\|_{\infty}^{2})$, with $c > 0$

The above two criteria is derived by the following,

$$\begin{aligned} \|\mathcal{E}\left[f(S_t, A_t)|f\right]\|_{\infty} &= P\left(S_{t+1} \mid \ S_t, \ A_t\right) * \ f\left(S_t, A_t\right) \\ &\leq \varphi \left\| Q\left(S_t, A_t\right) - Q^{+}\left(S_t, A_t\right)\right\|_{\infty} \\ &\leq \varphi \|\Delta\left(S_t, A_t\right)\|_{\infty} \end{aligned}$$

$$var\left[f(S_t, A_t)|f\right] \ = var\left[\Re_t + \theta * \max_{j \in \mathbb{A}} Q\left(S_{t+1}, j\right)|f\ \right]$$

However, $\Re_t$ is delimited with the following constraint,

$$var\left[f(S_t, A_t)|f\right] \leq c\left(1 + \|\Delta\left(S_t, A_t\right)\|_{\infty}^{2}\right)$$

where c is a constant. Thus, based on the above two constraints when the value of $\Delta(S_t, A_t)$ converges to zero, the critic component of the DRG-SP algorithm converges to an optimal Q-function $Q^{+}(S,\ A)$. Accordingly, the critic component estimates the efficiency of each action in the candidate set $\mathcal{A} = \{A_t^1, A_t^2, \ldots, A_t^x\}$, and the optimal action A^{+} is,

$$A^{+} = \arg\max_{A_j \in \mathbb{A}} Q^{+}\left(S_t, \ A_j\right)$$

Algorithm 1: DRG-SP Algorithm.

```
Input: Parameters of actor ($g^{\mu}$) and critic ($g^{Q}$) component and State $S_t$ of AV environment
Output: Updated target actor ($g^{\mu'}$) and critic ($g^{Q'}$) component
Initialize:
    parameters of actor and critic $g^{\mu}$ and $g^{Q}$,
    parameter of target actor and critic components
    $g^{\mu'} \leftarrow g^{\mu}$ and $g^{Q'} \leftarrow g^{Q}$,
    fixed $\wp$-sized rank-based replay memory $\mathcal{M}$,
    $G_{max}$ as the maximum number of generations
    and $\zeta$ as the training interval
repeat
    for each state $S_t$ of the actor component do
        Obtain action $A_t$
        if probability is $1-\varepsilon$ then
            $A_t \leftarrow A_t$
        else if $L(g^{Q'}) \leq \omega$ then
            $A_t \leftarrow A^{+}$
        else if $L(g^{Q'}) > \omega$ then
            $A_t \leftarrow rand(\mathcal{A})$
        end if
        Execute action $A_t$, obtain reward $\Re_t$ and new state $S_{t+1}$
        Store $(S_t, A_t, \Re_t, S_{t+1})$ in replay memory $\mathcal{M}$
        return $\mathcal{M}$
    end for
    if $t\text{mod}\zeta \ = 0$ then
        Minimize loss value using (10) and (11)
        Select experiences from rank-based replay memory
        Update $g^{\mu}$ and $g^{Q}$ using Adam optimizer
    end if
    Update target actor component:
    $g^{\mu'} \leftarrow \Omega g^{\mu} + (1-\Omega) g^{\mu'}$
    Update target critic component:
    $g^{Q'} \leftarrow \Omega g^{Q} + (1-\Omega) g^{Q'}$
until stopping condition is satisfied
```

E. Updating Decision Making Policy With Rank-Based Replay Memory

The DRG-SP algorithm accumulates and saves the obtained experiences in a fixed size ($\wp$) replay memory. The saved experiences is expressed as $\mathcal{M} = \{M_1, M_2, \ldots, M_{\wp}\}$, where $M_i = (S_t^i, A_t, \Re_t, \ S_{t+1}^i)$. During the training phase, some sets of experiences are selected randomly in the traditional deep reinforcement algorithms from the replay memory to update the parameters of the actor and critic component. This process of updating the deep reinforcement learning boosts the progress of maximizing the reward of the network. However, selecting a random set of experiences may omit the most essential experiences. To avoid this issue, the DRG-SP algorithm sorts and ranks the experiences from the replay memory in a SumTree format based on the priority [34]. Whenever the DRG-SP algorithm trains the model, it selects the top-ranked experiences for training towards maximizing the long term reward. Thus, the rank-based replay memory improves network efficiency by gradually reducing loss. Expressing the probability of choosing the most essential experiences from the rank-based replay memory as follows,

$$P\left(M_i\right) = \frac{pr_{M_i}}{\sum_j pr_{M_j}}$$

where pr_{M_i} is the rank of the experience M_i and always $pr_{M_i} > 0$. The rank-based replay memory of actor component with a set of time indices $\mathcal{T}$ is $\mathcal{M}_{\mathcal{T}} = \{(S_t,\ A_t) \mid t \in \mathcal{T}\}$. Similarly, the expression for the rank-based replay memory of the critic component is given as $\mathcal{M}'_{\mathcal{T}} = \{(S_t,\ A_t,\ \Re_t,\ S_{t+1}) \mid |t \in \mathcal{T}\}$. To minimize the loss values, the Adam optimizer [35] updates g^{μ} and g^{Q}. Thus the loss function of the actor component ($L(g^{\mu})$) is given as,

$$L\left(g^{\mu}\right) = \mathbb{E}_{\mathcal{M}_{\mathcal{T}}}\Big[A_t^T \log\left(\mu * S_t\right) + \left(1 - A_t\right)^T \log\left(1 - \left(\mu * S_t\right)\right)\Big] \quad (10)$$

where $\mu * S_t$ is the outcome of the present actor component computed with the input state S_t. Similarly, the loss function of the critic component is expressed as,

$$L\left(g^{Q}\right) = \mathbb{E}_{\mathcal{M}'_{\mathcal{T}}}\left[\Re_t + \theta * \max_{A_{t+1}} Q\left(S_{t+1}, A_{t+1}\right) - Q\left(S_t, A_t\right)\right]^2 \quad (11)$$

where θ is the discount factor. The pseudocode of the DRG-SP algorithm is given in Algorithm 1.

V. Smart-Platoon Features

This section spells out the features of the Smart-Platoon after applying the DRG-SP algorithm to the captain AV.

A. Balanced State of Smart-Platoon

When the AVs follow the Smart-Platooning strategy, the AV driving is in a balanced state. The balanced state (b) is the state at which the AVs ($AV_1^i,\ AV_2^i,\ \ldots, AV_n^i$) of the platoon SP_i possess no acceleration from the current velocity and thus the velocity difference becomes zero.

$$\begin{cases} acc_x(t) = 0 \\ \Delta V_x(t) = 0 \\ V_x(t) = V_{x+1}(t) = V_{x-1}(t) = V_b \end{cases} \quad (12)$$

where V_b is the velocity at b, at which the AVs do not suffer from any rise or fall in velocity. The linearization of the velocity and IVD of AV_x at b show that the velocity of AV_x ($V_x(t)$) is a combination of the velocity of the precedent AV ($V_{x+1}(t) = V_b$) and the change in velocity ($\Delta V_x(t)$). This is represented as follows,

$$V_x\left(t\right) = V_b + \Delta V_x\left(t\right)$$

Correspondingly, the IVD at balanced ($IVD_x(t)$) is a combination of balanced IVD (IVD_b) and the deviation in the IVD ($IVD_x^{dev}(t)$), which is,

$$IVD_x(t) = IVD_b + IVD_x^{dev}(t) \quad (13)$$

The IVD deviation ($IVD_x^{dev}(t)$) is the negligible error value in finding the distance between two AVs. Also, when $t \to \infty$, the sum of all deviation of IVDs becomes 0 (i.e., $\sum_{x=2}^{n} IVD_x^{dev}(t) = 0$, where x denotes the number of AVs in the Smart-Platoon). This statement reveals that the length of the Smart-Platoon will be a constant at b. Besides, the second-order derivations of the linear (13) yield a conventional damping oscillator equation as follows,

$$\frac{d^2 IVD_x^{dev}(t)}{dt^2} + 2\alpha\beta \frac{dIVD_x(t)}{dt} + \beta^2 IVD_x(t) = 0 \quad (14)$$

where α is the damping ratio, which refers to the measurement of how the oscillatory movement (unequal or change in velocities) of an AV reaches a balanced state. Further, β refers to the natural frequency, which is the frequency of an AV at which it has a change in velocity even in the absence of any driving force. The following equation calculates the natural frequency (β) of an AV,

$$\beta^2 = \left.\frac{\partial f}{\partial IVD}\right|_b \quad (15)$$

The partial derivative of (15) yields a constant value called damping ratio as,

$$\alpha\beta = -\frac{1}{2}\left(\left.\frac{\partial f}{\partial v}\right|_b + \left.\frac{\partial f}{\partial \Delta v}\right|_b\right)$$

Accordingly, the partial derivatives of the car-following function is,

$$\frac{\partial f}{\partial IVD} = \frac{2 * acc_{max} * \left(IVD_x^{des}\right)^2}{IVD^3} \quad (16)$$

$$\frac{\partial f}{\partial v} = \frac{-4 * acc_{max} * V^3}{{V_{max}}^4} - \frac{2 * acc_{max} * T_h * IVD_x^{des}}{IVD^2}$$

$$\frac{\partial f}{\partial \Delta v} = \frac{-acc_{max} * V * IVD_x^{des}}{IVD^2\sqrt{V * acc_{max}}}$$

The eigenvalues of (14) are,

$$\lambda_1 = -\alpha\beta + \sqrt{\beta^2(\alpha^2 - 1)} \quad (17)$$

$$\lambda_2 = -\alpha\beta - \sqrt{\beta^2(\alpha^2 - 1)} \quad (18)$$

As a whole, the equations (17) and (18) are consolidated as $\lambda = -\alpha\beta \pm \sqrt{\beta^2(\alpha^2 - 1)}$. In consonance with Lyapunov Stability Theory [36], both α and β are positive values (i.e., $\alpha, \beta > 0$). However, depending on the positive values of damping ratio (α) three cases exist,

Case 1: When $\alpha > 1$, the eigenvalues become negative real values, and (14) is said to be an overdamped condition. In this case, $IVD_x^{dev}(t)$ slowly returns to b without overshooting from its transient state. Hence, the adherent AVs of AV_x do not overshoot the velocities to cover the IVD, rather they accelerate and decelerate monotonically.

Case 2: When $0 < \alpha < 1$, the eigenvalues become imaginary values, and (14) is said to be the underdamped condition. In this case, $IVD_x^{dev}(t)$ overshoots in attempting to reach b. Hence, the adherent AVs of AV_x experience more changes in the acceleration and deceleration ahead of a balanced state.

Case 3: When $\alpha = 1$, the eigenvalues become negative real values, and (14) is said to be a critically damped condition. In this case, $IVD_x^{dev}(t)$ reaches b as early as possible (i.e., with the minimal time) without changing the velocity of AV_x. The velocity in this case is defined as the critical velocity (V_μ). At V_μ, the partial derivative of the car-following function with respect to the IVD at velocity V_μ is,

$$\left.\frac{\partial f}{\partial IVD}\right|_{V_\mu} = \frac{1}{4}\left(\left.\frac{\partial f}{\partial v}\right|_{V_\mu} + \left.\frac{\partial f}{\partial \Delta v}\right|_{V_\mu}\right)^2 \tag{19}$$

By substituting (16) in (19), the IVD of AV_x ($IVD_x(t)$) is,

$$IVD_x(t) = \frac{IVD_{min} + V_x(t) * T_h + \left(\frac{V_x(t) * \Delta V_x(t)}{2 * \sqrt{acc_{max} * dec_{max}}}\right)}{\sqrt{1 - \left(\frac{V_x(t)}{V_{max}}\right)^4 - \frac{acc_x(t)}{acc_{max}}}} \tag{20}$$

However by (12), $acc_x(t) = 0$, $\Delta V_x(t) = 0$, and $V_x(t) = V_b$ at b. Hence, the substitution of the values of (12) in (20) produces the balanced IVD between two AVs in the Smart-Platoon. The balanced IVD (IVD_b) is as follows,

$$IVD_b = \frac{IVD_{min} + V_b * T_h}{\sqrt{1 - \left(\frac{V_{eq}}{V_{max}}\right)^4}} \tag{21}$$

Thus, at b, $IVD_x(t)$ monotonically increases concerning $V_x(t)$ of AV_x.

B. Platoon Size

This section delivers the estimation of the maximum allowable number of AVs in a Smart-Platoon. The SP_i consists of one captain AV (AV_1^i), one tail AV (AV_n^i) and one relay AV (AV_r^i), which is in the center of the platoon. Hence, based on the position of the relay AV (r), the size is,

$$n = (2 * r) - 1 \tag{22}$$

The relay AV (AV_r^i) is responsible for transmitting the information passed by the captain AV to all the adherent AVs. Hence, the Smart-Platoon size (the number of AVs in the platoon) is based on $\mathbb{R}_{min}$ of V2V communication in the 5G-V2X technology. Accordingly, the distance between the relay AV (AV_r^i) and every other AV ($\{AV_1^i, AV_2^i, \ldots, AV_n^i\}$) in the Smart-Platoon should be less than $\mathbb{R}_{min}$. The following equation represents the distance between the captain AV and the relay AV.

$$D(1, r) = r * L(AV) + \sum_{x=2}^{r} IVD_b + \sum_{x=2}^{r} IVD_x(t) \leq \mathbb{R}_{min} \tag{23}$$

where $D(1, r)$ is the distance between the captain AV and the relay AV, r is typically $\lfloor (n+1)/2 \rfloor$, and $L(AV)$ is the length of an AV. In this work, all the AVs are considered to have similar lengths. Similarly, the distance between the relay AV and tail AV is given by,

$$\begin{aligned} D(r, n) = &\ (n - r) * L(AV) \\ &+ \sum_{x=r+1}^{n} IVD_b + \sum_{x=r+1}^{n} IVD_x(t) \leq \mathbb{R}_{min} \end{aligned}$$

where $D(r, n)$ is the distance between the relay AV and tail AV (the AV at the end of the platoon). Thus, based on $\mathbb{R}_{min}$ the size of the Smart-Platoon has decided accordingly. V_μ is another important parameter that plays a vital role in deciding the Smart-Platoon size. Considering V_μ, the size falls into three major cases. They are as follows,

Case 1: When $V_{-1} \geq V_\mu$, the velocity of the Smart-Platoon SP_i overdamps. Hence, on the ahead of reaching the balanced state, the IVD does not carry out any overshoot (i.e., $IVD_x(t) \geq 0$ at $V_b = V_0$). Now, the revised (23) is,

$$\begin{aligned} D(1, r) &= r * L(AV) + (r - 1) IVD_b \leq \mathbb{R}_{min} \\ D(r, n) &= (n - r) * L(AV) + (n - r) IVD_b \leq \mathbb{R}_{min} \end{aligned}$$

The above two equations finally derive the constraint that,

$$r \leq \lfloor \frac{\mathbb{R}_{min} + IVD_b}{L(AV) + IVD_b} \rfloor \tag{24}$$

where IVD_b is the IVD at b. By substituting (24) in (22), the Smart-Platoon size is estimated as,

$$n \leq 2\left[\frac{\mathbb{R}_{min} + IVD_0}{L(AV) + IVD_0}\right] - 1 \tag{25}$$

where IVD_0 is the stabilized IVD at b.

Case 2: When $V_{-1} < V_\mu$ and $V_0 \geq V_\mu$, the Smart-Platoon undergoes an overshoot in the velocity, and thus the IVD will reach the balanced state with the stabilized velocity V_0. In this case, the IVD changes uncontrollably with many influences in a non-linear fashion. However, this could be solved when there are adequate AVs in between any two AVs in the Smart-Platoon. Thus, the total overshoot of the IVD will become 0 (i.e., $\sum IVD_x(t) = 0$). Therefore from (25), the number of AVs in the Smart-Platoon for this case is derived as,

$$n \leq 2\left[\frac{\mathbb{R}_{min}}{L(AV)}\right] - 1$$

Case 3: When $V_0 < V_\mu$, the velocity of SP_i underdamped. Hence, the Smart-Platoon overshoots to reach b. Also, the IVD experiences the same for reaching the stabilized IVD (IVD_0). To estimate the IVD, a parameter γ is introduced. γ is the maximum value to find the appropriate Smart-Platoon size n. Thus, the stabilized IVD is calculated approximately by,

$$\widetilde{IVD_0} = IVD_0 * (1 + \gamma)$$

From (25), the Smart-Platoon size is calculated as,

$$n \leq 2\left[\frac{\mathbb{R}_{min} + IVD_0}{L(AV) + \widetilde{IVD_0}}\right] - 1$$

C. Inter-Platoon Distance (IPD)

The IPD in Smart-Platooning is defined as the distance between two adjacent platoons. In deciding the desired inter-platoon distance (IPD_i^{des}) between SP_i and SP_{i-1}, the velocity changes and the minimum transmission range plays a major role. Thus, the IPD_i increases when there is a velocity change in the platoons, which is,

$$IPD_i = \int_{t_{bd}}^{t_{ed}} \left(v_n^{i-1} - v_1^i\right) dt + IPD^{des} \tag{26}$$

where t_{bd} is the time at which the SP_{i-1} begins deceleration and t_{ed} is the time at which it ends deceleration to v_{-1}. v_n^{i-1} represents the velocity of n-th AV (tail AV), v_1^i represents the velocity of the first AV (captain AV), and D^{des} is the desired IPD in SP_i.

In the estimation of IPD^{des}, the change in the acceleration Δacc at different velocities v_n^{i-1} and v_1^i in the deceleration phase $[t_{bd}, t_{ed}]$ plays a major role. Thus, the changes are described as follows,

$$\begin{aligned}\Delta acc &= \int_{t_{bd}}^{t_{ed}} \left(v_n^{i-1} - v_1^i\right) dt \\ &= D\left(i-1\right)|_{t_{bd}} - D\left(i-1\right)|_{t_{ed}} \\ &\leq 2\mathbb{R}_{min} - (n_{i-1} * L(AV)) \\ &\quad - \left[(n_{i-1} - 1) * (1+\gamma) * IVD_{-1}\right]\end{aligned} \tag{27}$$

where IVD_{-1} is the IVD at velocity V_{-1}. The term IVD_{-1} is interpreted as,

$$IVD_{-1} = (V_{-1} * T_h) + IVD_{min}$$

Thus, (27) can be rewritten as,

$$\begin{aligned}\Delta acc \approx 2\mathbb{R}_{min} - (n_{i-1} * L(AV)) - [(n_{i-1} - 1) \\ * (1+\gamma) * ((V_{-1} * T_h) + IVD_{min})]\end{aligned}$$

The substitution of (26) in (27), yields the desired IPD as follows,

$$\begin{aligned}IPD^{des} \leq \{(n_{i-1} * L\left(AV\right)) - [(n_{i-1} - 1) \\ * \ (1-\gamma) * ((V_{-1} * T_h) + IVD_{min})]\}/2\end{aligned}$$

Here, $\mathbb{R}_{min}$ guarantees the parameter γ for transmitting the information to all the AVs in Smart-Platoon in one hop. This is given as, $IPD_i \leq \mathbb{R}_{min}$. However, at the balanced state, IPD^{des} should possess the following constraint to phase out the overshoot of IVD. This is mathematically illustrated as,

$$IPD^{des} \geq D - (n * L\left(AV\right)) - [IVD_0 * (n-1)] + IPD_{min}$$

where IPD_{min} is the minimum IPD, and IVD_0 is the stabilized IVD at the balanced point b. Accordingly, the constraint of desired IPD is,

$$\begin{aligned}&D - (n * L(AV)) - [IVD_0 * (n-1)] \\ &\quad + IPD_{min} \leq IPD^{des} \leq \{(n_{i-1} * L(AV)) - [(n_{i-1} - 1) \\ &\quad * \ (1-\gamma) * ((V_{-1} * T_h) + IVD_{min})]\}/2\end{aligned} \tag{28}$$

VI. Results and Discussions

To validate the theoretical analysis of Smart-Platooning mechanism, the platooning scenarios are simulated extensively. The simulation results support the evaluation of the proposed system. This section originates from the simulation settings, proceeds with the analysis of 5G-V2X connectivity, and concludes with analyzing the impact of caption-AV driving strategy in Smart-Platooning.

TABLE II
Simulation Parameters and Values

Simulation Parameter	Value
MAC protocol	SC-FDMA
Simulated area	5000 x 5000 m
Spectrum band	5.9 GHz
Number of 5G Base stations (gNB)	7
Acceleration	0.2 to 4 m/s^2
Deceleration	0.4 to 4 m/s^2
Minimum Transmission Range (R_{min})	300 to 600 m
Length of an AV	2.5 to 5 m
Minimum Inter-Vehicle Distance	0.5 to 3 m
Desired Headway Time	1.5 s
Maximum speed of an AV	10 to 35 m/s
Simulation time	6000 s

A. Simulation Settings

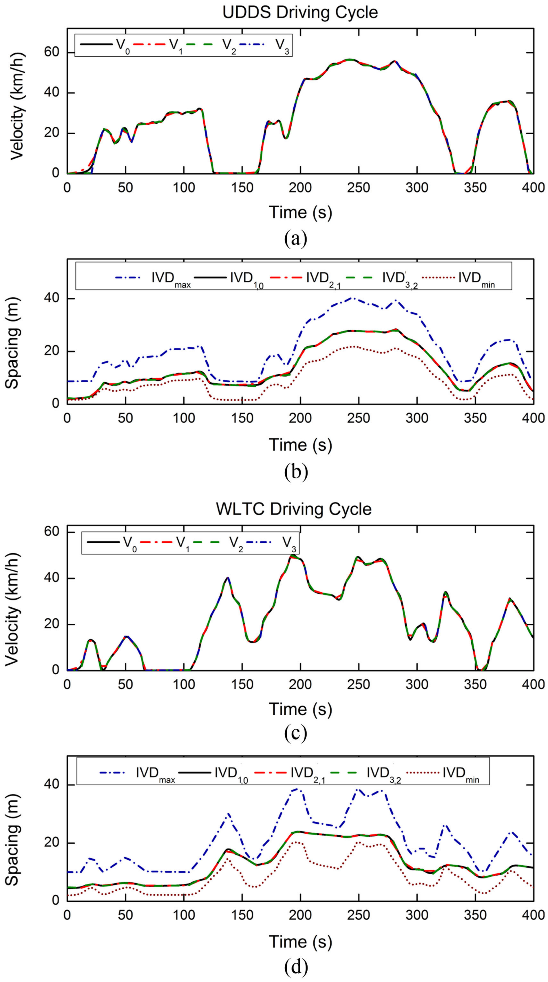
The simulation tools such as PLEXE, and Simulation of Urban Mobility (SUMO) are used to verify the stability and performance of the proposed Smart-Platooning approach. PLEXE [37] is a platooning simulator, which integrates network simulator OMNeT++/Veins and traffic simulator SUMO. PLEXE utilizes predecessor-follower topology for simulating the Smart-Platoon with the traffic characteristics obtained from SUMO. The output file of PLEXE-SUMO is the mobility file containing floating car data, which holds vehicle ID, position, speed, vehicle type, and angle of vehicle and road information such as lane, edge, lane changes, slope, etc. The floating car data is generated for each vehicle in the network for each timestamp. Thus, the dataset generated by PLEXE-SUMO is merged with the network simulator tool OMNeT++/Veins, and the DRG-SP algorithm is executed. The performance of Smart-Platoon is analyzed by implementing the proposed strategy in a four-lane highway road profile. The parameters used for the simulation are summarized in Table II.

B. Performance Analysis

1) Analysis of String Stability: The string stability of the DRG-SP strategy is evaluated comprehensively under the Urban Dynamometer Driving Schedule (UDDS) and Worldwide Harmonized Light Vehicles Test Cycle (WLTC) driving cycles. A platoon is said to be string stable if the degree of disturbance/perturbation over the captain AV is not amplified over the adherent AVs of the platoon downstream. The string stability can be mathematically expressed as [38],

$$\beta_{SP_i} = \left\| \frac{acc_x}{acc_{x-1}} \right\| \leq i \leq L(SP_i) \tag{29}$$

where β_{SP_i} is the frequency response function of the Smart-Platoon SP_i, acc_x is the acceleration of the follower AV AV_x and acc_{x-1} is the acceleration of the predecessor AV AV_{x-1}, and $L(SP_i)$ is the length of the Smart-Platoon. This evaluation considers one captain AV and three adherent AVs. The AVs'



(a)

(b)

(c)

(d)

Fig. 2. DRG-SP: (a) Velocity evolutions under UDDS driving cycle. (b) Spacing evolutions under UDDS driving cycle. (c) Velocity evolutions under WLTC driving cycle. (d) Spacing evolutions under WLTC driving cycle.

initial velocities are 2 m, 1.8 m, and 2.3 m. Fig. 2 shows the evolutions of velocities and spacings of DRG-SP under UDDS and WLTC driving cycles. The figure shows that the DRG-SP strategy adapts to the UDDS and WLTC driving cycles well, and the spacing between the AVs is maintained to the possible safer spacing range [39]. In addition, the V2V communication ensures string stability both theoretically and experimentally while enabling the AVs to maintain IVD [36]. Thus, DRG-SP ensures string stability and vehicle safety.

2) Complexity Analysis of DRG-SP Algorithm: The proposed DRG-SP algorithm begins with a single-step operation for initializing the parameters and the time complexity is $O(1)$. For determining action for each state of the AV environment, the algorithm carries the complexity of $O(n \log n)$. Once the actions are determined, the complexity of $O(\log n)$ is required for updating parameters of actor and critic components (g^{μ} and g^{Q}). For updating the target actor and critic component ($g^{\mu'}$ and $g^{Q'}$), the complexity is $O(1)$. Hence, the total complexity of the DRG-SP algorithm is $O(n \log n)$.

3) Convergence Property of DRG-SP Algorithm: The convergence property of a deep learning algorithm is essential since

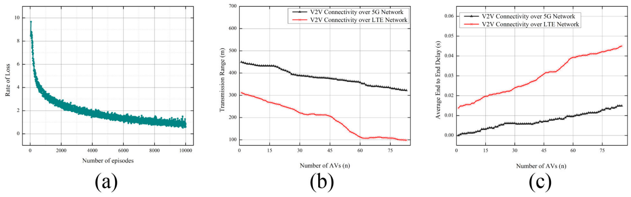


Fig. 3. (a) Analysis of DRG-SP algorithm's convergence rate. (b) Analysis of network connectivity for V2X communication. (c) Latency analysis of 5G-V2X connectivity for V2V communication.

it directly links with the algorithm's optimality. Typically, a deep reinforcement learning algorithm converges when its learning curve becomes flat. Thus, to obtain the convergence values, the loss function ($L(g^{\mu})$) of each training episode of the DRG-SP algorithm with the training samples of 50 AVs is plotted. The simulation results prove that the DRG-SP algorithm converges after 10^5 episodes. Fig. 3(a) shows the convergence property of DRG-SP and is observed that the algorithm converges at a reasonable stage.

4) 5G-V2X Connectivity for V2V Communication: This analysis examines and compares the 5G-V2X connectivity with the existing Cellular-V2X (LTE) technology. The typical V2V connectivity in the 5G network ranges from 108 m to 450 m [28]. The connectivity of networks is analyzed by varying the number of AVs in a platoon. Thus, the maximum size of the platoon is considered as $n = 50$. Fig. 3(b) visualizes the simulation results obtained. The figure shows that when the number of AVs is high (i.e., $n = 50$), the AV_{25}^{i} acts as relay AV for the platoon. The relay AV holds the transmission range of $\mathbb{R} = 322\ m \cong$ 320 m, which is appropriate for transmitting the information. By contrast, the existing LTE network offers a transmission range of $\mathbb{R} = 92\ m \cong$ 90 m. Similarly, with a minimum of $n = 5$ AVs, AV_3 is the relay AV for the platoon, so the Smart-Platoon shows a transmission range of $\mathbb{R} = 438\ m \cong$ 440 m. However, at the minimum number of AVs, the LTE offers the transmission range of $\mathbb{R} = 295\ m \cong$ 300 m. Thus from observation, the effective transmission range of 5G networks in offering V2V connectivity can be visualized.

5) Latency Analysis of 5G-V2X Connectivity for V2V Communication: Latency in V2V communication is the time delay calculated between the sending time and the receiving time of data. The analysis and comparison of the latency of the 5G network with the existing LTE cellular network is made by varying the number of AVs in a platoon. The comparative results of 5G-V2X latency analysis for V2V communication are illustrated in Fig. 3(c). This analysis considers the number of AVs, ranging from $n = 1$ to $n = 50$. When the number of AVs is low in range ($n = 7$), the caption AV (AV_1) transmits the data or control messages to the relay AV (AV_4), the relay AV then transmit to the remaining adherent AVs. Hence, the aggregated value of latency acquired in the transmission between captain AV and relay AV with the latency achieved in communication between relay AV and tail AV helps to assess the overall latency. In this manner, for the least number of AVs in the platoon ($n =$ 3), Smart-Platoon shows the latency of 0.001 s. Whereas, the LTE network suffers from 0.017 s latency. When the number

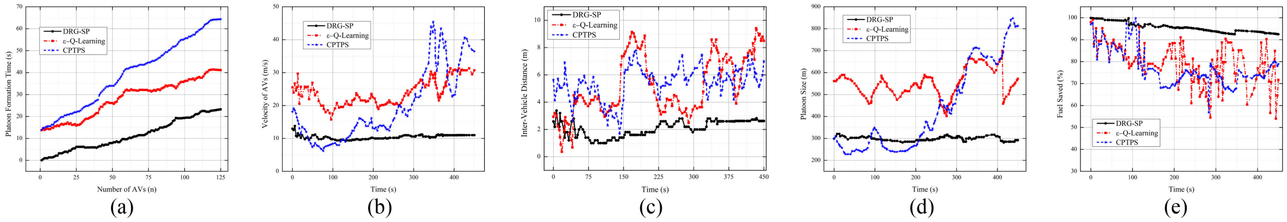


Fig. 4. Comparative analysis of (a) Time taken for platoon formation. (b) Change in velocity. (c) Changes in Inter-Vehicle Distance. (d) Changes in platoon size. (e) Fuel energy consumption.

of AVs gradually increases ($n = 25$) with relay AV (AV_{13}), the 5G network with a latency of 0.012 s surpasses the LTE network with a latency of 0.039 s holding the difference of 89%. Correspondingly, the latency difference moves to 91% when the number of AVs reaches a peak of $n = 50$.

6) Comparative Analysis of Time Taken for Platoon Formation: The time taken for forming a platoon is a crucial factor in the performance analysis. The lower the time is taken, the higher will be the effective communication and fuel efficiency. Fig. 4(a) shows the comparative study of time carried for forming a platoon when the number of AVs varies. The adaptive 5G-V2X communication in the Smart-Platooning aids the network to create the platoons rapidly. As shown in the figure, all the platoon mechanisms undergo an increase in platoon formation time when the number of AVs increases. However, when compared with the existing mechanisms, DRG-SP has minimized platoon formation time.

7) Comparative Analysis of Change in Velocity: This section delivers the study and analysis of the rate of change in the Smart-Platoon velocity by concerning the time. In comparison, the mechanism which results in a frequent change in velocity results in higher fuel consumption. For this performance analysis, the range of acceleration is from $acc = 0.8$ m/s^2 to $acc = 1.5$ m/s^2, with corresponding stabilized velocity $V_0 = 27$ m/s and lowest velocity up to $V_{-1} = 5$ m/s. The critical velocity ranges from $V_\mu \approx 14$ m/s to $V_\mu \approx 18$ m/s, with damping ratio $\alpha = [0.78,\ 1.35]$. The velocity of Smart-Platooning and existing mechanisms observed with varying time-periods is plotted in Fig. 4(b). From the figure, the observation is that the existing mechanism faces frequent changes in the velocity with large differences. The high-velocity changes are due to the loss in connectivity and control messages from the captain AV. Further, the DRG-SP algorithm does not possess velocity overshoot, reflecting in smoother travel with effective fuel consumption.

8) Comparative Analysis of Change in Inter-Vehicle Distance: IVD is one of the essential properties in deciding the platoon structure and paves the way for congestion-free travel. The estimation of this characteristic is based on $IVD_x^{dev}(t)$. The IVD observed for the Smart-Platooning, and existing platoon mechanisms is plotted against various time-periods and is displayed in Fig. 4(c). The observation from the figure shows that the existing mechanisms suffer from frequent and high changes in the IVD. Initially, the Smart-Platoon undergoes frequent changes in the IVD, due to the joining of AVs for creating the platoon. Later, when the balanced state is achieved, the IVD of the Smart-Platoon is stabilized. Thus, the stabilized IVD is $IVD_x(t) \approx 1$ m when $n = 17$ and $IVD_x(t) \approx 2.8$ m when $n = 50$.

TABLE III
AVERAGE FUEL CONSUMPTION

Strategy	Fuel Consumption (L/100 km)	Improvement
CPTPS	6.43	-
ϵ-Q-learning	5.78	4.65%
DRG-SP	5.12	8.57%

9) Comparative Analysis of Change in Platoon Size: The change in the size of the platoon is analyzed and visualized in Fig. 4(d). The figure shows the steadiness of the Smart-Platooning process in maintaining the platoon size. Unlike the existing mechanisms, the Smart-Platoon formed by the DRG-SP algorithm is not suffered from platoon size overshoot.

10) Comparative Analysis of Fuel Energy Consumption: Fig. 4(e) shows the average fuel energy saved with the impact of varying time periods. The amount of fuel energy consumption depends on the changes that the platoon undergoes and platoon stability. The platoon size is another essential factor that affects the percentage of fuel savings. The average fuel consumption of the DRG-SP and the existing strategies is tabulated in Table III. Since the Smart-Platooning is adaptive and robust to various changes and disturbances in the platoon structure, it has better fuel savings.

VII. Conclusion

In this paper, the various dynamics associated with the platooning process are studied, and the DRG-SP algorithm is proposed for Smart-Platooning the AVs. The DRG-SP algorithm control, manage and make decisions of the captain AV. Following the captain AV, the adherent AVs in the platoon forms a Smart-Platoon. The simulation results showed the improved performance of the DRG-SP algorithm in terms of convergence property, connectivity strength, fuel savings, and platoon features. On the analysis of the parameters such as time taken to form a platoon, and the frequent changes in velocity, IVD, platoon size, and platoon stability, the Smart-Platooning strategy offers better performance and a smooth platoon.

References

[1] G. Raja, S. Anbalagan, G. Vijayaraghavan, P. Dhanasekaran, Y. D. Al-Otaibi, and A. K. Bashir, "Energy-efficient end-to-end security for software defined vehicular networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5730–5737, Aug. 2021, doi: 10.1109/TII.2020.3012166.

[2] G. Raja, S. Anbalagan, G. Vijayaraghavan, S. Theerthagiri, S. V. Suryanarayan, and X. W. Wu, "SP-CIDS: Secure and private collaborative IDS for VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4385–4393, Jul. 2021, doi: 10.1109/TITS.2020.3036071.

[3] G. Raja, A. Ganapathisubramaniyan, S. Anbalagan, S. B. M. Baskaran, K. Raja, and A. K. Bashir, "Intelligent reward-based data offloading in next-generation vehicular networks," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3747–3758, May 2020.

[4] C. Zhai, Y. Liu, and F. Luo, "A switched control strategy of heterogeneous vehicle platoon for multiple objectives with state constraints," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1883–1896, May 2019.

[5] Y. Liu, H. Gao, C. Zhai, and W. Xie, "Internal stability and string stability of connected vehicle systems with time delays," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6162–6174, Oct. 2021, doi: 10.1109/TITS.2020.2988401.

[6] C. Zhai, F. Luo, Y. Liu, and Z. Chen, "Ecological cooperative look-ahead control for automated vehicles travelling on freeways with varying slopes," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1208–1221, Feb. 2019.

[7] C. Zhai, F. Luo, and Y. Liu, "Cooperative look-ahead control of vehicle platoon for maximizing fuel efficiency under system constraints," *IEEE Access*, vol. 6, pp. 37700–37714, Jun. 2018, doi: 10.1109/ACCESS.2018.2848480.

[8] N. Kumar, S. Misra, J. J. P. C. Rodrigues, and M. S. Obaidat, "Coalition games for spatio-temporal Big Data in internet of vehicles environment: A comparative analysis," *IEEE Internet Things J.*, vol. 2, no. 4, pp. 310–320, Aug. 2015.

[9] S. Anbalagan, D. Kumar, G. Raja, and A. Balaji, "SDN assisted Stackelberg game model for LTE-WiFi offloading in 5G networks," *Digit. Commun. Netw.*, vol. 5, no. 4, pp. 268–275, 2019.

[10] S. B. Prathiba, G. Raja, A. K. Bashir, A. A. Alzubi, and B. Gupta, "SDN-assisted safety message dissemination framework for vehicular critical energy infrastructure," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2021.3113130.

[11] N. Kumar, J. J. P. C. Rodrigues, and N. Chilamkurti, "Bayesian coalition game as-a-service for content distribution in Internet of Vehicles," *IEEE Internet Things J.*, vol. 1, no. 6, pp. 544–555, Dec. 2014.

[12] Z. Zhou, Z. Akhtar, K. L. Man, and K. Siddique, "A deep learning platooning-based video information-sharing Internet of Things framework for autonomous driving systems," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 11, pp. 15–23, 2019.

[13] H. Ma, S. Li, E. Zhang, Z. Lv, J. Hu, and X. Wei, "Cooperative autonomous driving oriented MEC-aided 5G-V2X: Prototype system design, field tests and AI-based optimization tools," *IEEE Access*, vol. 8, pp. 54 288–54 302, Mar. 2020, doi: 10.1109/ACCESS.2020.2981463.

[14] N. Kumar, N. Chilamkurti, and J. H. Park, "ALCA: Agent learning-based clustering algorithm in vehicular ad hoc networks," *Pers. Ubiquitous Comput.*, vol. 17, no. 8, pp. 1683–1692, 2013.

[15] S. B. Prathiba, G. Raja, S. Anbalagan, K. Dev, S. Gurumoorthy, and A. P. Sankaran, "Federated learning empowered computation offloading and resource management in 6G-V2X," *IEEE Trans. Netw. Sci. Eng.*, to be published, doi: 10.1109/TNSE.2021.3103124.

[16] C. Chen, T. Xiao, T. Qiu, N. Lv, and Q. Pei, "Smart-contract-based economical platooning in blockchain-enabled urban internet of vehicles," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4122–4133, Jun. 2020.

[17] G. Luo *et al.*, "Software defined cooperative data sharing in edge computing assisted 5G-VANET," *IEEE Trans. Mobile Comput.*, vol. 20, no. 3, pp. 1212–1229, Mar. 2021, doi: 10.1109/TMC.2019.2953163.

[18] H. I. Abbasi, R. C. Voicu, J. A. Copeland, and Y. Chang, "Towards fast and reliable multihop routing in VANETs," *IEEE Trans. Mobile Comput.*, vol. 19, no. 10, pp. 2461–2474, Oct. 2020.

[19] D. Wu, J. Wu, and R. Wang, "An energy-efficient and trust-based formation algorithm for cooperative vehicle platooning," in *Proc. Int. Conf. Comput., Netw. Commun.*, 2019, pp. 702–707.

[20] C. Zhai, X. Chen, C. Yan, Y. Liu, and H. Li, "Ecological cooperative adaptive cruise control for a heterogeneous platoon of heavy-duty vehicles with time delays," *IEEE Access*, vol. 8, pp. 146208–146219, Aug. 2020, doi: 10.1109/ACCESS.2020.3015052.

[21] M. Di Vaio, P. Falcone, R. Hult, A. Petrillo, A. Salvi, and S. Santini, "Design and experimental validation of a distributed interaction protocol for connected autonomous vehicles at a road intersection," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 9451–9465, Oct. 2019.

[22] Y. Li, C. Tang, K. Li, X. He, S. Peeta, and Y. Wang, "Consensus-based cooperative control for multi-platoon under the connected vehicles environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2220–2229, Jun. 2019.

[23] Y. Liu, C. Pan, H. Gao, and G. Guo, "Cooperative spacing control for interconnected vehicle systems with input delays," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 10 692–10 704, Dec. 2017.

[24] C. Zhai, F. Luo, and Y. Liu, "Cooperative power split optimization for a group of intelligent electric vehicles travelling on a highway with varying slopes," *IEEE Trans. Intell. Transp. Syst.*, to be published, doi: 10.1109/TITS.2020.3045264.

[25] Y. Feng, D. He, and Y. Guan, "Composite platoon trajectory planning strategy for intersection throughput maximization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6305–6319, Jul. 2019.

[26] K. Bian, G. Zhang, and L. Song, "Toward secure crowd sensing in vehicle-to-everything networks," *IEEE Netw.*, vol. 32, no. 2, pp. 126–131, Mar./Apr. 2018.

[27] N. Kumar, R. Iqbal, S. Misra, and J. J. Rodrigues, "Bayesian coalition game for contention-aware reliable data forwarding in vehicular mobile cloud," *Future Gener. Comput. Syst.*, vol. 48, pp. 60–72, 2015.

[28] C. Ho, B. Huang, M. Wu, and T. Wu, "Optimized base station allocation for platooning vehicles underway by using deep learning algorithm based on 5G-V2X," in *Proc. IEEE 8th Global Conf. Consum. Electron.*, 2019, pp. 1–2, doi: 10.1109/GCCE46687.2019.9014645.

[29] C. Chen, J. Jiang, N. Lv, and S. Li, "An intelligent path planning scheme of autonomous vehicles platoon using deep reinforcement learning on network edge," *IEEE Access*, vol. 8, pp. 99059–99069, May 2020, doi: 10.1109/ACCESS.2020.2998015.

[30] Y. Zheng, S. Eben Li, J. Wang, D. Cao, and K. Li, "Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 14–26, Jan. 2016.

[31] N. C. Luong *et al.*, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surv. Tut.*, vol. 21, no. 4, pp. 3133–3174, Oct.–Dec. 2019.

[32] H. Huang, Y. Yang, H. Wang, Z. Ding, H. Sari, and F. Adachi, "Deep reinforcement learning for UAV navigation through massive MIMO technique," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1117–1121, Jan. 2020.

[33] X. Wang, Y. Gu, Y. Cheng, A. Liu, and C. L. P. Chen, "Approximate policy-based accelerated deep reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 6, pp. 1820–1830, Jun. 2020.

[34] Q. Wang and Q. Wang, "Prioritized guidance for efficient multi-agent reinforcement learning exploration," 2019, *arXiv:1907.07847*.

[35] A. Stooke and P. Abbeel, "Accelerated methods for deep reinforcement learning," 2018, *arXiv:1803.02811*.

[36] Y. Dong, X. Tang, and Y. Yuan, "Principled reward shaping for reinforcement learning via Lyapunov stability theory," *Neurocomputing*, vol. 393, pp. 83–90, 2020.

[37] F. D. Rango, P. Raimondo, and D. Amendola, "Extending SUMO and PLEXE simulator modules to consider energy consumption in platooning management in VANET," in *Proc. IEEE/ACM 23rd Int. Symp. Distrib. Simul. Real Time Appl.*, 2019, pp. 1–9, doi: 10.1109/DS-RT47707.2019.8958692.

[38] Y. Yang *et al.*, "Cooperative ecological cruising using hierarchical control strategy with optimal sustainable performance for connected automated vehicles on varying road conditions," *J. Cleaner Prod.*, vol. 275, pp. 1–15, 2020, Art. no. 123056.

[39] X. Chen, J. Yang, C. Zhai, J. Lou, and C. Yan, "Economic adaptive cruise control for electric vehicles based on ADHDP in a car-following scenario," *IEEE Access*, vol. 9, pp. 74949–74958, May 2021, doi: 10.1109/ACCESS.2021.3081184.