Multi-Agent DRL for Resource Allocation in AoI-Aware Energy-Efficient C-V2X Networks

Yuxiang Zheng*, Bishmita Hazarika*, Trung Q. Duong*‡ Keshav Singh†, Vishal Sharma‡,
and Octavia A. Dobre*
*Memorial University, Canada (e-mails: {y.zheng, odobre, tduong}@mun.ca)
†National Sun Yat-sen University, Taiwan (e-mail: keshav.singh@mail.nsysu.edu.tw)
‡Queen's University Belfast, UK (e-mail: v.sharma@qub.ac.uk)

***Abstract*—This paper aims to tackle the complex problem of channel assignment and joint power-energy allocation within a cellular-vehicle-to-everything (C-V2X) network in an urban traffic intersection. Here, the C-V2X network is strategically deployed to facilitate the coordination of multiple vehicle platoons. This includes updating platoon states to the roadside unit (RSU) and managing the exchange of cooperative awareness messages (CAMs) among vehicles. Our objective is to minimise the average age of information (AoI), maximise the CAM delivery probability, and promote sustainable, green communication practices through optimal power-energy management. Given the intricate nature of this challenge, we adopt a multi-agent deep reinforcement learning (MADRL) approach based on Markov decision process (MDP). Next, we introduce two innovative algorithms based on multi-agent deep deterministic policy gradient (MADDPG) and twin delayed deep deterministic policy gradient (TD3) to effectively address the optimisation problem. Finally, the simulation results demonstrate remarkable performance in terms of energy efficiency, while maintaining algorithm convergence speed and AoI level.**

***Index Terms*—Vehicle-to-everything, multi-agent deep reinforcement learning, age of information, resource allocation, green communication.**

I. Introduction

With the advancements in autonomous systems, self-driving vehicles and intelligent transportation systems (ITS) have emerged as critical elements in the blueprint of any smart city application [1]. Consequently, they have become a focus of long-term interest and extensive study [1]–[9]. ITS holds considerable potential for alleviating traffic congestion, lowering the risk of car accidents, and improving urban air quality [1]. To address the information transmission issues in ITS, vehicle-to-everything (V2X) [2]–[4] plays an important role, as it enables vehicle-to-vehicle (V2V), vehicle-to-pedestrian (V2P), vehicle-to-infrastructure (V2I), and vehicle-to-network (V2N) communications to broadcast almost real-time updates on surrounding transportation conditions and potential hazards. One pattern that utilises the concept of V2X to realise ITS is the platoon-based control strategy [2]–[7] which packs the close same-line self-driving vehicles into platoons to achieve more efficient traffic control and higher traffic flow. In one pack of vehicles (a platoon), the first vehicle is considered the platoon leader (PL), which is responsible for uploading platoon state messages to and receiving control commands from the roadside unit (RSU), as well as exchanging cooperative awareness messages (CAMs) [9] with the other vehicles in the platoon, namely V2I and V2V communications. Keeping highly frequent updates of platoon states to the RSU is essential in this design, which allows RSU to maintain control of the time-critical information (e.g. safety information). Age of information (AoI) [8] is introduced as a measure of the freshness of information, which grows over time if the PL does not update the RSU, and is reset after each successful update. A lot of work has been done in minimising the AoI in the vehicular networks [6]–[8].

In this work, we address the resource allocation problem in a cellular-V2X (C-V2X) network at an intersection, aiming to minimise AoI and energy consumption using a platoon-based strategy. Building on prior research [5], [6], our model integrates the distributed resource allocation of Mode 4 [4] and urban scenarios [2], [4]. Given the dynamic environment—rapid vehicle movements and diverse communication demands, we employ multi-agent deep reinforcement learning (MADRL) using the extension of standard deep deterministic policy gradient (DDPG)—multi-agent DDPG (MADDPG), enhanced by decomposed MADDPG (DE-MADDPG) [10], task decomposition (TDec) algorithm [11], and twin delayed deep deterministic policy gradient (TD3) [12] to tackle this complex challenge.

Conventional studies on vehicular networks have largely focused on the immediate impact of power consumed by the PL on communication quality, often overlooking the broader, long-term implications of energy usage which significantly affects operational costs. Our research addresses this gap by adopting a holistic approach to power and energy allocation, enhancing both the effectiveness of operational communications and the long-term sustainability of the C-V2X network. This approach supports the global shift towards green communication technologies, which are increasingly valuable for their environmental benefits [13]. Thus, this study

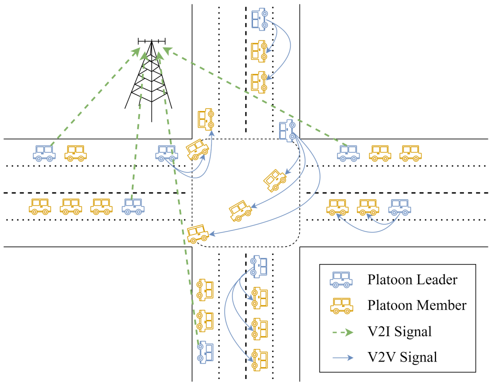

Fig. 1: A single-antenna multi-platoon C-V2X network.

makes the following key contributions to the field of ITS and green communications by introducing innovative techniques and methodologies:

- We formulate a problem that optimises the AoI, the probability of CAM delivery, and critically, the joint power-energy consumption within a platoon-based C-V2X network. This framework is designed to enhance network efficiency and sustainability simultaneously.
- To tackle the formulated problem, we employ a MADRL approach by combining MADDPG with several performance enhancement techniques such as DE-MADDPG, TDec, and TD3 to achieve better optimisation results.
- Finally, the numerical results confirm that our energy-focused algorithms considerably lower energy usage compared to traditional methods, while maintaining similar convergence rates and AoI. Additionally, we introduce a metric specifically for assessing energy efficiency, underscoring the benefits of our approaches.

II. System Model and Problem Formulation

A. Environment Model

Fig. 1 illustrates an intersection managed by a single-antenna RSU at its centre, coordinating L platoons ($L \in \mathbb{N}$, $\mathcal{L} = \{1, 2, \dots, L\}$) via a C-V2X system. Each platoon $l \in \mathcal{L}$ consists of V_l vehicles ($V_l \in \mathbb{N}^+$, $\mathcal{V}_l = \{1, 2, \dots, V_l\}$). The head-of-line vehicle in each platoon, designated as the PL ($v_l = 1$ for each $v_l \in \mathcal{V}_l$), leads the communication with the RSU and within its platoon. In V2I mode, the RSU gathers states and relays states and commands based on the information from each PL. In V2V mode, CAMs are exchanged among vehicles within the same platoon.

B. Communication Model

In this work, orthogonal frequency-division multiplexing (OFDM) is used to cope with the frequency-selective wireless channels [14]. The system's wireless channel is segmented into K orthogonal subchannels, each with bandwidth W, where $K \in \mathbb{N}^+$ and $\mathcal{K} = \{1, 2, \dots, K\}$ denotes the set of subchannels. The channel fading is independent across all subchannels and remains constant within a single coherence time, Δt. Accordingly, the timeline is divided into intervals of duration Δt, with $t \in \mathbb{N}^+$ serving as the index for time steps. The channel gain for PL$_l$ (leader of platoon l) in subchannel $k \in \mathcal{K}$ at time step t is expressed as $h^t_l[k] = \beta^t_l g^t_l[k]$, where β^t_l and $g^t_l[k]$ are the large and small scale fading, respectively. In addition, two decision parameters $\delta^t_{l,k}, \lambda^t_l \in \{0, 1\}$ are defined for the channel and communication mode selections. If $\delta^t_{l,k} = 1$, the subchannel k is allocated to the platoon l at time step t, PL$_l$ will use it for V2I mode communication when $\lambda^t_l = 0$, and V2V mode communication when $\lambda^t_l = 1$. The channel capacity for V2I and V2V communications can hence be written as

$$\begin{aligned} C^t_{l,\mathcal{R}}[k] &= \log_2\left(1 + \frac{(1-\lambda^t_l)\delta^t_{l,k}p^t_l[k]h^t_{l,\mathcal{R}}[k]}{I^t_{l,\mathcal{R}}[k] + \sigma^2}\right), \\ I^t_{l,\mathcal{R}}[k] &= \sum_{l',l'\neq l} \delta^t_{l',k}p^t_{l'}[k]h^t_{l',\mathcal{R}}[k], \end{aligned} \tag{1}$$

$$\begin{aligned} C^t_{l,v_l}[k] &= \log_2\left(1 + \frac{\lambda^t_l\delta^t_{l,k}p^t_l[k]h^t_{l,v_l}[k]}{I^t_{l,v_l}[k] + \sigma^2}\right), \\ I^t_{l,v_l}[k] &= \sum_{l',l'\neq l} \delta^t_{l',k}p^t_{l'}[k]h^t_{l',v_l}[k], v_l \in \mathcal{V}_l \backslash \{1\}, \end{aligned} \tag{2}$$

where the signal-to-interference-plus-noise ratio (SINR) is estimated from the interference of other platoons which is treated as noise. The power used by PL$_l$ on subchannel k is denoted by $p^t_l[k]$. The channel gains from PL$_l$ to the RSU and to other vehicles within platoon l are represented by $h^t_{l,\mathcal{R}}[k]$ and $h^t_{l,v_l}[k]$, respectively. The noise power is denoted by σ^2. The interference power experienced at the RSU and the vehicles within platoon l, denoted as $I^t_{l,\mathcal{R}}[k]$ and $I^t_{l,v_l}[k]$, is calculated from the transmit power $p^t_{l'}[k]$ of other PLs and the corresponding channel gains $h^t_{l',\mathcal{R}}[k]$ and $h^t_{l',v_l}[k]$.

C. Age of Information Model

AoI is crucial in ensuring that the PLs maintain timely information exchange with the RSU via V2I communication. This exchange is necessary for updating the platoon state and receiving control commands. The AoI level for platoon l at the t^{th} coherence time step is represented by A^t_l and updated as

$$A^{t+1}_l = \begin{cases} 1, & \text{if } C^t_{l,\mathcal{R}}[k] \geq C^{\min}_{l,\mathcal{R}}, \\ A^t_l + 1, & \text{otherwise,} \end{cases} \tag{3}$$

where '1' represents one time step, $C^{\min}_{l,\mathcal{R}}$ is the minimum required data transmission rate for V2I communication. If the current transmission rate is less than the requirement, i.e., V2I communication fails or V2I mode is not selected, AoI will increase by 1. In contrast, if the requirement is satisfied, i.e., V2I communication is successful, AoI will be reset to 1.

D. Problem Formulation

Based on the discussions and models outlined in the preceding sections, we can now formulate the optimisation problem for platoon l as follows,

$$\begin{aligned} \min_{\boldsymbol{\delta},\boldsymbol{\lambda},\boldsymbol{p},\boldsymbol{E}} & \left\{ \sum_{t=1}^{T} \left[\frac{1}{T} A_l^t, \frac{1}{T} \sum_k p_l^t[k], \sum_k E_l^t[k] \right], \right. \\ & \left. -\Pr\left(\sum_{t=1}^{T} \sum_k \min_{v_l} \left\{ \mathcal{C}_{l,v_l}^t[k] \right\} \Delta t \ge D_l \right) \right\}, \end{aligned} \tag{4}$$

$$\text{s.t. } p_l^t[k] \in \left[0, p^{max}\right], \forall t, l, k, \tag{4a}$$

$$\delta_{l,k}^t, \lambda_l^t \in \{0,1\}, \forall t, l, k, \tag{4b}$$

$$\sum_k \delta_{l,k}^t \le 1, \forall t, l, \tag{4c}$$

$$\sum_l \sum_k \delta_{l,k}^t \le K, \forall t, \tag{4d}$$

where $E_l^t[k] = p_l^t[k]\Delta t$ is the energy consumption of PL$_l$, focusing primarily on communication energy, though it is noted that energy for decision-making and other activities by PL$_l$ could also be considered. The data size for the CAM from platoon l is denoted by D_l. The primary objective of this optimisation problem is to minimise the average AoI, average power consumption, and total energy consumption, while maximising CAM delivery probability within a designated time slot T—where the CAM dissemination frequency ranges from 10 to 100 Hz, implying that the exchange period should be less than 100 ms.

Constraint (4a) ensures PL$_l$'s power consumption remains below p^{max}. Constraints (4c) and (4d) limit PL$_l$ to one subchannel per time step t and restrict total allocations to K subchannels.

Given the complexities of solving this mixed-integer nonlinear programming problem for L platoons, we employ a MADRL approach. The application of this method to address the optimisation problem will be explored in subsequent sections.

III. Preliminaries of the MADRL algorithm

A. Markov Decision Process

From the problem (4) in Sec. II-D, the MADRL issue is modelled as a Markov Decision Process (MDP) [15], described by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, covering state and action spaces $\mathcal{S}, \mathcal{A}$, transition probability $\mathcal{P}$, reward function $\mathcal{R}$, and discount factor $\gamma \in [0,1]$. The formulation is as follows:

- *Agent*: The L platoons at the intersection collectively form the MADRL environment, with each platoon regarded as an agent. PLs observe states, take actions based on their policies, and optimise policies by interacting with the environment.
- *State space*: At time step t, observed by PL$_l$, the state space is defined as
$$\mathcal{S}_l^t = \left[\begin{array}{c} h_{l,\mathcal{R}}^t[k], h_{l,v_l}^t[k], T', D_l', A_l^t, \\ I_{l,\mathcal{R}}^t[k], I_{l,v_l}^t[k], \sigma^2, Local_l \end{array} \right], \tag{5}$$
where $T' \in [0, T]$ represents the remaining time in the time slot T, $D_l' \in [0, D_l]$ is the residual CAM data size to be transferred, and $Local_l$ indicates platoon l's location. Platoons dynamically update their locations at each time step t and make random movement decisions.
- *Action space*: The action by PL$_l$ comprises four components, represented as
$$\mathcal{A}_l^t = \left[\delta_{l,k}^t, \lambda_l^t, p_l^t[k], E_l^t[k]\right]. \tag{6}$$
Given the state space $\mathcal{S}_l^t$, PL$_l$ chooses subchannel $k \in \mathcal{K}$, the communication mode (V2I/V2V), and regulates power and energy at time t. These actions must adhere to constraints (4a)–(4d).
- *Transition probability*: The transition probability $\mathcal{P}$ captures the likelihood of state s moving to state s' upon action a by an agent. It includes: 1) Changes in interference within other PLs' state spaces due to the subchannel and power settings of PL$_l$ for V2I/V2V communication; 2) The inherent randomness of platoon movements (turning right/left or continuing straight), independent of the actions taken.
- *Reward function*: Each of the agents in this multi-agent environment receives a local reward as feedback for its action, while there is also a global reward that measures the joint performance of all the agents. Based on the optimisation problem (4), the local reward function for agent l is defined as
$$\begin{aligned} \mathcal{R}_l^t = & -\mathcal{F}_1\left(A_l^t\right) - \mathcal{F}_2\left(p_l^t[k]\right) - \mathcal{F}_3\left(E_l^t[k]\right) \\ & -\mathcal{F}_4\left(D_l'/D_l\right) + w\mathit{\Gamma}\left(\mathcal{C}_{l,\mathcal{R}}^t[k] - \mathcal{C}_{l,\mathcal{R}}^{\min}\right), \end{aligned} \tag{7}$$
where $\mathcal{F}_1$–$\mathcal{F}_4$ map the first four terms to the same range and weight their contributions, $w > 0$ weights the last term $\mathit{\Gamma}$ which is a stepwise function:
$$\mathit{\Gamma}(x) = \begin{cases} 1, & x \ge 0, \\ 0, & \text{otherwise}. \end{cases} \tag{8}$$
The first four components of (7) align with those in (4), and following [6], the final term uses a stepwise function to promote successful V2I communications. The global reward, considering the interference impact from each PL's subchannel and power choices, is defined as the average interference:
$$\mathcal{R}_g^t = -\frac{1}{L} \sum_l \sum_k \mathcal{F}_5\left(I_{l,x}^t[k]\right), \tag{9}$$
where $\mathcal{F}_5$ maps the interference power to a suitable range, and the global reward is affected by PLs differently according to their communication mode: $I_{l,x}^t[k] = I_{l,\mathcal{R}}^t[k]$ for V2I, $I_{l,x}^t[k] = I_{l,v_l}^t[k]$ for V2V. This design encourages channel selections that minimise disruption to other PLs.

At each time step t, PL_l observes a state $s \in \mathcal{S}_l^t$ from the environment, and then takes action $a \in \mathcal{A}_l^t$ with a probability $\pi_l(a|s)$, where $\pi_l(a|s)$ is the conditional probability that $A_l^t = a$ if $S_l^t = s$, i.e., the policy of PL_l. A reward $R_l^t = r$ will be received from interacting with the environment using the selected action, and then the time is moved to $t+1$.

B. Policy and Value Function

Following previous discussions, the optimisation problem focuses on maximising the expected discounted return via the state-value function V:

$$\begin{aligned} V_l^*(s) &= V_l^{\pi_l^*}(s) = \max_{\pi_l} V_l^{\pi_l}(s) \\ &= \max_{\pi_l} \mathbb{E}_{l,\pi_l}\left[\sum_{n=0}^{\infty} \gamma^n R_l^{t+n+1} \middle| S_l^t = s\right], \forall s \in \mathcal{S}, \end{aligned} \tag{10}$$

where $V_l^{\pi_l}(s)$ is the state-value function, $V_l^{\pi_l^*}(s)$ is the state-value function under the optimal policy π_l^*, $V_l^*(s)$ is the optimal state-value function, and $\mathbb{E}_{l,\pi_l}[\cdot]$ is the expected value of the discounted return $G_l^t = \sum_{n=0}^{\infty} \gamma^n R_l^{t+n+1}$ given that PL_l follows policy π_l. Bellman optimality equation [15] is reviewed to help us solve the problem stated in (10), which shows that the state value under the optimal policy equals the expected return from the state under the best action:

$$V_l^{\pi_l^*}(s) = \max_{a \in \mathcal{A}_l^t(s)} Q_l^{\pi_l^*}(s,a), \forall s \in \mathcal{S}, \tag{11}$$

where $Q_l^{\pi_l}(s,a) = \mathbb{E}_{l,\pi_l}[G_l^t | S_l^t = s, A_l^t = a]$ is the action-value function and $Q_l^{\pi_l^*}(s,a)$ is the action-value function under the optimal policy. Similarly, the optimal action-value function $Q_l^*(s,a)$ is equivalent to $Q_l^{\pi_l^*}(s,a)$, and hence

$$V_l^*(s) = \max_{a \in \mathcal{A}_l^t(s)} Q_l^*(s,a), \forall s \in \mathcal{S}. \tag{12}$$

In the MDP, PL_l continually updates its policy to optimise the state-value function as in (10) and seeks actions that maximise the action-value function—the Q-function in (12). Both strategies aim to achieve the optimal state-value function, effectively solving the optimisation problem in (4).

IV. MADRL Approach

According to the Q-function in Sec. III-B, we implemented the DDPG algorithm, or more precisely, MADDPG algorithm for the joint optimisation of the policy π in (10) and Q-function in (12) in the multi-agent environment. Similar to the work that has been done in [6], we apply the DE-MADDPG algorithm, which is a combination of the standard MADDPG and DDPG algorithms that can be found in [10]. We also combine it with the TDec algorithm proposed in [11] to apply the DE-MADDPG-TDec algorithm. In addition, we introduce TD3 [12] and its extension—multi-agent TD3 (MATD3) to overcome the overestimation problem in Q-functions to further enhance the performance.

A. Decomposed Multi-Agent Deep Deterministic Policy Gradient

Different from the conventional MADDPG (or MATD3) that trains multiple agents with only one critic (or two critics), the main idea behind DE-MADDPG is to introduce the standard DDPG (or TD3) for each local agent and combine with a centralised global critic. The objective becomes optimising the policy to maximise both the local and global critics. The combined policy gradient for agent l is

$$\begin{aligned} \nabla J(\theta_l) = & \overbrace{\mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\mathcal{B}}\Big[\nabla_{\theta_l}\pi_l\left(a_l|s_l\right)\nabla_{a_l}Q_\psi^g(\boldsymbol{s},\boldsymbol{a})\Big]}^{\textit{MADDPG: Global Critic}} \\ & \underbrace{+\,\mathbb{E}_{s_l,a_l\sim\mathcal{B}}\Big[\nabla_{\theta_l}\pi_l\left(a_l|s_l\right)\nabla_{a_l}Q_{\phi_l}^{\pi_l}\left(s_l,a_l\right)\Big]}_{\textit{DDPG: Local Critic}}, \end{aligned} \tag{13}$$

where θ_l parameterises the policy of agent l, $\boldsymbol{s} = (s_1,...,s_L)$, $\boldsymbol{a} = (a_1,...,a_L)$, $\mathcal{B}$ is the experience replay buffer, $a_l = \pi_l(s_l)$ is the action of agent l under its policy, ψ and ϕ_l parameterises the Q-functions—Q_ψ^g for global critic and $Q_{\phi_l}^{\pi_l}$ for local critic. The global critic and the local critic are updated by minimising the following loss functions:

$$\mathcal{L}(\psi) = \mathbb{E}_{\boldsymbol{s},\boldsymbol{a},\boldsymbol{r},\boldsymbol{s}'}\left[\Big(Q_\psi^g(\boldsymbol{s},\boldsymbol{a}) - y_g\Big)^2\right], \tag{14}$$

$$\mathcal{L}(\phi_l) = \mathbb{E}_{s_l,a_l,r_l,s_l'}\left[\Big(Q_{\phi_l}^{\pi_l}(s_l,a_l) - y_l\Big)^2\right], \tag{15}$$

where $\boldsymbol{r} = (r_1,...,r_L)$, $\boldsymbol{s}' = (s_1',...,s_L')$ are the next states, $s_l' \in \boldsymbol{s}'$. The global and local target values are written as

$$y_g = r_g + \gamma Q_{\psi'}^g(\boldsymbol{s}',\boldsymbol{a}')\Big|_{a_l'=\pi_l'(s_l')}, \tag{16}$$

$$y_l = r_l + \gamma Q_{\phi_l'}^{\pi_l}(s_l',a_l')\Big|_{a_l'=\pi_l'(s_l')}, \tag{17}$$

where $\boldsymbol{a}' = (a_1',...,a_L')$ is the next set of actions, $a_l' \in \boldsymbol{a}'$, $Q_{\psi'}^g$ and $Q_{\phi_l'}^{\pi_l}$ represent the target global and local critics, and π_l' is the target policy for agent l.

B. Twin Delayed Deep Deterministic Policy Gradient

In order to tackle the overestimation problem in DDPG, the idea of TD3 is introduced to improve the parameter update in DE-MADDPG. The policy gradient with TD3 for agent l is

$$\begin{aligned} \nabla J(\theta_l) = & \mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\mathcal{B}}\Big[\nabla_{\theta_l}\pi_l\left(a_l|s_l\right)\nabla_{a_l}Q_{\psi_1}^g(\boldsymbol{s},\boldsymbol{a})\Big] \\ & +\mathbb{E}_{s_l,a_l\sim\mathcal{B}}\Big[\nabla_{\theta_l}\pi_l\left(a_l|s_l\right)\nabla_{a_l}Q_{\phi_l}^{\pi_l}\left(s_l,a_l\right)\Big]. \end{aligned} \tag{18}$$

The twin global critics $Q_{\psi_1}^g$ and $Q_{\psi_2}^g$ are updated by minimising the loss function:

$$\mathcal{L}(\psi_i) = \mathbb{E}_{\boldsymbol{s},\boldsymbol{a},\boldsymbol{r},\boldsymbol{s}'}\left[\Big(Q_{\psi_i}^g(\boldsymbol{s},\boldsymbol{a}) - y_g\Big)^2\right], i = 1,2, \tag{19}$$

$$y_g = r_g + \gamma \min_i Q_{\psi_i'}^g(\boldsymbol{s}',\boldsymbol{a}')\Big|_{a_l'=\pi_l'(s_l')}. \tag{20}$$

By delaying the update of the local network by d loops, the final DE-MADDPG (TD3) is described in Algorithm 1.

Algorithm 1: DE-MADDPG (TD3)

```
Initialise intersection environment & replay buffer $\mathcal{B}$.
Initialise global critic networks: $\{Q^g_{\psi_i}, Q^g_{\psi'_i}\}, i = 1, 2$.
Initialise actor & critic networks for each agent:
    $\{\pi_l, \pi'_l, Q^{\pi_l}_{\phi_l}, Q^{\pi_l}_{\phi'_l}\}, l = 1, 2, ..., L$.
for episode $= 1$ to $loop$ do
    Update platoon location & channel information.
    Reset time budget & CAM size: $\{T', D'_l\} = \{T, D_l\}$.
    for $t = 1$ to $T$ do
        for agent 1 to $L$ do
            Observe state $s^t_l$, select action $a^t_l = \pi_l(s^t_l)$,
              receive local & global rewards: $\{r^t_l, r^t_g\}$.
        Update channel fast fading & interference.
        Each agent $l$ observes new state $s^{t+1}_l$.
        Store $(\boldsymbol{s}^t, \boldsymbol{a}^t, \boldsymbol{r}^t, r^t_g, \boldsymbol{s}^{t+1})$ into $\mathcal{B}$.
        Sample mini-batch of $M$ transitions
          $(\boldsymbol{s}^m, \boldsymbol{a}^m, \boldsymbol{r}^m, r^m_g, \boldsymbol{s}^m_{new})\big|^M_{m=1}$ from $\mathcal{B}$.
        Update global critics: minimising $\mathcal{L}(\psi_i)$ (19) by
          one-step gradient descent.
        Target soft update: $\psi'_i \leftarrow \tau\psi_i + (1-\tau)\psi'_i$.
        if $t$ mod $d$ then
            for agent 1 to $L$ do
                Update local critic: minimising $\mathcal{L}(\phi_l)$ (15)
                  by one-step gradient descent.
                Update local actor: maximising $\nabla J(\theta_l)$
                  (18) by one-step gradient ascent.
                Target soft update: $\phi'_l \leftarrow \tau\phi_l + (1-\tau)\phi'_l$
                                    $\theta'_l \leftarrow \tau\theta_l + (1-\tau)\theta'_l$
```

C. *Task Decomposition Algorithm*

According to *Theorem 1* in [11], if the reward function in the MDP can be decomposed into N sub-functions (tasks), i.e., $\mathcal{R}(s,a) = \sum^N_{n=1} \mathcal{R}_n(s,a)$, the state and action value functions can also be decomposed, i.e., $V^\pi(s) = \sum^N_{n=1} V^\pi_n(s)$, $Q^\pi(s,a) = \sum^N_{n=1} Q^\pi_n(s,a)$. Therefore, we adopt the idea of TDec algorithm for our local reward function in (7), and the decomposed functions are written as

$$\begin{aligned}\mathcal{R}^t_{l,1} =& -\mathcal{F}_1\left(A^t_l\right) + w\Gamma\left(\mathcal{C}^t_{l,\mathcal{R}}[k] - \mathcal{C}^{\min}_{l,\mathcal{R}}\right)\\ &-\left(1-\lambda^t_l\right)\left[\mathcal{F}_2\left(p^t_l[k]\right) + \mathcal{F}_3\left(E^t_l[k]\right)\right],\end{aligned} \tag{21}$$

$$\begin{aligned}\mathcal{R}^t_{l,2} =& -\mathcal{F}_4\left(D'_l/D_l\right)\\ &-\lambda^t_l\left[\mathcal{F}_2\left(p^t_l[k]\right) + \mathcal{F}_3\left(E^t_l[k]\right)\right],\end{aligned} \tag{22}$$

where $\mathcal{R}^t_{l,1}$ is the task 1 for V2I mode communication, $\mathcal{R}^t_{l,2}$ is the task 2 for V2V mode communication, and $\mathcal{R}^t_l = \mathcal{R}^t_{l,1} + \mathcal{R}^t_{l,2}$. Hence, our policy gradient (18), local loss function (15), and local target function (17) can be written as

$$\begin{aligned}\nabla J(\theta_l) =& \mathbb{E}_{\boldsymbol{s},\boldsymbol{a}\sim\mathcal{B}}\left[\nabla_{\theta_l}\pi_l\left(a_l|s_l\right)\nabla_{a_l}Q^g_{\psi_1}(\boldsymbol{s},\boldsymbol{a})\right]\\ &+\sum_n \mathbb{E}_{s_l,a_l\sim\mathcal{B}}\left[\nabla_{\theta_l}\pi_l\left(a_l|s_l\right)\nabla_{a_l}Q^{\pi_l}_{\phi_{l,n}}\left(s_l,a_l\right)\right],\end{aligned} \tag{23}$$

$$\mathcal{L}(\phi_{l,n}) = \mathbb{E}_{s_l,a_l,r_l,s'_l}\left[\left(Q^{\pi_l}_{\phi_{l,n}}\left(s_l,a_l\right) - y_{l,n}\right)^2\right], \tag{24}$$

$$y_{l,n} = r_{l,n} + \gamma Q^{\pi_l}_{\phi'_{l,n}}\left(s'_l, a'_l\right)\Big|_{a'_l=\pi'_l(s'_l)}, \tag{25}$$

where $n = 1, 2$, and the global critics are still updated as in (19). Based on this design, the number of local critics is increased from 1 to $N = 2$. The two local critics for two tasks with two sub-Q-functions work jointly to move towards the best estimation of the overall Q-function, $Q^{\pi_l}_{\phi_l}(s_l,a_l) = \sum_n Q^{\pi_l}_{\phi_{l,n}}(s_l,a_l)$, and hence the best update for the policy π_l of the local actor. The structure of DE-MADDPG-TDec is similar to Algorithm 1, the only difference is that for each agent, an extra **for** loop is added for the N tasks.

TABLE I: Simulation Parameters.

Parameters	Value
Carrier frequency	2 GHz
Number of resource blocks	3
Resource block bandwidth	180 kHz
PL maximum power	$p^{max} = 30$ dBm
Noise power	$\sigma^2 = -114$ dBm
CAM size	$D_l = 4$ KB [9]
V2V time limitation	$T = 100$ ms [6]
V2V gap	25 m
Fast fading update period	1 ms [2]
Slow fading update period	100 ms [2]

V. Numerical Results

In this section, we present the simulation results demonstrating the performance of the proposed algorithms. We use a single-cell urban C-V2X network operating at 2 GHz with 3 resource blocks, adhering to the urban specifications outlined in 3GPP TR 36.885 [2]. The primary simulation parameters are detailed in Table I. The algorithms evaluated include:

- *DE-MADDPG (TD3)*, *DE-MADDPG-TDec (TD3)*.
- *Decentralised MADDPG (Dec-MADDPG)*: No global critic, agents functioning independently.
- *Baseline-DDPG*: Centralised actor-critic network.
- *E-X*: Corresponding energy-concerned versions.

We use Python with PyTorch to implement our MADRL framework. The structure of our deep neural network consists of two hidden layers for the local actor (1024, 512 neurons) and critic (512, 256 neurons), and three hidden layers for the global critic (1024, 512, 256 neurons). The activation function and optimiser are chosen as the rectified linear unit and Adam. The learning rates of the actor and critic networks are set as 0.0001 and 0.001, while the target soft update parameter τ and the discount factor γ are set as 0.005 and 0.99.

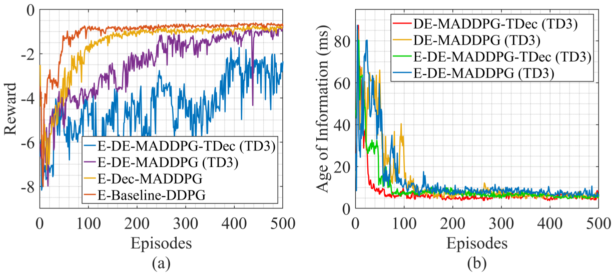

Fig. 2: Performance comparison ($P = 5, V_l = 4$). (a) Average reward of the energy-concerned algorithms. (b) AoI convergence with and without energy consideration.

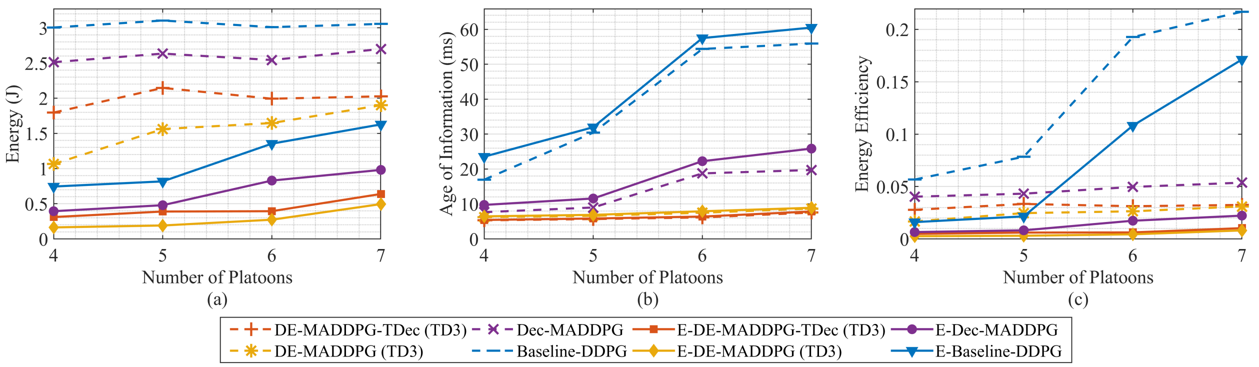

Fig. 3: Algorithms performance comparison (in the last 100 episodes). (a), (b) Average total energy consumption and AoI level of platoons. (c) Energy consumed per decreased AoI, calculated as $E/(T - AoI)$, describes the efficiency of using energy.

Fig. 2(a) shows the performance of the four energy-concerned algorithms in terms of reward convergence speed. It is clear that the proposed algorithms outperform the other baseline algorithms. Due to different designs in the reward functions, we compare the AoI convergence speed of our algorithms with those proposed in [6] in Fig. 2(b) for a fair comparison. For both DE-MADDPG (TD3) and DE-MADDPG-TDec (TD3), our energy-concerned algorithms converge at similar times and to similar levels.

Fig. 3 shows the algorithm performance for various platoon numbers ($P = 4, 5, 6, 7$) over the last 100 episodes, with most algorithms stabilising. In Fig. 3(a), our energy-focused algorithms significantly reduce the energy use for the two proposed algorithms, with a 68.47% to 87.79% decrease compared to their non-energy-focused counterparts. While they exhibit a slight increase in AoI—between 2.02% and 5.48%—the overall performance remains comparable, as illustrated in Fig. 3(b). In Fig. 3(c), using the efficiency metric $E/(T - AoI)$, our proposed energy-focused solutions demonstrate superior energy efficiency in reducing AoI, which furthermore highlights their effectiveness.

VI. Conclusion

This paper presents a DRL-based optimal resource allocation scheme for a platoon-based C-V2X network at an intersection. Two MADRL algorithms, DE-MADDPG and DE-MADDPG-TDec with TD3 technique, have been proposed for the joint optimisation of the AoI, CAM delivery, and power-energy consumption. Numerical results have shown a remarkable decrease in the energy consumption of platoons compared with existing research, while keeping similar levels of algorithm convergence speed and AoI level. The metric of energy efficiency also demonstrates that our algorithms are much more energy-saving.

References

[1] Y. Sun, Y. Hu, H. Zhang, H. Chen, and F.-Y. Wang, "A parallel emission regulatory framework for intelligent transportation systems and smart cities," *IEEE Trans. Intell. Veh.*, vol. 8, no. 2, pp. 1017–1020, Feb. 2023.

[2] 3GPP, "Study on LTE-based V2X services," 3rd Generation Partnership Project (3GPP), Technical Report (TR) 36.885, 2016, version 14.0.0.

[3] ——, "Study on enhancement of 3GPP support for 5G V2X services," 3rd Generation Partnership Project (3GPP), Technical Report (TR) 22.886, 2018, version 16.2.0.

[4] S. Chen *et al.*, "Vehicle-to-everything (V2X) services supported by LTE-based systems and 5G," *IEEE Comm. Stand. Mag.*, vol. 1, no. 2, pp. 70–76, 2017.

[5] H. V. Vu *et al.*, "Multi-agent reinforcement learning for channel assignment and power allocation in platoon-based C-V2X systems," in *Proc. 2022 IEEE 95th Veh. Technol. Conf. VTC2022-Spring*, Helsinki, Finland, Jun. 2022, pp. 1–5.

[6] M. Parvini, M. R. Javan, N. Mokari, B. Abbasi, and E. A. Jorswieck, "AoI-aware resource allocation for platoon-based C-V2X networks via multi-agent multi-task reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 72, no. 8, pp. 9880–9896, Aug. 2023.

[7] M. Kim *et al.*, "Age of information based beacon transmission for reducing status update delay in platooning," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 11 306–11 310, Oct. 2022.

[8] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. 2011 8th Annu. IEEE Commun. Soc. Conf. Sens. Mesh Ad Hoc Commun. Netw.*, Salt Lake City, UT, USA, Jun. 2011, pp. 350–358.

[9] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, ETSI Std. EN 302 637-2, Apr. 2019.

[10] H. U. Sheikh and L. Bölöni, "Multi-agent reinforcement learning for problems with combined individual and team reward," in *Proc. Int. Joint Conf. Neural Netw.*, Glasgow, UK, Jul. 2020, pp. 1–8.

[11] C. Sun, W. Liu, and L. Dong, "Reinforcement learning with task decomposition for cooperative multiagent systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2054–2065, May 2021.

[12] S. Fujimoto, H. van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, Jul. 2018, pp. 1587–1596.

[13] T. Huang *et al.*, "A survey on green 6G network: Architecture and technologies," *IEEE Access*, vol. 7, pp. 175 758–175 768, Dec. 2019.

[14] Z. Xu and A. Petropulu, "A bandwidth efficient dual-function radar communication system based on a MIMO radar using OFDM waveforms," *IEEE Trans. Signal Process.*, vol. 71, pp. 401–416, Feb. 2023.

[15] D. P. Bertsekas, *Dynamic programming and optimal control: Volume I.* Belmont, MA, USA: Athena Sci., 1995.