An Explainable AI Framework for Dynamic Resource Management in Vehicular Network Slicing

Haochen Sun*, Yifan Liu*, Ahmed Al-Tahmeesschi*, Swarna Chetty*, Syed Ali Raza Zaidi†
Avishek Nag‡, and Hamed Ahmadi*,
*Institute for Safe Autonomy and School of Physics, Engineering and Technology, York, YO10 5DD, UK
†School of Electrical and Electronic Engineering, University of Leeds, Leeds, UK
‡School of Computer Science, University College Dublin, Dublin, Ireland

***Abstract*—Effective resource management and network slicing are essential to meet the diverse service demands of vehicular networks, including Enhanced Mobile Broadband (eMBB) and Ultra-Reliable and Low-Latency Communications (URLLC). This paper introduces an Explainable Deep Reinforcement Learning (XRL) framework for dynamic network slicing and resource allocation in vehicular networks, built upon a near-real-time RAN intelligent controller. By integrating a feature-based approach that leverages Shapley values and an attention mechanism, we interpret and refine the decisions of our reinforcement learning agents, addressing key reliability challenges in vehicular communication systems. Simulation results demonstrate that our approach provides clear, real-time insights into the resource allocation process and achieves higher interpretability precision than a pure attention mechanism. Furthermore, the Quality of Service (QoS) satisfaction for URLLC services increased from 78.0% to 80.13%, while that for eMBB services improved from 71.44% to 73.21%.**

***Index Terms*—Sixth Generation (6G), V2X communication, Resource allocation, Explainable AI (XAI), Network slicing, Reinforcement learning**

I. Introduction

Vehicular communication networks are poised to play a central role in 6G wireless systems, where ultra-high data rates, extremely low latency, and ubiquitous intelligence are envisioned as key enablers for future mobility and automation [**?**]. However, achieving reliable real-time communication for highly dynamic vehicular environments remains a major challenge. The increasing complexity of Vehicle-to-Everything (V2X) services demands efficient and dynamic resource management [1]. Conventional networks often fall short in addressing the challenges of vehicular services, which are marked by high mobility, fluctuating channel conditions, and diverse QoS requirements. For instance, safety-critical applications demand ultra-reliable low-latency communications (URLLC), while multimedia services rely on enhanced mobile broadband (eMBB). In response, network slicing has emerged as a pivotal technology that partitions a single physical infrastructure into multiple, logically isolated networks, each tailored to specific service demands [2].However, the operation of network slicing relies on efficient and precise resource allocation. While traditional mathematical optimization methods prove effective in static and predictable scenarios, they struggle to cope with the inherent dynamics, strict delay constraints, and frequent topology changes typical of vehicular environments [3].

Recent advancements in Deep Reinforcement Learning (DRL) have demonstrated significant promise for dynamically allocation resources in highly variable environments. For instance, algorithms based on Deep Deterministic Policy Gradient (DDPG) have shown both efficiency and adaptability in addressing multi-dimensional resource requirements through coordinated strategies among distributed agents [4]. Similarly, [5] proposed a multi-agent DRL framework leveraging dueling double deep Q-networks (D3QN) to effectively address joint user association and resource allocation problems in heterogeneous cellular networks, further demonstrating DRL's ability to handle non-convex, combinatorial resource management scenarios. However, despite their high performance, such DRL-based approaches often function as closed boxes, lacking the interpretability required to build trust and ensure reliability in safety-critical vehicular applications [6].

To mitigate these challenges, EXplainable AI (XAI) techniques have been developed to provide post-hoc explanations that enhance the transparency of model decisions [7]. Among these, the Shapley Values for Explaining Reinforcement Learning (SVERL) method offers a rigorous, game-theoretic foundation to elucidate the contributions of individual features to an agent's performance [8]. Nevertheless, the traditional SVERL approach suffers from significant computational overhead, making it unsuitable for real-time applications, especially in latency-sensitive vehicular networks.

To overcome this limitation, we propose an integrated framework that combines an attention mechanism with the SVERL methodology, yielding an efficient and interpretable DRL solution tailored for vehicular network slicing. Our approach guarantees high-quality real-time explainability by supervising the attention layer with approximate Shapley values during training. This integration ensures that the trained models can quickly generate both accurate resource allocation decisions and comprehensible explanations.

The primary contributions of this work include the development of a novel interpretable DRL framework that fuses an attention-based DDPG model with SVERL for dynamic vehicular network slicing and resource management. The proposed solution not only provides an interpretable decision-making process—thereby enhancing trust and transparency in highly dynamic vehicular environments—but also improves

QoS satisfaction for various V2X services.

II. System Model

A. Network Topology

We consider a vehicular network deployed over a grid road topology with four primary intersections, as illustrated in Fig. 1. A set of gNodeBs (gNBs), denoted by $\mathcal{M}_{\text{all}} = 1, 2, \ldots, M$, is strategically deployed based on classical deployment strategies aimed at coverage optimization and interference minimization to ensure comprehensive coverage. A set of vehicles, $\mathcal{N} = \{1, 2, \ldots, N\}$, traverses the network along random paths while requesting services (e.g., URLLC or eMBB) according to their specific QoS demands [1], [2]. Time is discretized into slots $t \in \{1, 2, \ldots\}$, and the connection between vehicle i and gNB m at time t is indicated by the binary variable $q_{i,m}^t \in \{0, 1\}$, which satisfies

$$\sum_{m=1}^{M} q_{i,m}^t = 1, \quad \forall i \in \mathcal{N}. \tag{1}$$

This ensures that each vehicle is connected to exactly one gNB per time slot. Moreover, we assume that all gNBs are interconnected via low-latency wired backhaul links to a centralized Software Defined Network (SDN) controller, enabling efficient global resource management [4], [9].

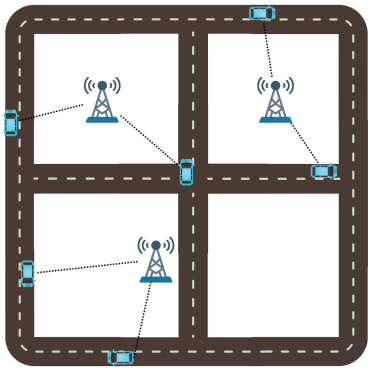

Fig. 1. Vehicular Network Environment

B. Communication Model

Communication between vehicles and gNBs is enabled by the allocation of Physical Resource Blocks (PRBs) rather than direct bandwidth assignment. In 5G NR, a PRB comprises 12 consecutive subcarriers, with its bandwidth determined by the subcarrier spacing (e.g., 360 kHz for a 30 kHz spacing) [10]. The instantaneous throughput for vehicle i connected to gNB m at time t is given by:

$$R_{i,m}^t = B_{i,m} \cdot 3.6 \times 10^5 \cdot \log_2\left(1 + \frac{P\, G_{i,m}^t}{\sigma^2 + I_m^t}\right), \tag{2}$$

where $B_{i,m}$ denotes the number of allocated PRBs, P is the transmission power, σ^2 represents the noise power, and I_m^t is the interference power. For a vehicle with data demand D_i^t, the total communication delay is calculated as:

$$T_{i,m}^t = \frac{D_i^t}{R_{i,m}^t} + T_{\text{processing}} + T_{\text{scheduling}}, \tag{3}$$

with $T_{\text{processing}} + T_{\text{scheduling}}$ fixed at 1.5 ms, in accordance with relevant technical standards [11].

C. Slice Assignment Model

To address the heterogeneous service requirements in vehicular networks, vehicle services are categorized into two network slices: URLLC for latency-sensitive, safety-critical applications and eMBB for high data-rate applications [12], [13]. The service requests for each vehicle are denoted by the binary variables s_i^{URLLC} and s_i^{eMBB} which are generated randomly upon the vehicle's entry into the vehicular network environment and remain constant throughout its journey. For vehicles requesting URLLC services, the delay must remain below the maximum threshold T_{th}:

$$s_i^{\text{URLLC}} \cdot T_{i,m}^t \le s_i^{\text{URLLC}} \cdot T_{\text{th}}, \quad \forall i \in \mathcal{N}, \tag{4}$$

while for vehicles requiring eMBB services, the throughput must meet a minimum threshold R_{th}:

$$s_i^{\text{eMBB}} \cdot R_{i,m}^t \ge s_i^{\text{eMBB}} \cdot R_{\text{th}}, \quad \forall i \in \mathcal{N}. \tag{5}$$

Since vehicles may simultaneously request both services, this flexible slice assignment effectively accommodates diverse QoS requirements.

III. Problem Formulation

The primary objective of this study is to minimize the overall QoS violations experienced by all vehicles while ensuring that diverse service requirements are met and that the model's resource allocation decisions remain interpretable. In this work, QoS violations are quantified as penalties incurred under two conditions: for vehicles in the URLLC slice, a penalty is imposed when the communication delay $T_{i,m}^t$ exceeds the maximum allowable latency $T_{\max}$; for vehicles in the eMBB slice, a penalty is incurred when the achieved throughput $R_{i,m}^t$ falls below the required threshold $R_{\min}$. Consequently, the optimization problem is formulated as:

$$\begin{aligned} \min_{\{B_{i,m}^t,\, q_{i,m}^t\}} \quad & \sum_{t=1}^{T} \sum_{i=1}^{N} \Big[s_i^{\text{URLLC}} \max\big(0,\, T_{i,m}^t - T_{\text{th}}\big) \\ & + s_i^{\text{eMBB}} \max\big(0,\, R_{\text{th}} - R_{i,m}^t\big) \Big], \end{aligned} \tag{6}$$

s.t.:

$$\sum_{i \in \mathcal{N}} B_{i,m}^t \le W_m, \quad \forall m \in \mathcal{M}, \forall t, \tag{7}$$

$$\sum_{m \in \mathcal{M}} q_{i,m}^t = 1, \quad q_{i,m}^t \in \{0, 1\}, \quad \forall i \in \mathcal{N}, \forall t, \tag{8}$$

$$B_{i,m}^t \le W_m \cdot q_{i,m}^t, \quad \forall i \in \mathcal{N}, \forall m \in \mathcal{M}, \forall t. \tag{9}$$

Constraint (7) ensures that the total number of PRBs allocated by any gNB does not exceed its capacity W_m.

Constraint (8) enforces that each vehicle is connected to exactly one gNB each slot. Finally, Constraint (9) links the resource allocation directly to the active connection. Due to the inherent complexity of vehicular network slicing—marked by dynamic mobility, fluctuating channel conditions, and unpredictable traffic—traditional optimization techniques are often impractical. Thus, we adopt a DRL framework that continuously interacts with the network environment, enabling real-time, data-driven resource management without requiring explicit forecasting [14]. While our DRL-based framework inherently avoids explicit long-term predictions of future network conditions or complicated state forecasting, we utilize readily available short-term mobility information, such as the next timestep vehicle positions, to enhance state representation without incurring significant computational overhead.

IV. MDP Formulation

To address vehicular networks' dynamic and uncertain nature, we formulate the resource allocation and slice assignment problem as a Markov Decision Process (MDP). The key components of the MDP are described below.

1) State (S): The system state at each discrete time slot t is represented by a comprehensive feature vector that includes both vehicle-specific and network-wide information:

$$S_t = \Big\{(x_i^t,\, y_i^t,\, D_i^t,\, s_i^{\text{URLLC}},\, s_i^{\text{eMBB}},\, a_i^t,\, q_{i,m}^{t-1},\, B_{i,m}^t,\, G_{i,m}^t, \\ x_i^{\text{next},t},\, y_i^{\text{next},t},\, v_i^t) \,\big|\, i \in \mathcal{N},\, m \in \mathcal{M}\Big\} \cup \Big\{L_m^t \,\big|\, m \in \mathcal{M}\Big\}, \tag{10}$$

where x_i^t and y_i^t are the normalized coordinates of vehicle i, D_i^t represents its normalized URLLC data demand, and s_i^{URLLC} and s_i^{eMBB} indicate the service requests. The binary variable a_i^t denotes the activity status of vehicle i. Additionally, $q_{i,m}^{t-1}$ and $B_{i,m}^t$ provide historical context by representing the previous vehicle-to-gNB association and current resource allocation, respectively. The channel gain $G_{i,m}^t$ reflects the wireless link quality, while v_i^t is speed of vehicle which can capture next time step vehicle's position $x_i^{\text{next},t}$, $y_i^{\text{next},t}$, Finally, L_m^t denotes the normalized load of gNB m, which assists the agent in anticipating congestion.

2) Action (A): At each time slot t, the RL agent selects an action that comprises two components: the vehicle-to-gNB association and the allocation of resource blocks. The action space is defined as:

$$A_t = \Big\{(q_{i,m}^t,\, B_{i,m}^t) \,\big|\, i \in \mathcal{N},\, m \in \mathcal{M}\Big\}, \tag{11}$$

where $q_{i,m}^t \in [0,1]$ is a relaxed association variable and $B_{i,m}^t \in [0, W_m]$ represents the allocated resources. While practical systems require binary connectivity and discrete resource allocation, the relaxation facilitates gradient-based optimization during training [15].

3) Reward (R): The reward function guides the RL agent to minimize QoS violations while efficiently utilizing network resources. At each time slot t, the immediate reward is defined as:

$$r_t = -\Bigg(\sum_{i\in\mathcal{N}} s_i^{\text{URLLC}}\, \omega_{\text{URLLC}}\, \max(0,\, T_{i,m}^t - T_{\text{th}}) \\ + \sum_{i\in\mathcal{N}} s_i^{\text{eMBB}}\, \omega_{\text{eMBB}}\, \max(0,\, R_{\text{th}} - R_{i,m}^t)\Bigg), \tag{12}$$

where ω_{URLLC} and ω_{eMBB} are weighting factors that emphasize the relative importance of latency and throughput requirements. The environment inherently enforces resource constraints; the reward function primarily focuses on penalizing QoS deviations.

Together, these MDP components allow the DRL agent to iteratively improve its resource allocation decisions under the complex dynamics of vehicular networks.

V. Proposed XRL Algorithm

To enhance the transparency and reliability of resource allocation decisions in V2X slicing environments, we propose an explainable deep reinforcement learning (XRL) framework that integrates an attention mechanism with offline Shapley value estimation. By integrating our XRL solution into the near-RT RIC framework, we aim to achieve both improved system performance and transparent decision-making, crucial for safety-critical V2X scenarios. The Shapley Value for Explaining Reinforcement Learning (SVERL) method introduced in [8] rigorously quantifies the contributions of individual state features based on game-theoretic principles; however, due to the need for exhaustive enumeration or Monte Carlo sampling over all feature subsets, the computational overhead of the SVERL method is not suitable for real-time applications such as V2X. In contrast, our proposed method employs a lightweight attention module to enable real-time interpretability while leveraging offline Shapley value estimation to ensure theoretical soundness of internal feature importance. This design significantly reduces the computational burden while preserving high explanation fidelity. The overall architecture of the proposed framework is illustrated in Fig. 2.

A. XAI Integration

For a given input state vector $\mathbf{s} \in \mathbb{R}^d$ with d representing the total number of state features, our model first computes an attention weight vector via a linear transformation and softmax activation:

$$\boldsymbol{\alpha} = \text{softmax}(W\mathbf{s} + b), \tag{13}$$

where $W \in \mathbb{R}^{d\times d}$ and $b \in \mathbb{R}^d$ are trainable parameters. The state vector is then reweighted element-wise as

$$\tilde{\mathbf{s}} = \boldsymbol{\alpha} \odot \mathbf{s}, \tag{14}$$

with $\odot$ denoting the Hadamard product. During inference, $\boldsymbol{\alpha}$ directly explains feature importance.

To further enhance interpretability, we approximate the Shapley value for each feature. Given the feature set $F = \{1, 2, \ldots, d\}$ and any subset $C \subseteq F$, we define a characteristic

function based on the expected return when only the features in C are observed:

$$v(C) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_{t+1} \,\middle|\, s_0 = \mathbf{s}, \text{ observing only } C\right]. \quad (15)$$

The Shapley value for the ith feature is then given by

$$\psi_i = \sum_{C \subseteq F \setminus \{i\}} \frac{|C|!(|F| - |C| - 1)!}{|F|!} \Big[v(C \cup \{i\}) - v(C)\Big], \quad (16)$$

and is approximated via Monte Carlo sampling:

$$\psi_i \approx \frac{1}{M} \sum_{j=1}^{M} \Big[v(C_j \cup \{i\}) - v(C_j)\Big], \quad (17)$$

where each C_j is randomly sampled from $F \setminus \{i\}$, and M is the number of samples [8].

To fuse the attention mechanism with the Shapley value estimation, we align their normalized values by defining

$$\hat{\alpha}_i = \frac{\alpha_i}{\sum_{k=1}^{d} \alpha_k} \quad \text{and} \quad \hat{\psi}_i = \frac{\psi_i}{\sum_{k=1}^{d} \psi_k}.$$

The auxiliary explanation loss is then defined as

$$L_{\text{explain}} = \frac{1}{d} \sum_{i=1}^{d} \left(\hat{\alpha}_i - \hat{\psi}_i\right)^2. \quad (18)$$

Finally, the overall loss function combines the standard DDPG loss with the explanation loss and is calculated as:

$$L_{\text{total}} = L_{\text{DDPG}} + \lambda\, L_{\text{explain}}, \quad (19)$$

where λ is a hyperparameter controlling the trade-off between performance and explainability.

Algorithm 1: Attention-DDPG-SVERL

Input: Environment, networks, replay buffer, batch size K, Shapley threshold τ, evaluation interval N_{eval}, explanation loss weight λ
Output: Resource allocation decisions with interpretable feature weights
Initialize environment, networks, replay buffer, and SVERL module;
Embed an attention layer in the Actor network to output both action a_t and attention weights α_t at time t;
for *each episode* $e = 1$ *to* E **do**
  Reset environment to get initial state s_0;
  for *each time step* $t = 1$ *to* T **do**
    $(a_t, \alpha_t) \leftarrow \text{Actor}(s_t)$;
    Execute a_t, observe s_{t+1} and reward r_t;
    Store $\{s_t, a_t, r_t, s_{t+1}\}$ in replay buffer;
    $s_t \leftarrow s_{t+1}$;
    if *replay buffer size* $\geq K$ **then**
      Sample mini-batch from buffer and update Critic and Actor;
      Soft-update target networks;
      if $e \mod N_{eval} = 0$ **then**
        Sample a subset of states $\{s\}$;
        For each state, compute approximate Shapley values ψ via Monte Carlo rollouts;
        Compute explanation loss: $L_{\text{explain}} = \text{MSE}(\text{normalize}(\alpha), \text{normalize}(\psi))$;
        Update Actor network with total loss: $L_{\text{total}} = L_{\text{DDPG}} + \lambda L_{\text{explain}}$;
    Add decaying Gaussian noise to a_t and clip to $[0, 1]$;

B. O-RAN Near-Real-Time RIC for Resource Management and Control

Based on the Open Radio Access Network (O-RAN) architecture [16], we deploy a near-real-time (near-RT) RAN Intelligent controller (RIC) to manage dynamic resource allocation and network slicing for vehicular communications. Unlike conventional scheduling processes that operate at millisecond-level subframe intervals, the near-RT RIC performs strategic resource management at coarser intervals, typically on the order of one second [17]. Specifically, at each decision interval t, the RIC observes the global network state S_t (as defined in (10)) and produces an action A_t (as described in (11)), which specifies the vehicle-to-gNB association variable $q_{i,m}^t$ and the resource block allocation decision $B_{i,m}^t$.

These high-level decisions are executed by lower-layer scheduling mechanisms within each decision interval. At the start of every new interval, the near-RT RIC collects updated state information reflecting changes in vehicle mobility, channel conditions, and service demand. This continuous feedback loop enables the RIC to dynamically re-optimize the slicing and resource allocation strategies.

By aggregating fine-grained millisecond-level scheduling into coarser-interval strategic decisions, this approach significantly reduces the computational complexity of the near-RT RIC, maintaining scalability and responsiveness even in rapidly changing network conditions. Moreover, it aligns with the O-RAN objective of decoupling control-plane and user-plane functionalities, thereby enabling centralized and efficient network slicing orchestration.

VI. Results

We consider a V2X network with an area of 1000m×1000m consisting of 3 gNBs, and the rest of the settings are shown in Table I.

A. QoS Satisfaction Performance

Fig. 3 provides a comparative analysis of QoS satisfaction rates for URLLC and eMBB services under four different approaches: (i) random algorithm, (ii) standard DDPG, (iii) DDPG with an attention module, and (iv) our proposed method combining DDPG with attention and SVERL.

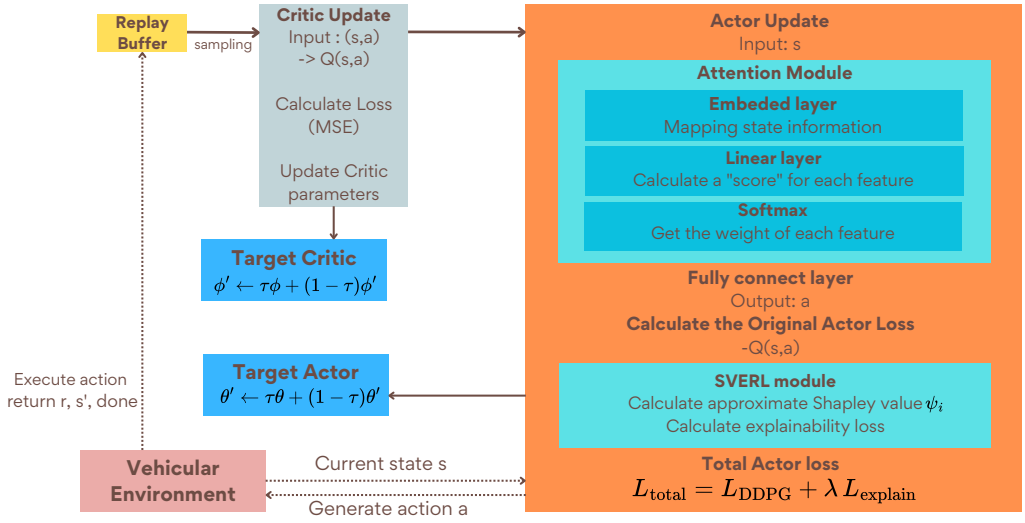

Fig. 2. Proposed XRL Framework Overview

TABLE I
LIST OF PARAMETERS AND VALUES

Parameters	Value/Setting
Number of Vehicles	5
Resource Blocks per BS	273
Transmission Power	50dBm
Noise Power	1.4×10^{-15} W
Resource Block Bandwidth	360 kHz
Maximum Delay for URLLC	0.015 sec
Minimum Data Rate Requirement for eMBB	20 Mbps
Vehicle Velocity	15m/s
Path Loss Exponent	3.5
URLLC Data Requirement	3×10^5 bits

In addition to the three learning-based methods, the random baseline achieves only 33.04% for URLLC and 34.44% for eMBB, which is substantially lower than any Reinforcement Learning (RL)-based approach. The results indicate that our proposed combined attention and SVERL mechanism attains the highest QoS satisfaction, reaching 80.13% for URLLC and 73.21% for eMBB, marking an effective improvement over the pure DDPG model. Furthermore, integrating SVERL with attention outperforms the attention-augmented DDPG variant, implying that the explainability component provides transparency and bolsters overall performance.

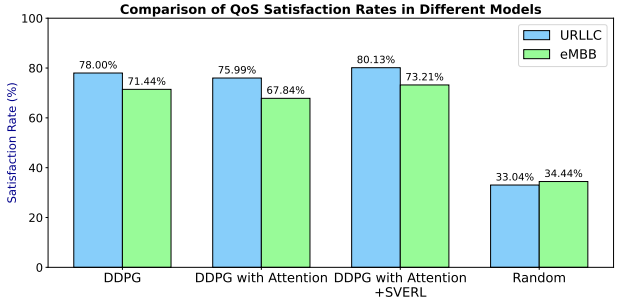

Fig. 3. QoS Satisfaction Comparison for URLLC and eMBB Services

B. Explainability Analysis

Figure 4 presents a merged visualization of the attention weights generated by our attention-DDPG-SVERL method. Given the large number of state features and the very low weights for most, we focus only on the ten most significant features for brevity and clarity. The blue panel of the figure shows the average attention weights over the entire test set, while the orange panel displays the attention distribution at a specific time step.

Here, the features align with the state, as shown in Eq. 10. These distributions exhibit a consistent trend in which location-related and predicted mobility features (such as current and next position coordinates) receive higher weights, indicating that the model prioritizes spatial information and mobility patterns to adapt resource allocation decisions in real time. We can directly interpret the agent's behavior by examining the resulting weight values. For instance, a high weight assigned to the next time slot's coordinates reflects the agent's attempt to adjust proactively to maintain channel condition and minimize QoS violations. This emphasis on current and anticipated vehicular positions aligns well with the overarching goal of minimizing QoS violations in a dynamic vehicular environment.

C. Explanation Fidelity Evaluation

To evaluate our proposed model's interpretability and explanation fidelity, we employ the Pearson correlation coefficient as a quantitative metric, measuring the relationship between feature perturbations and the resulting changes in the model's output [18]. Specifically, we systematically perturb each feature, measure the change in the model's output, and compare it to the assigned feature importance to assess how well the explanations align with the model's actual behavior. A higher correlation coefficient indicates that the explanations accurately reflect the model's internal decision-making process.

Table. II compares the explanation fidelity between a purely attention-based model and our combined attention-SVERL approach. The Pearson correlation coefficient of our integrated

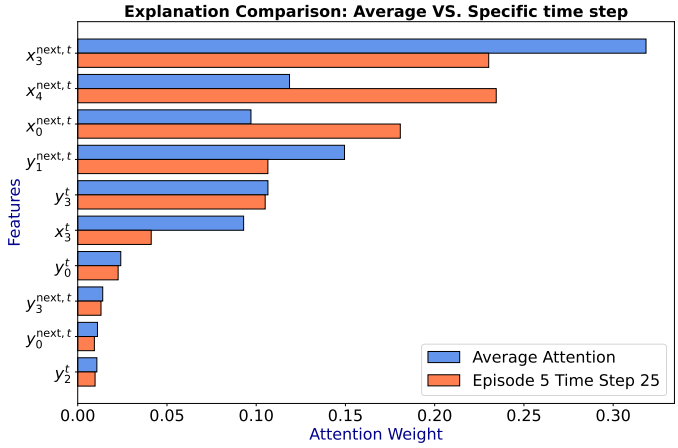

Fig. 4. Feature Weights Distribution across Test Episodes VS Specific Time Slot

attention and SVERL model significantly exceeds the pure attention model. This notable improvement demonstrates that aligning attention weights with Shapley value estimations effectively enhances the transparency and accuracy of feature importance explanations.

The increased fidelity offered by our explainability-enhanced model enables a more reliable and intuitive understanding of the resource management policy — an essential requirement for trustworthy deployment in safety-critical vehicular communication networks.

TABLE II
COMPARISON OF EXPLANATION FIDELITY BETWEEN PURE ATTENTION AND ATTENTION-DDPG-SVERL MODELS

Method	Mean Correlation
Attention+SVERL	0.3054
PureAttention	0.1708

VII. CONCLUSION

In this paper, we proposed an interpretable DRL framework for dynamic network slicing and resource allocation in vehicular networks. By integrating an attention mechanism with offline Shapley value estimation, our approach achieves high QoS satisfaction for URLLC and eMBB services, while also enabling real-time interpretability of the decision-making process. Simulation results demonstrate that the proposed method effectively balances resource allocation and explains the model's behavior transparently. Leveraging theoretically grounded Shapley values, along with an auxiliary loss to align feature importance, further enhances the robustness of the model. Overall, our integrated framework addresses key challenges in dynamic vehicular network management and offers a solid foundation for future research on scalable, interpretable, and reliable resource management solutions.

REFERENCES

[1] H. Sun, Y. Liu, A. Al-Tahmeesschi, A. Nag, M. Soleimanpour, B. Canberk, H. Arslan, and H. Ahmadi, "Advancing 6g: Survey for explainable ai on communications and network slicing," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 1372–1412, 2025.

[2] L. U. Khan, I. Yaqoob, N. H. Tran, Z. Han, and C. S. Hong, "Network slicing: Recent advances, taxonomy, requirements, and open research challenges," *IEEE Access*, vol. 8, pp. 36 009–36 028, 2020.

[3] R. Su, D. Zhang, R. Venkatesan, Z. Gong, C. Li, F. Ding, F. Jiang, and Z. Zhu, "Resource allocation for network slicing in 5g telecommunication networks: A survey of principles and models," *IEEE Network*, vol. 33, no. 6, pp. 172–179, 2019.

[4] Y. Cui, H. Shi, R. Wang, P. He, D. Wu, and X. Huang, "Multi-agent reinforcement learning for slicing resource allocation in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 2, pp. 2005–2016, 2023.

[5] N. Zhao, Y.-C. Liang, D. Niyato, Y. Pei, M. Wu, and Y. Jiang, "Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5141–5152, 2019.

[6] F. Tang, B. Mao, N. Kato, and G. Gui, "Comprehensive survey on machine learning in vehicular network: Technology, applications and challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 2027–2057, 2021.

[7] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan *et al.*, "Explainable ai (xai): Core ideas, techniques, and solutions," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–33, 2023.

[8] D. Beechey, T. M. Smith, and Ö. Şimşek, "Explaining reinforcement learning with shapley values," in *International Conference on Machine Learning*. PMLR, 2023, pp. 2003–2014.

[9] D. Das and D. Das, "Radio access technology selection in sdn controlled reconfigurable base station," *Computers & Electrical Engineering*, vol. 66, pp. 260–273, 2018.

[10] 3rd Generation Partnership Project (3GPP), "3GPP TS 38.211 V16.3.0 (2020-11): NR; Physical channels and modulation," ETSI, Tech. Rep., 2020. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138200_138299/138211/16.03.00_60/ts_138211v160300p.pdf

[11] 3GPP, "Technical Specification Group Services and System Aspects; Enhancement of 3GPP support for V2X scenarios; Stage 1 (Release 16)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 22.186, June 2019. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/22_series/22.186/22186-g20.zip

[12] International Telecommunication Union, "IMT Vision – Framework and overall objectives of the future development of IMT for 2020 and beyond," International Telecommunication Union, Recommendation M.2083-0, Sep. 2015. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/m/r-rec-m.2083-0-201509-i!!pdf-e.pdf

[13] H. Khan, P. Luoto, S. Samarakoon, M. Bennis, and M. Latva-Aho, "Network slicing for vehicular communication," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, p. e3652, 2021.

[14] J. A. Hurtado Sánchez, K. Casilimas, and O. M. Caicedo Rendon, "Deep reinforcement learning for resource management on network slicing: A survey," *Sensors*, vol. 22, no. 8, p. 3031, 2022.

[15] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," *arXiv preprint arXiv:1611.00712*, 2016.

[16] "O-RAN: Towards an Open and Smart RAN," https://mediastorage.o-ran.org/white-papers/O-RAN.White-Paper-2018-10.pdf, o-RAN Alliance White Paper, Oct. 2018. Accessed: Month Day, Year.

[17] O-RAN Alliance, "O-RAN: Use Cases and Deployment Scenarios," O-RAN Alliance, Tech. Rep., February 2020. [Online]. Available: https://mediastorage.o-ran.org/white-papers/O-RAN.WG1.Use-Cases-and-Deployment-Scenarios-White-Paper-2020-02.pdf

[18] P. Sedgwick, "Pearson's correlation coefficient," *Bmj*, vol. 345, 2012.