

Delay-Energy Tradeoff in Actor-Critic-Based Relay Selection for Multi-Hop V2X-MEC Networks

Sheng-Zhi Huang*, Shiang-Ming Huang*, Kun-Yu Lin*, and Chih-Lin Hu†

*Department of Computer Science and Information Engineering, Tamkang University, New Taipei City, Taiwan

†Department of Communication Engineering, National Central University, Taoyuan, Taiwan

Email: 167320@o365.tku.edu.tw, 613411155@o365.tku.edu.tw, 168920@o365.tku.edu.tw, clhu@ce.ncu.edu.tw

Abstract—This paper investigates the dynamic relay selection problem in multi-hop Vehicle-to-Everything (V2X) and Multi-access Edge Computing (MEC) environments. While existing reinforcement learning approaches primarily focus on minimizing end-to-end latency, the consequential impact on energy consumption and load distribution remains under-explored. In this work, we build a high-fidelity SUMO-MEC co-simulation platform and implement an actor-critic learning framework that follows the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) architecture with Centralized Training and Decentralized Execution (CTDE). By utilizing Gumbel-Softmax reparameterization, the discrete relay selection is optimized with continuous gradients. Empirical results from 50 independent runs show that the agent attains the lowest latency by rationally concentrating tasks on high-performance Roadside Units (RSUs), thereby incurring an explicit, quantifiable delay–energy tradeoff and reduced load fairness. A value-based baseline (DQN) matches the latency closely in this small discrete action space, while an on-policy baseline (A2C) reaches low energy and high fairness only by converging to a relay-avoiding policy. These behavioral insights, rather than a single headline metric, offer guidelines for robust edge intelligence under strict QoS constraints.

Index Terms—Vehicle-to-Everything (V2X), Multi-access Edge Computing (MEC), Multi-Agent Deep Deterministic Policy Gradient (MADDPG), Delay-Energy Tradeoff, Relay Selection.

I. INTRODUCTION

Vehicle-to-Everything (V2X) communication is turning vehicles into networked nodes that sense, communicate, and cooperate. Applications such as cooperative perception, map updates, and real-time analytics require computationally intensive tasks to be processed under tight latency budgets. Multi-access Edge Computing (MEC) offloads computation from resource-constrained vehicles to nearby edge servers, avoiding the round-trip latency of remote clouds [1]. In urban traffic, however, the source and destination are often beyond direct range or the nearest server is congested. A multi-hop relay model $A \rightarrow B \rightarrow C$, in which an intermediate node B executes the task and forwards the result, therefore captures practical V2X offloading more faithfully than the single-hop abstraction [2], [3]. Vehicle mobility, changing topology, and time-varying edge load make this a dynamic decision rather than a static assignment.

Most reinforcement-learning (RL) offloading studies optimize end-to-end latency in isolation [5], [6]. Yet the effect of an aggressively latency-minimizing policy on energy and load distribution—precisely the quantities determining

deployability—is seldom examined. A policy that drives a few high-frequency RSUs to shave latency may look optimal while being energy-inefficient in operation.

This paper does not aim to claim a new state-of-the-art result. Using a high-fidelity SUMO-MEC co-simulation and a MADDPG-style agent with CTDE and Gumbel-Softmax discrete relay selection [7]–[9], we characterize the *learned behavior* of the agent over 50 independent runs and show, with statistical validation, that the lowest-latency policy is the one that rationally trades energy for delay by concentrating tasks on high-CPU RSUs. Recent surveys document the rapid adoption of single- and multi-agent deep RL for vehicular edge offloading [1]; representative methods include attention-augmented MADDPG for NOMA-based vehicular MEC [4], MADDPG-based joint task partitioning and resource allocation [5], many-to-many multi-agent offloading in vehicular fog computing [6], and mobility-aware multi-hop relay selection [2], [3]. These studies optimize latency or utility directly, whereas the behavioral consequences of such optimization are the focus of this work.

The main contributions of this paper are summarized as follows.

- We build a traffic-realistic SUMO [11] co-simulation platform and formulate multi-hop relay selection over heterogeneous edge nodes as an MDP with a delay-prioritized composite reward.
- We implement a MADDPG-style CTDE agent in which Gumbel-Softmax reparameterization enables deterministic policy gradients over the discrete relay-selection action space.
- Through a statistically validated evaluation (50 seeds, Mann-Whitney U tests) against DQN and A2C, we quantify an explicit delay–energy tradeoff rooted in the CMOS f^2 energy scaling, and show that favorable energy or fairness figures can arise from degenerate relay-avoiding behavior rather than sound decision-making.

II. SYSTEM CONCEPT AND REWARD DESIGN

We consider an urban V2X scenario in which connected vehicles continuously generate computation tasks that exceed their local computing budgets and must be offloaded. As illustrated in Fig. 1, a source vehicle A cannot always reach a capable edge server directly; the task is therefore relayed

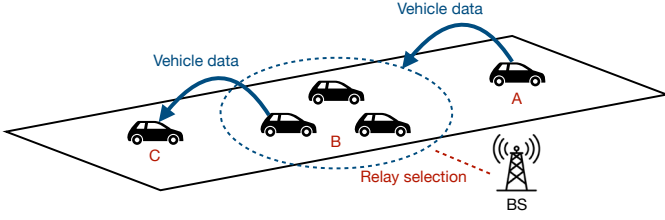


Fig. 1. Multi-hop V2X-MEC relay scenario: a task from source A is executed by a selected relay B and forwarded to destination C .

through an intermediate node B , which executes it and forwards the result to the destination C . The candidate relays are heterogeneous, and their channel conditions and queue backlogs change continuously. The central problem addressed in this paper is how to select, at each step, the relay that minimizes end-to-end latency, and what such a latency-driven policy implies for energy and load distribution.

A. Multi-Hop V2X-MEC Network Model

Computation offloading is abstracted as a multi-hop transmission model $A \rightarrow B \rightarrow C$. At each decision step t , a source vehicle (A) generates a task characterized by the tuple (s_i, c_i) , where s_i is the data size in bits and c_i the required CPU cycles. The task is transmitted to a selected relay node (B) for execution, and the result is forwarded to the destination vehicle (C). The core problem is to dynamically select the relay B from a candidate set $\mathcal{B} = \{b_1, \dots, b_M\}$ to minimize end-to-end latency under mobility. Consistent with this single-source instantiation, the learning system employs a centralized critic over the global state and a single actor that selects the relay; the multi-source, multi-user contention setting is left to future work.

B. Heterogeneous Relay Nodes

The candidate set $\mathcal{B}$ comprises two node types. Stationary MECs (S-MECs, i.e., RSUs) are deployed at fixed infrastructural points and offer high, stable computation capacity (f_s) but limited coverage. Mobile MECs (M-MECs) are computing resources mounted on surrounding vehicles, characterized by stochastic wireless links, time-varying availability, and lower capacity ($f_m \ll f_s$). This heterogeneity in compute and mobility drives the offloading-strategy variations analyzed here.

C. Delay and CMOS Energy Models

The end-to-end delay of task i , $T_i = T_i^{\text{tx}} + T_i^{\text{comp}} + T_i^{\text{queue}}$, comprises transmission, computation, and queuing components. The instantaneous wireless rate over a link of length d is modeled with distance attenuation and log-normal shadow fading $h \sim \text{LogNormal}(\mu_h, \sigma_h)$:

$$R(d) = \frac{1}{h} \max \left(R_{\min}, \frac{R_{\max}}{1 + (d/d_{\text{ref}})^2} \right), \quad (1)$$

where $R_{\max}$ is the peak rate, d_{ref} the reference distance, and $R_{\min}$ a protective floor. Transmission and computation delays follow $T_i^{\text{tx}} = s_i/R(d_{AB}) + s_i/R(d_{BC})$ and $T_i^{\text{comp}} = c_i/f_B$, while queuing delay $T_i^{\text{queue}} = Q_B(t)/f_B$ reflects the relay's

instantaneous backlog. High-frequency S-MECs reduce T_i^{comp} , but intrinsically raise power consumption per the CMOS dynamic-power model:

$$E_i^{\text{comp}} = \kappa \cdot f_B^2 \cdot c_i, \quad (2)$$

where κ is the energy coefficient. Since $E_i^{\text{comp}} \propto f_B^2$, selecting a high- f_B S-MEC sharply increases energy: doubling f_B halves computation delay but quadruples its energy. This is the physical basis of the delay-energy tradeoff.

D. Delay-Prioritized Composite Reward Design

The relay selection is cast as a Markov Decision Process (MDP) with a composite reward:

$$r_t = -\alpha T_t - \beta E_t - \lambda \mathbb{I}(T_t > T^{\text{QoS}}) + \mu R_t^{\text{margin}} + \delta J_{\text{fair},t}, \quad (3)$$

where T_t and E_t are the per-step latency and energy, the third term penalizes QoS violations, and the last two are a marginal-latency reward and Jain's fairness index for load balancing. To reflect the latency-sensitive nature of the target cooperative-computing applications, the weights are deliberately set delay-prioritized ($\alpha \gg \beta$; specifically $\alpha=1.0, \beta=0.15$). This configuration explicitly compels the agent to prioritize latency minimization, knowingly trading energy efficiency and load fairness—a design choice whose behavioral consequences are the subject of Section IV.

III. THE PROPOSED LEARNING APPROACH

A. Centralized Training with Decentralized Execution

To handle the dynamic V2X topology, we adopt a MADDPG-style architecture with CTDE [7]. During training, a centralized critic evaluates the action-value function $Q_\omega(s_t, \hat{a}_t)$ from the global state—the channel conditions and queue lengths of all candidate relays—which stabilizes learning. At execution, the actor selects relays from local observations alone, avoiding real-time state synchronization. Training stability is reinforced by experience replay, soft target updates, a warmup phase, and gradient clipping.

B. Differentiable Discrete Selection via Gumbel-Softmax

Conventional MADDPG targets continuous actions, whereas relay selection is strictly discrete (one node from $\mathcal{B}$). To let the deterministic policy gradient flow from critic to actor, we apply the Gumbel-Softmax reparameterization [9]. Given actor logits z_k for candidate k , the differentiable selection weight is

$$\hat{a}_k = \frac{\exp((z_k + g_k)/\tau_g)}{\sum_{j=1}^M \exp((z_j + g_j)/\tau_g)}, \quad g_k \sim \text{Gumbel}(0, 1), \quad (4)$$

where τ_g is a temperature. As training progresses, τ_g is exponentially annealed from 1.0 to 0.1, smoothly transitioning from near-uniform exploration to near-deterministic exploitation while preserving end-to-end differentiability. At execution time the action is taken as $a_t = \arg \max_k z_k$.

TABLE I
SIMULATION AND TRAINING PARAMETERS

Category	Value
Road network	600 × 600 m grid, 4 intersections
Vehicles	314 (max speed 15 m/s)
Relays	$N_s=2$ RSU (8,10 GHz); $N_m=3$ M-MEC $\mathcal{U}(3,7)$ GHz; $M=5$
Task	$s_i \sim \mathcal{U}(4 \times 10^6, 1.6 \times 10^7)$ bits; $c_i \sim \mathcal{U}(3,9) \times 10^9$ cyc
Channel	$R_{\max}=20$ Mbps, $d_{\text{ref}}=80$ m, $h \sim \text{LogN}(0, 0.25)$
Energy / Tx	$\kappa=10^{-28}$, $P_x=0.6$ W, $\gamma_q=0.85$
Reward weights	$\alpha=1.0$, $\beta=0.15$, $\lambda=2.0$, $\mu=0.5$, $\delta=0.3$
QoS threshold	$T^{\text{QoS}}=20$ s
Shared NN	hidden $H=256$, $\gamma=0.99$, grad clip 5.0
Proposed	lr (A/C) $10^{-4}/5 \times 10^{-4}$, $\tau_g : 1.0 \rightarrow 0.1$, warmup 3k steps
Protocol	50 seeds × 500 episodes; eval on last 35%

IV. EMPIRICAL INSIGHTS: THE DELAY-ENERGY TRADEOFF

A. Simulation Setup

The platform couples SUMO 1.11.0 with a PyTorch decision layer via TraCI. Key parameters are summarized in Table I. We compare the proposed agent against DQN [10], a representative value-based method, and A2C, an on-policy actor-critic method. All algorithms share an identical environment, observation and action space, reward, and a fixed traffic seed, ensuring a fair comparison. Each algorithm is run for 50 independent seeds (1000–1049) of 500 episodes; metrics are averaged over the final 35% of episodes (the converged regime), and differences are assessed with the two-sided Mann-Whitney U test.

B. Latency Performance vs. Energy Cost

Table II reports the core metrics. The proposed agent attains the lowest mean latency, 12.14 ± 0.82 s, a 12.0% reduction over A2C ($p < 10^{-11}$), together with the best task success rate (85.9%). Against DQN the latency difference (3.0%) is not statistically significant ($p=0.406$). We read this as an informative result rather than a shortfall: in a discrete space of only five candidates, DQN’s $\arg \max_a Q$ rule is already near-optimal once the Q -ranking is correct, so the precise policy gradient of an actor-critic offers limited benefit here, its advantage emerging in larger or continuous action spaces. The proposed agent nonetheless shows a smaller cross-seed standard deviation (0.82 vs 1.17) and a small but significant edge in effective rate ($p=0.036$).

This latency is purchased with energy: the proposed agent records the *highest* mean energy, 33.29 ± 5.76 J—43.3% above A2C and 6.6% above DQN. Fig. 2 and Fig. 3 make the tension explicit: the agent sits at the bottom of the latency curves and the top of the energy curves. This is not a defect but the rational outcome of the delay-prioritized reward ($\alpha/\beta \approx 6.7$): lowering one unit of delay is worth $6.7 \times$ more than lowering

TABLE II
PERFORMANCE COMPARISON (50 SEEDS, MEAN $\pm$ STD; p VS PROPOSED)

Metric	Proposed	DQN	A2C
Delay (s)↓	12.14 $\pm$ 0.82	12.52 $\pm$ 1.17	13.80 $\pm$ 0.02
Success (%)↑	85.9 $\pm$ 2.9	84.8 $\pm$ 4.1	80.8 $\pm$ 0.2
Rate (Mbps)↑	1.14 $\pm$ 0.06	1.11 $\pm$ 0.08	1.03 $\pm$ 0.00
Energy (J)↓	33.29 $\pm$ 5.76	31.23 $\pm$ 7.38	23.24 $\pm$ 0.04
Fairness↑	0.369 $\pm$ 0.18	0.524 $\pm$ 0.24	0.830 $\pm$ 0.00
RSU ratio	0.486	0.365	0.000
p (delay)	—	0.41 n.s.	3.7e−12
p (rate)	—	0.036 *	1.0e−15

Rate denotes the effective per-task rate (task bits over delay), a monotone transform of inverse delay; reported for completeness.

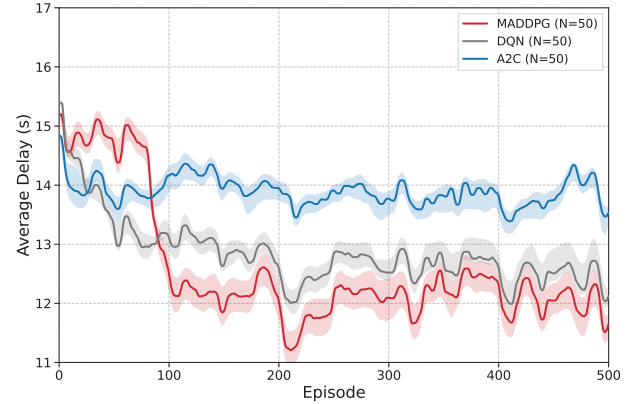


Fig. 2. Average end-to-end delay over training (50 seeds, mean $\pm$ 95% CI).

one unit of energy, so the agent rationally expends energy to reduce delay.

C. RSU Selection Ratio and Policy Behavior

The RSU selection ratio (Fig. 4) exposes the underlying mechanism. The proposed agent starts near the random baseline ($2/5=0.4$) and, around episode 80—coinciding with the end of critic warmup and the onset of temperature annealing—rises before stabilizing near 0.49. In other words, once the critic forms reliable Q -estimates, the actor learns to exploit the high-CPU RSUs to minimize delay. With RSU CPU frequencies roughly twice those of M-MECs and energy scaling as f^2 , an RSU costs about $4 \times$ the computation energy of an M-MEC; routing about half of the tasks to RSUs therefore directly explains the agent’s energy ranking. DQN settles at a moderate RSU ratio (0.3–0.4), which is consistent with its latency being close to the proposed agent’s.

A2C presents a contrasting pattern: its RSU ratio falls from ~ 0.35 to 0.000 by episode 50 and remains there, with near-zero cross-seed variance. We report this as an observation rather than a general property of A2C, and we do not exclude sensitivity to the training configuration or sample budget as a cause. Under the present setting, A2C converges to a policy that uses only the low-CPU M-MECs, which is why

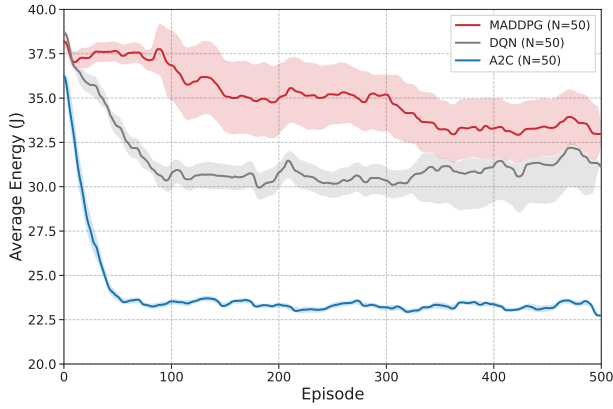


Fig. 3. Average energy consumption over training.

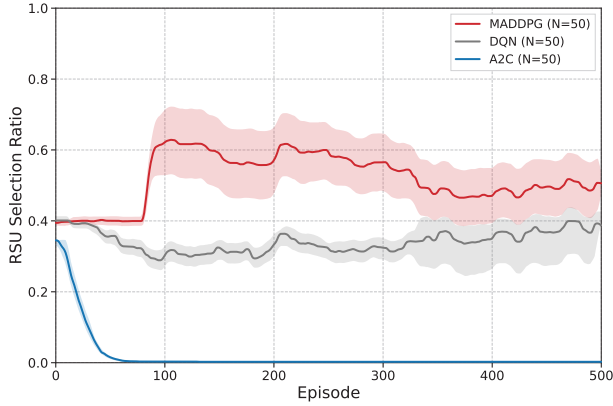


Fig. 4. RSU selection ratio over training.

it records the lowest energy (23.24J) and the highest Jain fairness (0.830). This carries a practical caution: low energy and high fairness are not, by themselves, evidence of a well-performing policy—here they coincide with the worst latency because the high-performance RSUs are simply left unused.

D. Scope and Limitations

Two boundaries of this study should be stated explicitly. First, the evaluation considers a single source–destination pair; the multi-source, multi-user contention regime—where inter-agent coupling is strongest—remains future work. Second, all algorithms were trained under an identical budget of 500 episodes, a protocol that structurally favors off-policy methods reusing transitions through replay; the behavior of the on-policy baseline should be interpreted within this sample budget rather than as an asymptotic property.

V. CONCLUSION

This paper characterized multi-hop V2X-MEC relay selection through the lens of behavior rather than headline latency. Under a deliberately delay-prioritized reward, a MADDPG-style agent with CTDE and Gumbel-Softmax achieves the lowest latency and best QoS satisfaction, but does so by rationally concentrating load on high-frequency RSUs, incurring the

highest energy and lowest fairness—an explicit, quantifiable delay–energy tradeoff. In this small discrete action space its latency is statistically indistinguishable from a strong value-based baseline, while a competing on-policy baseline reaches its favorable energy and fairness numbers only by leaving the high-performance RSUs unused. These results argue that energy and load fairness must be treated as first-class design constraints rather than incidental byproducts. Promising next steps are constrained RL with a hard energy cap ($E \leq E_{\max}$) to prevent RSU overload, and an extension to multi-agent, multi-user contention where the CTDE advantage should be most pronounced.

ACKNOWLEDGMENT

This work was supported in part by the National Science and Technology Council (NSTC), Taiwan, under Grant NSTC 111-2221-E-008-064.

REFERENCES

- [1] A. Uddin, A. H. Sakr, and N. Zhang, “Intelligent offloading in vehicular edge computing: A comprehensive review of deep reinforcement learning approaches and architectures,” *arXiv preprint arXiv:2502.06963*, 2025.
- [2] L. Liu, M. Zhao, M. Yu, M. A. Jan, D. Lan, and A. Taherkordi, “Mobility-aware multi-hop task offloading for autonomous driving in vehicular edge computing and networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2169–2182, February 2023.
- [3] M. Ahmed, S. Raza, H. Ahmad, W. U. Khan, F. Xu, and K. Rabie, “Deep reinforcement learning approach for multi-hop task offloading in vehicular edge computing,” *Engineering Science and Technology, an International Journal*, vol. 59, Article no. 101854, 2024.
- [4] L. Wu, J. Qu, S. Li, C. Zhang, J. Du, X. Sun, and J. Zhou, “Attention-augmented MADDPG in NOMA-based vehicular mobile edge computational offloading,” *IEEE Internet of Things Journal*, vol. 11, no. 16, pp. 27000–27014, 2024.
- [5] K. Lu, R.-D. Li, M.-C. Li, and G.-R. Xu, “MADDPG-based joint optimization of task partitioning and computation resource allocation in mobile edge computing,” *Neural Computing and Applications*, vol. 35, no. 22, pp. 16559–16576, 2023.
- [6] Z. Wei, B. Li, R. Zhang, X. Cheng, and L. Yang, “Many-to-many task offloading in vehicular fog computing: A multi-agent deep reinforcement learning approach,” *IEEE Transactions on Mobile Computing*, vol. 23, no. 3, pp. 2107–2122, March 2024.
- [7] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, “Multi-agent actor-critic for mixed cooperative-competitive environments,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, USA, 2017, pp. 6382–6393.
- [8] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Juan, Puerto Rico, 2016.
- [9] E. Jang, S. Gu, and B. Poole, “Categorical reparameterization with Gumbel-Softmax,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, Toulon, France, 2017.
- [10] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [11] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, “Microscopic traffic simulation using SUMO,” in *Proceedings of the 21st IEEE International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI, USA, 2018, pp. 2575–2582.