Scalable Energy-Aware Semantic Caching and Routing for Smart-City IoV Fleets: A Distributed Quantum Soft Actor-Critic Framework

James Adu Ansere*, Simon L. Cotton†, Joongheon Kim‡, Trung Q. Duong*†
*Memorial University, Canada, (e-mails:{jaansere, tduong}@mun.ca)
†Queen's University Belfast, UK, (e-mail: {simon.cotton, trung.q.duong}@qub.ac.uk)
‡Korea University, South Korea (e-mails: joongheon@korea.ac.kr)

***Abstract*—Rapid advancements in smart-city logistics and fleet management require strong solutions for optimizing energy efficiency and real-time decision-making in dynamic large-scale Internet-of-Vehicles (IoV) networks. Traditional methods for resource allocation and route optimization struggle with network uncertainty, time-varying traffic requirements, and energy constraints. This paper introduces a quantum soft actor-critic (Q-SAC) framework, an energy-aware quantum reinforcement learning algorithm, to optimize semantic resource caching and adaptive routing in large-scale IoV environments. By combining quantum computing with deep reinforcement learning, Q-SAC utilizes quantum superposition and parallelism to efficiently explore complex and high-dimensional continuous state spaces, allowing for swift responsiveness to dynamic network requirements and vehicular movement patterns. The framework uses semantic caching to prioritize context-relevant data, reducing latency and energy consumption. Simulation results in dynamic urban logistics show the proposed algorithm's superior performance and faster route convergence in uncertain environments.**

I. Introduction

Recent advances in semantic communication have demonstrated the potential to compress and transmit only the most meaningful content. In [1], a latency minimization problem was formulated to jointly optimize content caching, service placement, and computation task offloading to enhance the network performance. In [2], a demand prediction approach based on the spatio-temporal graph neural network was presented to obtain the optimal load with caching content in the 6G enabled IoV. An average delay minimization problem is formulated to jointly optimize execution mode, transmission path selection, and cache management with energy harvesting in vehicular networks [3]. These state-of-the-art methods fail to rapidly retrieve mission-critical and time-sensitive data, which increases latency, energy consumption, and reduces decision-making efficiency in high-density vehicular networks. In [4], a hierarchical SAC algorithm is presented to enhance dynamic performance of automated guided vehicles in a complex dynamic environment. In [5], a double bootstrapped SAC is proposed to accelerate the decision-making and convergence speed among autonomous vehicles. The complexity of these models in high-dimensional settings reduces algorithm performance in uncertain environments. Quantum algorithms have been applied in IoV ecosystems to improve semantic task processing as vehicles and decision variables grow exponentially with network size and complexity [6].

Despite advances in semantic-aware caching and routing for large-scale IoV systems, three issues remain in energy-constrained smart-city fleets [7]. First, cache policies overlook data relevance, treating important alerts like routine updates, wasting energy with unnecessary low-value transmissions [8]. Second, routing protocols focus solely on shortest paths or link quality, ignoring message importance and vehicle battery levels, risking downtime and reduced service [9]. Third, although quantum algorithms excel at large combinatorial problems, their use in real-time semantic caching and routing for IoV remains unexplored [10], [11].

To address these challenges, this paper presents the application of a Quantum-SAC (Q-SAC) algorithm for energy-aware semantic caching and routing in large-scale IoV networks, offering computational scalability, adaptability in complex IoV, and energy efficiency unmatched by traditional deep reinforcement learning approaches. This algorithm empowers smart-city IoV fleets to make intelligent, collaborative decisions, minimizing energy consumption while ensuring the timely and context-aware delivery of high-priority information under dynamic urban conditions.

II. System Model and Problem Formulation

We examine system models for semantic optimization in smart-city IoV and offer a mathematical formulation of the problem.

A. Mathematical System Model

We consider an IoV network consisting of vehicles, roadside units (RSUs), and an edge/cloud server. The system aims to optimize energy-sensitive semantic caching and resource allocation while ensuring efficient communication and mobility-sensitive connectivity. Let $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ be the set of vehicles, $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$ be the set of RSUs, $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$ be the set of contents that can be cached, $\mathcal{S} = \{s_1, s_2, \ldots, s_L\}$ be the set of edge/cloud servers, and $\mathcal{T}$ be the total time horizon divided into time slots τ. Vehicles dynamically move through the network requesting content, while RSUs and edge servers store semantic content representations to reduce redundant transmissions and save energy.

1) Semantic Modeling: We use a semantic-aware strategy to enhance data transmission, caching, and computation by emphasizing meaningful content over raw data. Our model incorporates semantic relevance, complexity s_c and transmission efficiency σ_{ij} between node i and j for task offloading and caching. The semantic complexity of content c_k at time τ, denoted as $s_c(c_k, \tau)$, evaluates the contextual relevance for specific tasks as:

$$s_c(c_k, \tau) = \lambda_1 U_k(\tau) + \lambda_2 P_k(\tau) + \lambda_3 T_k(\tau), \tag{1}$$

where $U_k(\tau)$ is the request frequency of content c_k, $P_k(\tau)$is its predicted future demand, $T_k(\tau)$ is its time sensitivity (e.g., real-time safety alerts), $\lambda_1, \lambda_2, \lambda_3$ are tunable weight parameters. The overall semantic task performance at time τ is defined as:

$$\mathcal{S}_p(\tau) = \varphi_1 \cdot s_c(\tau) + \varphi_2 \cdot \sigma_{ij}(\tau), \tag{2}$$

where φ_1, φ_2 are weighting factors that balance semantic complexity and efficiency. Tasks with higher complexity are prioritized using a semantic priority score is given by:

$$\mathcal{P}_{score} = \varphi \cdot s_c(\tau). \tag{3}$$

2) Dynamic Network Topology Model: We represent the IoV network as a dynamic graph $G_\tau = (\mathcal{V}, \mathcal{R}, \mathcal{E}_\tau)$, where $\mathcal{E}_\tau$ denotes the communication links that change over time τ. The link $(i, j) \in \mathcal{E}_\tau$ is determined by a probabilistic function that considers mobility, separation, and wireless channel conditions. The link capacity C_{ij} between the nodes i and j at time slot τ is given by:

$$C_{ij}(\tau) = B \log_2 \left(1 + \frac{P_{ij}(\tau) h_{ij}(\tau)}{N_0 B} \right), \tag{4}$$

where B is the bandwidth, $P_{ij}(\tau)$ is the transmission power, h_{ij} is the channel gain between node i and node j at time τ and N_0 is the noise power spectral density.

3) Vehicle Mobility Modeling: Using stochastic differential equations, the vehicle trajectories can be predicted. Let X_τ be the position of a vehicle at time t:

$$dX_\tau = \mu(X_\tau, \tau) d\tau + \sigma_{ij}(X_\tau, \tau) dW_\tau, \tag{5}$$

where $\mu(X_\tau, \tau)$ is the drift term (deterministic trend, e.g., road layout), $\sigma_{ij}(X_\tau, \tau)$ is the diffusion coefficient (randomness due to traffic fluctuations), and dW_τ represents increments of a Wiener process.

B. Energy-Aware Semantic Caching Model

Each vehicle v_i and edge node e_{ij} is equipped with a semantic-aware cache, each with a finite storage capacity. The semantic similarity between two contents c_k and $c_{k'}$ is defined as:

$$S(c_k, c_{k'}) = \frac{\sum_{i=1}^{d} v_{i,k} v_{i,k'}}{\|\mathbf{v}_k\| \|\mathbf{v}_{k'}\|}, \tag{6}$$

where $\mathbf{v}_k$ represents semantic vector of content. Content c_k caches if its similarity with cached content $c_{k'}$ is below the predefined threshold θ, as mentioned in $S(c_k, c_{k'}) < \theta$, as the predefined threshold θ ensures content diversity. The binary caching decision variable $x_{i,k}(\tau)$ at time step τ is expressed as:

$$x_{i,k}(\tau) = \begin{cases} 1, & \text{if content } d_k \text{ is cached at node } i, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Globally, all the energy cost are measured in Joules (J). The energy cost for caching is:

$$E_i^{\text{cache}}(\tau) = x_{i,k}(\tau) \left(P_{\text{cache}} + \gamma S_k \right), \tag{8}$$

where P_{cache} represents the fixed caching power, S_k is the content size, and γ is a scaling factor.

C. Energy-Aware Routing Model with Constraints

Each vehicle must optimize its route $\mathcal{R}_i(\tau)$ while minimizing latency and energy consumption. The routing decision variable $y_{i,j}(\tau)$ at time step τ is given by:

$$y_{i,j}(\tau) = \begin{cases} 1, & \text{if vehicle } i \text{ transmits through edge node } j, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

The latency cost $L_i^{\text{route}}(\tau)$ at time step τ is:

$$L_i^{\text{route}}(\tau) = \sum_{(i,j) \in \mathcal{L}(\tau)} y_{i,j}(\tau) \left(\frac{D_k}{B_{i,j}(\tau)} + D_{\text{proc},j} \right), \tag{10}$$

where $\frac{D_k}{B_{i,j}(\tau)}$ represents time taken to transmit data of size D_k over transmission link with available bandwidth $B_{i,j}(\tau)$ and $D_{\text{proc},j}$ accounts for the processing delays.

D. Energy-Aware Network Model

Each node i has a real-time energy budget $E_i(\tau)$. The energy consumption is partitioned into three main components:

1) Communication Energy: For a transmission over link (i, j):

$$E_{\text{comm}}^{ij}(\tau) = \kappa_{ij} \, d_{ij}(\tau), \tag{11}$$

where $d_{ij}(\tau)$ is the distance between nodes and κ_{ij} is the energy cost per unit distance.

2) Computation Energy: For a computational load $C_i(\tau)$ (e.g., data processing, model inference):

$$E_{\text{comp}}^{i}(\tau) = \eta_i \, C_i(\tau), \tag{12}$$

with η_i as the energy cost per computation unit.

3) Caching Energy: For storing data of size s_d at node i:

$$E_{\text{cache}}^{i}(\tau) = \lambda_i \sum_{d \in \mathcal{D}} s_d \, y_{id}(\tau), \tag{13}$$

where $y_{id}(\tau) \in \{0, 1\}$ is the caching decision (1 if data d is cached at node i at time t), and λ_i is the energy per unit storage cost.

E. Combinatorial Caching Problem Formulation

We aim to minimize the total energy consumption throughout the network while ensuring low latency content delivery. Let the total energy consumption $\Omega_{(i,j)}(\tau)$ for the nodes i and

j to facilitate the communication and computation of caching and routing at time step τ as:

$$\Omega_{(i,j)}(\tau)=\sum_{i\in\mathcal{V}}\left(E_i^{\text{total}}(\tau)+\beta L_i^{\text{route}}(\tau)\right) \tag{14}$$

Given $\mathbf{X}=\{x_{i,k}(\tau)\}$ is the binary caching decision for content k at vehicle i, $\mathbf{Y}=\{y_{i,j}(\tau)\}$ is the routing decision variable indicating if vehicle i transmits via edge node j), $\mathbf{P}=\{P_{ij}(\tau)\}$ is the power allocation for transmission between nodes, $E_i^{\text{cache}}(\tau)$, $E_i^{\text{comm}}(\tau)$, and $E_i^{\text{comp}}(\tau)$ are the energy costs for caching, communication, and computation, respectively, $L_i^{\text{route}}(\tau)$ is the latency cost, and β is a tradeoff factor controlling the balance between energy and latency. The optimization problem can be formulated as follows.

$$\textbf{P1:}\quad \min_{\mathbf{X},\mathbf{Y},\mathbf{P}}\sum_{i\in\mathcal{V}}\left(\Omega_{(i,j)}(\tau)\right) \tag{15a}$$

$$\text{s.t.}:\quad \sum_{k\in\mathcal{C}}x_{i,k}(\tau)S_k\le C_i^{\max},\quad \forall i\in\mathcal{V}\cup\mathcal{R} \tag{15b}$$

$$E_i^{\text{total}}(\tau)\le E_i^{\max},\quad \forall i\in\mathcal{V} \tag{15c}$$

$$0\le P_{ij}(\tau)\le P_i^{\max},\quad \forall (i,j)\in\mathcal{E}_t \tag{15d}$$

$$D_k\le C_{ij}(\tau)T_s,\quad \forall (i,j)\in\mathcal{E}_t,\forall k\in\mathcal{C} \tag{15e}$$

$$\sum_{j\in\mathcal{N}_i}y_{i,j}(\tau)-\sum_{j\in\mathcal{N}_i}y_{j,i}(\tau)=d_i^t,\quad \forall i\in\mathcal{V} \tag{15f}$$

$$x_{i,k}(\tau)+\sum_{j\in\mathcal{N}_i}y_{i,j}(\tau)\ge z_{i,k}(\tau),\quad \forall k\in\mathcal{C},\forall i\in\mathcal{V} \tag{15g}$$

$$\pi_{ij}(\tau)\ge\theta,\quad \forall (i,j)\in\mathcal{E}_t \tag{15h}$$

Herein, (15b) ensures that cached content size doesn't exceed the vehicle or RSU storage capacity, (15c) limits total energy for caching, communication, and computation, (15d) requires nonnegative communication power within device limits, (15e) ensures feasible data transmission under changing channel conditions. (15f) maintains data flow consistency by aligning packet transmission with content request flow, (15g) handles binary caching and routing decisions, while (15h) uses Markovian probabilistic link modeling.

III. Proposed Q-SAC Algorithm Design

In real-time, (15a) becomes NP-hard and impractical with increasing IoV devices, complicating offloading in high-dimensional spaces. We present the Q-SAC algorithm leveraging quantum variational circuits (QVCs) and regularized RL for quicker task processing. The method uses quantum policy optimization and hybrid quantum-classical critics to efficiently handle network dynamics, allowing for scalable and energy-efficient decisions in large IoV environments. The next subsection explains the Q-SAC framework and its real-time applications.

A. Markov Decision Process (MDP) Reformulation

We reformulated (15a) as an MDP represented by a tuple $(\mathcal{S},\mathcal{A},\mathcal{P},r,\rho)$: $\mathcal{S}$ is the state set, $\mathcal{A}$ is the continuous action space, $\mathcal{P}(s'|s,a)$ is the transition probability, with $\mathcal{P}\left(s_{\tau+1}\mid s_\tau,a_\tau\right)$ as the probability of moving from state s_τ to $s_{\tau+1}$ after action a_τ at time τ. $r(s,a)$ is the reward function, and $\rho\in[0,1)$ is the discount factor. The state, action, and reward functions are defined as follows.

1) State space: captures all the necessary information that describes the current network conditions and resource status. The state vector can be defined as:

$$s_\tau=\Big(E_i^{\text{total}}(\tau),x_{i,k}(\tau),C_{ij}(\tau),d_i(\tau),L_i^{\text{route}}(\tau)\Big). \tag{16}$$

2) Action space: at each time step τ, the agent takes an action $a_\tau\in\mathcal{A}$ that consists of making decisions. Thus, the overall action vector consists of decision variables given as:

$$a_\tau=\Big(\{x_{i,k}(\tau),y_{i,j}(\tau),\{P_{ij}(\tau)\}_{(i,j)\in\mathcal{E}_\tau}\Big). \tag{17}$$

3) Reward function: provides feedback where the agent (via Q-SAC) will learn to select actions that minimize these costs in expectation. The expected reward function is given by:

$$r(s_\tau,a_\tau))=-\sum_{i\in\mathcal{V}}\left(E_i^{\text{total}}(\tau)+\beta L_i^{\text{route}}(\tau)\right)+\sum_{i,j}\lambda\mathcal{Q}_e \tag{18}$$

where $\mathcal{Q}_e$ represents the penalties for constraints violations and λ is the large penalty factors. If constraints are violated, we impose heavy penalties.

B. Quantum Soft Actor for Scalable Energy-Aware Caching and Routing in Smart-City IoV Fleets

1) Quantum State and Action Representation for IoV: The state of vehicle is encoded as $\psi_s=U_\phi(s)0^{\otimes n}$, where $U_\phi(s)$ maps classical state features (location, energy levels, cache contents, connectivity) to quantum states and $0^{\otimes n}$ is the initial zero state. We use higher-order encoding to transform the classical space into a multi-qubit quantum Hilbert space. The routing and caching actions are encoded using:

$$\psi_s=\sum_{i=0}^{2^n-1}z_i(s)i, \tag{19}$$

where i represents the computational basis states $0,1,...,2^n-1$ and $z_i(s)$ represents normalized amplitudes for cache operations (store, update, remove) and routing decisions (vehicle-to-vehicle, vehicle-to-edge).

2) Soft Policy Optimization with Entropy Regularization: Q-SAC boosts efficiency in large-scale smart-city IoV by optimizing the soft policy objective, defined as:

$$J_\pi(\theta)=\mathbb{E}_{s\sim\rho_\pi,a\sim\pi_\theta}\left[Q_\phi(s,a)-\varphi\log\pi_\theta(a|s)\right] \tag{20}$$

where $Q_\phi(s,a)$ is the soft Q-value function evaluating energy-aware caching and routing efficiency, φ is the entropy regularization coefficient promoting route diversity and caching adaptability, and $-\varphi\log\pi_\theta(a|s)$ promotes exploration to minimize congestion and enhance energy optimization.

3) Quantum Gradient Estimation for Fleet Policy Optimization: The quantum gradient of the policy loss function for the routing and caching policy is given by:

$$\nabla_\theta J_\pi(\theta) = \mathbb{E}_{s,a}\left[\nabla_\theta \log \pi_\theta(a|s)(Q_\phi(s,a) - \varphi \log \pi_\theta(a|s))\right], \tag{21}$$

where $\nabla_\theta \log \pi_\theta(a|s)$ represents quantum policy gradient methods for variational quantum circuits (VQCs).

C. Quantum Soft Critic for Scalable Smart-City IoV Fleets

In Q-SAC, the quantum soft critic estimates the soft Q-value function $Q_\phi(s,a)$ to evaluate routing and caching efficiency based on energy savings and data retrieval latency. The soft Q-value function is estimated using a quantum variational circuit as:

$$Q_\phi(s,a) = \psi_{s,a,\phi} H_Q \psi_{s,a,\phi}, \tag{22}$$

where $\psi_{s,a,\phi} = U_\phi(\psi_s \otimes \psi_a)$ represents the quantum state of the system and H_Q is a Hermitian observable corresponding to Q-value estimation. By measuring H_Q in the quantum state $\psi_{s,a,\phi}$, we obtain an estimate of $Q_\phi(s,a)$, ensuring efficient routing and caching decision-making. The soft Q-value function follows the soft Bellman equation given by:

$$Q_\phi(s,a) = r(s,a) + \rho \mathbb{E}_{s' \sim P}\left[V_\phi(s')\right], \tag{23}$$

where the soft state value function of IoV fleet is:

$$V_\phi(s) = \mathbb{E}_{a \sim \pi}\left[Q_\phi(s,a) - \varphi \log \pi_\theta(a|s)\right]. \tag{24}$$

This soft Bellman update is performed using quantum Gibbs sampling to estimate future routing and caching benefits.

1) Quantum Temporal Difference Learning for Smart-City Optimization: The quantum soft critic is trained by minimizing the soft Bellman loss is given by:

$$\mathcal{L}_Q(\phi) = \mathbb{E}_{s,a,r,s'}\left[(Q_\phi(s,a) - \beta)^2\right], \tag{25}$$

where $\beta = r(s,a) + \rho\left[V_\phi(s')\right]$ denotes the energy-aware target value. This enables optimal routing and caching strategies for scalable, energy-efficient IoV fleets in smart-city environments.

2) Quantum Gradient Update for Q-Value Estimation: The critic network parameters ϕ are updated using:

$$\nabla_\phi \mathcal{L}_Q(\phi) = \mathbb{E}_{s,a,r,s'}\left[\nabla_\phi Q_\phi(s,a)(Q_\phi(s,a) - y)\right]. \tag{26}$$

We apply quantum gradient descent to update the parameters as:

$$\theta \leftarrow \theta - \Theta \nabla_\theta \mathcal{L}(\theta), \tag{27}$$

where Θ is the learning rate and is $\mathcal{L}(\theta)$ the cost function. This quantum-enhanced update improves policy evaluation for large-scale smart-city IoV Fleets as illustrated in **Algorithm 1**. Moreover, the computational complexity of the actor network, utilizing a quantum feature map and variational ansatz, is $\mathcal{O}(r \cdot n_q^2)$, influenced by n_q qubits and r ansatz depth. The critic network, a hybrid model with a classical neural network and quantum Q-value estimation, has a forward pass complexity of $\mathcal{O}(n_s + n_a + r \cdot n_q^2)$, based on state dimension n_s and action dimension n_a. The per-step training complexity is $\mathcal{O}(n_s \cdot n_q + n_a + r \cdot n_q^2)$, illustrating scalability and polynomial efficiency over classical methods. The hybrid quantum-classical approach optimizes large-scale IoV environments, making Q-SAC ideal for real-time smart-city routing and caching decisions.

Algorithm 1: Quantum Soft Actor-Critic (Q-SAC)

Initialize: π_θ, Q_ϕ, $\mathcal{D}$, ρ, φ, M, K
for each iteration **do**
  for each environment step **do**
    Observe state s_t and encode as quantum state $\psi_{s_t} = U_\phi(s_t)0^{\otimes n}$
    Sample action $a_t \sim \pi_\theta(a_t \mid s_t)$
    Execute a_t, observe reward r_t and next state s_{t+1}
    Store (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}$
  end for
  for K epochs **do**
    Sample minibatch of size M from $\mathcal{D}$
    Compute soft Q-loss using equation (23):
    Compute soft value function via equation (24):
    Compute Q-target: $y_t = r_t + \rho V_\phi(s_{t+1})$
    Update critic parameters: $\phi \leftarrow \phi - \Theta \nabla_\phi \mathcal{L}^Q$
    Compute policy loss through equation (26):
    Update policy parameters: $\theta \leftarrow \theta - \Theta \nabla_\theta \mathcal{L}^\pi$
  end for
end for

IV. Experimental Results and Discussions

We evaluate Q-SAC against classical SAC under identical settings using IBM Qiskit. Vehicles are uniformly positioned at RSU locations for users to process tasks locally or offload them to the quantum server. Path loss and channel fading in vehicular wireless environments are provided in [6], [10]. Models use a batch size of 4 with the Adam optimizer; actor and critic learning rates are 5×10^{-6} and 2×2^{-2}. VQC has 1 layer, buffer size is 1,000,000, and episode steps are 20. Noise power density is -70 dBm, maximum latency is 40 ms, and system bandwidth is 20 MHz.

A. Results Discussion

Fig. 1 shows energy consumption trends of classical SAC and proposed Q-SAC as IoV users increase. Q-SAC uses quantum-enhanced techniques to considerably achieve lower energy consumption than classical SAC. Notably, the performance gap between the two methods widens with the growth in IoV user count, highlighting the superior scalability and efficiency of the proposed Q-SAC approach in large-scale, dynamic network environments.

Fig. 2 shows how semantic cache size (GB) affects system latency and cache hit ratio in a smart-city IoV network using the Q-SAC framework. Larger caches reduce latency, boost data retrieval efficiency, and lessen congestion. They also raise the cache hit ratio by serving more data locally

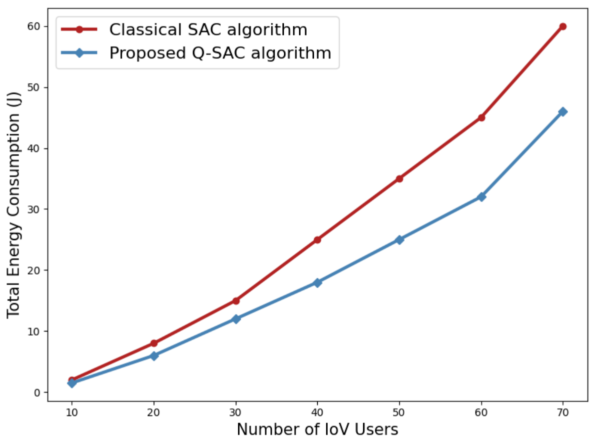

Fig. 1: Comparison of the total energy consumption

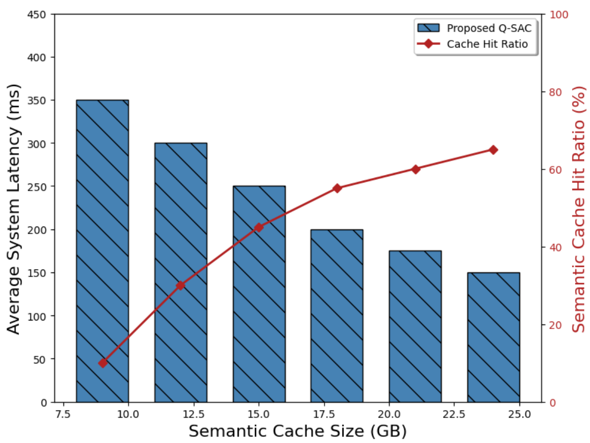

Fig. 2: Average latency performance

and cutting transmission delays. This balance underscores Q-SAC's capability in optimizing storage, achieving high cache hit ratios, and ensuring scalability for real-time IoV fleet management.

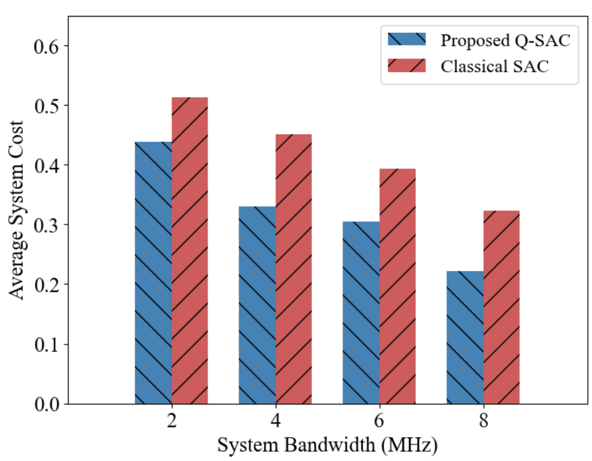

Fig. 3: Average system cost performance

Fig. 3 shows the average system cost versus system bandwidth for Q-SAC and classical SAC in a smart-city IoV. With rising bandwidth, both methods reduce system costs, enhancing resource use and communication. Yet, Q-SAC consistently achieves lower costs, proving its superiority in optimizing decisions under uncertain and dynamic networks.

V. Concluding Remarks

We introduce a Q-SAC framework for energy-aware semantic caching and routing in large-scale smart-city IoV environments, bridging quantum variational RL with semantic communication. Beyond demonstrating superior energy efficiency and faster convergence, we highlight the feasibility of integrating quantum learning into low-latency vehicular networks.

Acknowledgments

The work of T. Q. Duong was supported in part by the Canada Excellence Research Chair (CERC) Program CERC-2022-00109 and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant Program RGPIN-2025-04941.

References

[1] Y. Zhao, C. Liu, X. Hu, J. He, M. Peng, D. Wing Kwan Ng, and T. Q. S. Quek, "Joint content caching, service placement, and task offloading in uav-enabled mobile edge computing networks," *IEEE Journal on Selected Areas in Communications*, vol. 43, no. 1, pp. 51–63, 2025.

[2] X. Zhou, M. Bilal, R. Dou, J. J. P. C. Rodrigues, Q. Zhao, J. Dai, and X. Xu, "Edge computation offloading with content caching in 6g-enabled iov," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 3, pp. 2733–2747, 2024.

[3] L. Liu, X. Yuan, N. Zhang, D. Chen, K. Yu, and A. Taherkordi, "Joint computation offloading and data caching in multi-access edge computing enabled internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 72, no. 11, pp. 14 939–14 954, Jun. 2023.

[4] H. Tang, A. Wang, F. Xue, J. Yang, and Y. Cao, "A novel hierarchical soft actor-critic algorithm for multi-logistics robots task allocation," *IEEE Access*, vol. 9, pp. 42 568–42 582, Feb. 2021.

[5] J. Yang, J. Zhang, M. Xi, Y. Lei, and Y. Sun, "A deep reinforcement learning algorithm suitable for autonomous vehicles: Double bootstrapped soft-actor–critic-discrete," *IEEE Trans. Cogn. Develop. Syst.*, vol. 15, no. 4, pp. 2041–2052, Jun. 2021.

[6] D. Wang, B. Song, P. Lin, F. R. Yu, X. Du, and M. Guizani, "Resource management for edge intelligence (EI)-assisted IoV using quantum-inspired reinforcement learning," *IEEE Internet of Things J.*, vol. 9, no. 14, pp. 12 588–12 600, Jul. 2022.

[7] J. A. Ansere, G. Han, and H. Wang, "A novel reliable adaptive beacon time synchronization algorithm for large-scale vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11 565–11 576, Dec. 2019.

[8] C. Ling, W. Zhang, Q. Fan, Z. Feng, J. Wang, R. Yadav, and D. Wang, "Cooperative service caching in vehicular edge computing networks based on transportation correlation analysis," *IEEE Internet of Things J.*, vol. 11, no. 12, pp. 22 754–22 767, 2024.

[9] B. Bandyopadhyay, P. Kuila, M. Bey, and M. C. Govil, "Quantum-inspired differential evolution for freshness-aware caching-aided offloading in digital twin-enabled internet of vehicles," *IEEE Trans. Intell. Veh.*, pp. 1–15, 2024.

[10] J. Adu Ansere, D. T. Tran, O. A. Dobre, H. Shin, G. K. Karagiannidis, and T. Q. Duong, "Energy-efficient optimization for mobile edge computing with quantum machine learning," *IEEE Wireless Commun. Lett.*, vol. 13, no. 3, pp. 661–665, Mar. 2024.

[11] T. Q. Duong, L. D. Nguyen, B. Narottama, J. A. Ansere, D. V. Huynh, and H. Shin, "Quantum-inspired real-time optimization for 6G networks: Opportunities, challenges, and the road ahead," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 1347–1359, 2022.