Q^2oE: Balancing QoE Fairness and Preference for Video Streaming from Cooperative Edge Servers

Xinwei Huang[1,3] Yin Xu[2,3] Jinbo Cai[2,3] He Sun[2,3] Jie Wu[4] Mingjun Xiao[2,3]
[1]School of Software Engineering, University of Science and Technology of China, China
[2]School of Computer Science and Technology, University of Science and Technology of China, China
[3]Suzhou Institute for Advanced Study, University of Science and Technology of China, China
[4]Department of Computer and Information Sciences, Temple University, USA
Corresponding Email: xiaomj@ustc.edu.cn

***Abstract*—Quality of Experience (QoE) is a pivotal factor that defines the overall quality within the realm of video streaming services. To improve users' QoE indicators, existing methods have proposed many QoE models and Adaptive Bitrate (ABR) algorithms. However, these advanced methods do not account for the tradeoff between users' personalized preferences and fairness, leading to resource wastage on the edge server and an inefficient bitrate allocation. Given resource constraints, non-cooperative edge servers frequently request the cloud to enhance user QoE, consequently resulting in high-latency transmissions. In this paper, we address this issue by introducing collaborative edge servers, which alleviate the burden of video chunk transcoding and caching on individual servers. Specifically, we design a novel QoE model named Q^2oE that incorporates user-level QoE preference and QoE fairness in this scenario. Furthermore, we propose Q2D3, a Q^2oE-aware reinforcement learning algorithm based on a video chunk transcoding strategy for mobility-aware vehicular networks, to improve Q^2oE model performance.**

***Index Terms*—Video streaming, quality of experience, deep reinforcement learning, vehicular network.**

I. Introduction

A. Background

With the rapid expansion of video media, video streaming traffic has increased dramatically [1]. Various video conferencing platforms (such as Tencent Meeting and Zoom) are ubiquitous in daily work, and 360-degree videos have steadily emerged as a prevailing trend [2], [3]. In this context, users have many diverse requirements for video quality, therefore Quality of Experience (QoE) is a crucial research area in the field [4]–[6].

High-quality user experience typically features high video bitrate, minimal transmission delay, low rebuffering time, and low frequency of bitrate variation [7]. Therefore, in real-time video streaming, caching, and transcoding video chunks at the edge servers are particularly important [8], [9]. Nevertheless, the limited caching and transcoding capabilities inherent in an individual edge server (*e.g.*, the millimeter-wave base station) result in frequent requests to the cloud for assistance, causing significant transmission delays. However, collaborative edge servers can effectively address and optimize for this limitation [10]. The advantages of collaborative edge server design include: *(1) Low latency:* Latency is lower compared to methods directly requesting video chunks from the cloud. *(2) Alleviated cloud load:* If all users send requests to the cloud, the pressure on resource retrieval and compute workloads increases dramatically. Through a combination of these advantages, collaborative edge servers effectively enhance QoE.

Due to the mobility characteristics of the user, especially the vehicles that move at high speeds, the resource utilization of edge servers needs to be rationalized. For instance, when a high number of vehicle users access the same edge server, achieving high QoE for all users becomes challenging due to the inherent limitations of the edge server resources [6]. In certain scenarios, users who specifically require only medium-level bitrate video chunks may be assigned high-level bitrate video chunks, resulting in suboptimal resource allocation and wastage. This complication further emphasizes the research challenge related to user-level QoE preference and fairness.

1) User-level QoE preference: As a basic observation, users exhibit diverse preferences regarding various QoE metrics [11], [12]. For example, certain vehicle users may tolerate delays caused by rebuffering or transmission issues and opt for high-bitrate video chunks, while others might prioritize low latency and choose medium-bitrate or low-bitrate video chunks. In this paper, we refer to users' tolerance for different QoE metrics, such as video bitrate level and latency, as user-level QoE preferences.

2) QoE fairness: All vehicle users within the current server must be allocated bandwidth for requesting video chunks, and video chunks of different bitrate levels require varying bandwidths [6]. However, the server's bandwidth is constrained. For example, a new vehicle user enters the coverage area of the server and requests high-bitrate video chunks, but the server's unallocated bandwidth can only meet the transmission of low-bitrate video chunks. This will degrade user QoE and lead to fairness issues. In this paper, we quantify fairness by using the lowest QoE in user-acquired video chunks.

B. Motivations

User-level QoE preferences are crucial for designing effective QoE models and Adaptive Bitrate (ABR) algorithms. However, the design philosophy behind these QoE models targets the "average user", thereby neglecting the individuality of each user [4], [5], [13]. And, traditional ABR algorithms are also agnostic to user preferences, because they usually optimize towards a fixed QoE model [14], [15]. However, each user's perceptual preferences are different [11]. Therefore, the differences between user preferences cannot be ignored, and the "average user" cannot represent all users. Environmental factors also need to be considered, such as watching videos at home, watching videos while walking, or watching videos while driving. User QoE perception is different in these scenarios. Especially when the vehicle is traveling at high speed, increasing the user video bitrate will not significantly improve QoE. Blindly increasing video bitrate not only fails to improve QoE but may even lead to a waste of edge server resources, resulting in bitrate inefficiency [16].

QoE fairness is of paramount importance in the context of vehicle networks [17], [18]. As new vehicles continually enter the coverage area of edge servers, issues related to server bandwidth preemption can arise easily. Most existing methods use average bandwidth allocation to provide video services to ensure the fairness of edge server resources. However, since users exhibit diverse QoE preferences, the practice of evenly allocating bandwidth places excessive emphasis on fairness, potentially at the expense of individual user characteristics. Therefore, bandwidth allocation is an important part of ensuring QoE fairness. To better accommodate mobile scenarios, Yuan *et al* [6] explored an approach to bandwidth allocation for users in motion. This method takes into account factors such as user speed and direction to tailor bandwidth allocation, recognizing the dynamic nature of mobile environments. While this bandwidth allocation method effectively resolves the bandwidth preemption challenges posed by mobile scenarios, enhancing the overall QoE indicators for all users, it does have limitations. Notably, this allocation method does not consider individual user QoE preferences. Consequently, users with high preferences may not be fully satisfied, while those with lower preferences may encounter issues related to low-efficiency bitrate allocation. So, this method can result in inefficient utilization of some server resources.

C. Key Contributions

Considering the aforementioned factors, we propose a novel QoE model that balances preferences and fairness named the Q^2oE model. Based on this model, we design a corresponding transcoding algorithm with reinforcement learning deployed on collaborative edge servers. The main contributions of this paper are summarized as follows:

- We propose the Q^2oE model, a novel QoE model that combines user QoE preferences and fairness. To the best of our knowledge, we are the first to address user-level QoE preferences and QoE fairness issues in vehicle networks.
- We introduce a collaborative edge system where edge servers can collaborate with neighboring servers for caching and transcoding. This characteristic greatly improves the efficiency of user requests for video chunks, and significantly reduces latency compared to traditional edge servers.

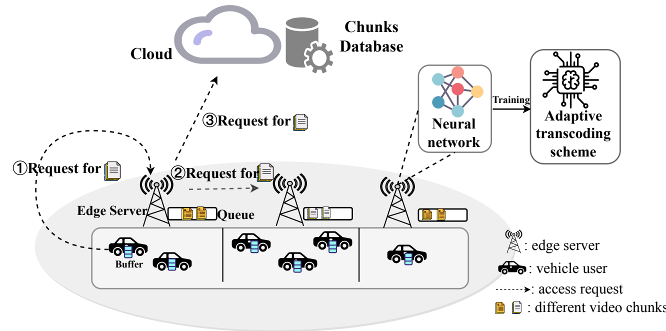


Fig. 1: Q^2oE-aware system in vehicular networks.

- We present Q2D3, a reinforcement learning algorithm based on vehicle networks, to implement the Q^2oE model. When edge server resources are limited, this algorithm can maximize satisfying users' QoE preferences while improving fairness as much as possible. Compared to other state-of-the-art algorithms, our algorithm satisfied an average of 103% of user preferences and improved fairness by 13% on average.

II. Related Works

Building upon previous studies on transcoding and caching of traditional video streams, extensive research has now emerged on video streaming issues in mobile environments [6], [16], [19]–[22]. Yuan *et al.* [6] proposed the VSiM system, which aims to address the issue of QoE fairness in mobile environments. This system integrates client mobility profiles and QoE-related information to enable dynamic and fair bandwidth allocation. However, this study concentrates solely on enhancing the fairness of QoE and overlooks users' individualized preferences. In vehicular networks, the work [19] proposed an infrastructure-assisted millimeter wave vehicular network, which utilizes mmWave base stations (mBSs) as assisting infrastructures to transmit video chunks. It utilizes the DDPG (Deep Deterministic Policy Gradients) algorithm to execute the video chunk push strategy. Nonetheless, this system lacks the integration of collaborative edge nodes, resulting in excessive transmission pressure on the MBS. Furthermore, it does not address user-level preference and fairness issues.

When investigating ABR algorithms, Zuo *et al.* [11] discovered that many existing algorithms are designed for an abstract "average user" without addressing individual users' specific QoE preferences. Therefore, they proposed a user preference-based QoE model and incorporated it into an ABR algorithm to validate the accuracy of modeling user preferences. Mehrabi *et al.* [23] proposed the GreedyMSMC algorithm which combines user QoE and fairness in multi-user and multi-server scenarios. GreedyMSMC utilizes proportional fairness to allocate base station resources, thus achieving resource allocation across multiple users and servers.

For edge collaborative resource allocation, Yang *et al.* [24] proposed an edge server collaborative ABR algorithm where the local edge server connected to the user can collaborate with neighboring edge servers. The servers jointly perform caching and video transcoding.

III. PREFERENCE, FAIRNESS AND Q^{2}OE MODELS

A. System Overview

The system is primarily composed of three components: vehicle users are responsible for requesting and receiving video chunks, edge servers with caching and transcoding capabilities, and a cloud server holding all video chunk resources. Fig. 1 illustrates the entire workflow of the system, depicting how vehicle users request and receive video chunks. (1) Users first establish a connection with their affiliated edge server and request the required video chunks. (2) After receiving the user request, the edge server retrieves its cache area and returns it to the user if there is a corresponding video chunk. (3) If the current edge server lacks the requested video chunk, it seeks assistance from neighboring servers. The neighboring server checks its cache, returning the video chunk if a cache hit occurs. (4) If neither the current edge server nor the neighboring servers have the user's requested video chunk, the current server initiates a request to the cloud. The cloud server retrieves the needed video chunk and completes the transmission. (5) The cloud server proactively pushes video chunks to each edge server based on user location and viewed video chunks, aiding the edge servers in pre-caching.

In our system, there are a total of J edge servers, denoted as $\mathcal{J}$={0, 1 , 2, ..., J}, where j = 0 specifically represents the center cloud. Each edge server is equipped with a neural network capable of independently performing transcoding operations on video chunks, as illustrated in Fig. 1. The computational capacity, caching capacity, and bandwidth of each edge server are limited, represented as P_j, W_j, and B_j, respectively. N represents the number of users in all edge servers, and the sets of all video chunks and video chunk bitrate levels are denoted as $\mathcal{K}$={1, 2, ..., K} and $\mathcal{Q}$={1, 2, ..., Q}, respectively. V_k represents the bitrate level of the k^{th} video chunk, where $V_k \in Q$.

B. User-level Preference Model

According to the work [16], under different environments, the user's perceptual ability is different. For the same bitrate video stream, the user's perceptual ability is the strongest when static, and QoE is the highest. However, when the vehicle is moving at high speed, the user's perceptual ability is easily interfered, and QoE is slightly lower. Further increasing the video bitrate will not significantly improve the user's QoE, resulting in bitrate-inefficiency.

Therefore, in the vehicle network, we need to consider the characteristics of high-speed movement of the vehicle, and on this basis, define the user QoE preference suitable for the mobile environment. Due to the difference between static and dynamic user QoE perception, the user's initial QoE preference (static case) must be adjusted. According to Eq. (1), we use p_i to represent the user i 's initial QoE preference, and define P_i to indicate the user i 's true QoE preference when the vehicle is moving. And, γ is a hyperparameter. Then, we obtain the user's QoE preference in the vehicle network.

$$P_i = p_i - \gamma * v_i \tag{1}$$

C. QoE Model

In the scenario of high-speed vehicle movement, the QoE perceived by vehicle users is mainly affected by three factors: (1) quality variations between continuous video chunks, (2) the access latency of edge servers, including transmission latency and transcoding latency, we only consider transmission latency in this work, and (3) vehicle driving status. The QoE model is defined as follows:

$$QoE = Q_0 - \omega * I_c - \gamma * v - \delta * d \tag{2}$$

Q_0 is modeled with a Michaelis-Menten function, *i.e.*, $Q_0 = \max\left(1, \min\left(5, 1 + 4 \cdot \frac{c_1 \cdot \mathcal{V}_k}{c_2 + \mathcal{V}_k}\right)\right)$, where V_k is the bitrate level of the k^{th} video chunk, and c_1 and c_2 are the model parameters. We use the parameters from [10], *i.e.*, $c_1 = 1.036$ and $c_2 = 0.429$. These values are set based on subjective quality assessment experiments. The parameter ω penalises the gain in QoE with I_c for loss of smoothness while γ penalises the QoE gain with vehicle mobility status and δ penalises gain with access latency.

I_c represents fluctuations in the quality of continuous video chunks watched by the vehicle user. Due to the fluctuation of video block quality, users are likely to feel dizzy, which can significantly reduce user experience. Then, the QoE model should consider quality variations, as shown in Eq. (3) where $I_{c,k}$ represents the bitrate fluctuation when the user watches the k^{th} video chunk, and $Q_{0,k}$ represents the Q_0 value of the k^{th} video chunk.

$$I_{c,k} = \frac{max(V_{k-1} - V_k, 0)}{V_k} * Q_{0,k} \tag{3}$$

In Eq. (3), the difference between the bitrate of the $(k-1)^{th}$ video chunk and the k^{th} video chunk represents the oscillation of the quality of two consecutive video chunks. Here, only the downward fluctuation in video block quality will have negative effects, while an increase in bitrate often leads to positive effects.

D. Fairness Model

We transform the problem of improving the QoE fairness of the entire system into the problem of improving the QoE fairness of each edge server. Therefore, to improve the fairness of users in each edge server range with limited bandwidth B_j, we define user QoE fairness in Eq. (4) where M denotes that the video watched by each user i consists of M blocks. This is a standard QoE fairness metric. $QoE_{i,t}$ denotes the QoE actually felt by the user i in the timeslot t. In Eq. (4), c_t denotes the minimum value of the fairness index at timeslot t. Max-min QoE fairness reflects the QoE improvement of the worst performing clients, which helps service providers to offer a fairer service for clients [6], [25].

$$c_t = \min_{i \in N}(QoE_{i,t}/M) \tag{4}$$

E. Mobility Model

According to the work [6], information such as the moving direction, location, and speed of a vehicle can be captured by a user's mobile device. We use ρ to represent the probability that the vehicle user will move to the next edge server, and $(1-\rho)$ to indicate the probability that the vehicle user does not move out of the current edge server. Eq. (5) gives the transition probability as:

$$\rho = \frac{1000 * v'}{3600 * R} \tag{5}$$

where v' represents the vehicle's speed (*e.g.*, 80km/h) and R represents the range of the edge server (*e.g.*, 200m). For example, if a user's vehicle speed is 100km/h and the coverage range of each edge server is 200m, then $\rho = 0.139$.

F. Q^2oE Model

To combine considerations of user-level preference and QoE fairness, we must formulate an optimization object to find the trade-off point. As described above, we aim to satisfy each user's self-preferences and improve their QoE fairness. However, there is an inhibitory relationship between them. So we can formally define our optimization problem using the weighted sum method as below. In Eq. (6), α and β are two hyperparameters, and $\alpha + \beta = 1$. P_i is the expected QoE of each user i, and $QoE_{i,t}$ is the actual QoE at time t. We also define c_t as $\min_{i \in N}(QoE_{i,t}/M)$ at moment t and $F_t = c_t - c_{t-1}$. Therefore, F_t represents the improvement of the fairness index. With a smaller β , we tend to prioritize the preferences of certain users, while with a larger β, we prefer higher QoE fairness (*e.g.* when bandwidth is limited, priority is given to users with the lowest actual QoE). P_i minus $QoE_{i,t}$ represents the difference between the actual and the ideal, which we aim to minimize.

Finally, we derive the optimization object of our Q^2oE model with Eq. (6). Next we can incorporate the Q^2oE model with reinforcement learning algorithms to achieve optimization.

$$maxmize \sum_{t}^{T} (\alpha * \frac{1}{\sum_i^N |P_i - QoE_{i,t}|} + \beta * F_t) \tag{6}$$

Given the number of edge servers and the frequent handoffs of vehicles, edge servers require a significant amount of information to make transcoding decisions. This includes information such as the association between vehicles and edge servers, the caching status of edge servers, the current bandwidth allocation status, and user preferences. These uncertain factors complicate the optimization problem, making it difficult to solve with traditional optimization methods. Therefore, we employ reinforcement learning to optimize this issue.

IV. Q^2OE-AWARE DEEP REINFORCEMENT LEARNING

A. The proposed Q2D3 algorithm

1) State space: The state of each edge server S_t in the current environment consists of four components: (1) The location of the user vehicle $L(t)$, including which edge node's coverage area the user is currently driving within. (2) The cache status of the edge server $C(t)$, equivalent to the bitrate of the video chunks currently cached at the edge node. (3) The velocity of the user's vehicle $V(t)$. (4) The QoE preference of the user $P(t)$. It is important to note that all these states possess the Markov property. Hence, the state of the defined MDP can be represented as $S_t = [L(t), C(t), V(t), P(t)]$, where $L(t) \in \mathbb{L}$, $C(t) \in \mathbb{C}$, $V(t) \in \mathbb{V}$ and $P(t) \in \mathbb{P}$. Here, $\mathbb{L}$, $\mathbb{C}$, $\mathbb{V}$ and $\mathbb{P}$ denote the set of all possible user locations, cache states of edge servers, user mobility probabilities, and user preferences respectively.

2) Action space: Due to the characteristics of edge server collaboration, each edge server has two possible actions: (1) The transcoding decision $T(t)$ for the video chunks of in-coverage users at the current edge server. (2) The transcoding operation $T'(t)$ for the video chunks for the users that are about to enter the coverage range of the current edge server, which can be inferred based on the velocity and location information of users approaching the coverage area. Therefore, the action taken at time t can be defined as $A_t = [T(t), T'(t)]$, where $T(t) \in \mathbb{T}$ and $T'(t) \in \mathbb{T}$. Here, $\mathbb{T}$ represent all possible transcoding decisions.

3) Reward function: The formulation of the reward function is of critical importance for the efficacy of reinforcement learning algorithms. Based on the previously mentioned optimization goals of user-level QoE preference and fairness, we propose the reward equation in Eq. (7). We utilize the reciprocal of an exponential function to represent the proximity between the user's preference value and actual value. The constant C is used to control the range of the value. In our simulation experiments, C is set to 5.

$$r_{t,i} = \alpha * \frac{1}{\exp(P_i - QoE_{i,t})} * C + \beta * F_t \tag{7}$$

For edge server j, the reward for the actions it performs is the sum of all user rewards $r_{t,i}$ belonging to edge server j during time slot t. As shown in Eq. (8):

$$r_{t,j} = \sum_{i}^{N_j} r_{t,i} \tag{8}$$

where N_j represents the number of all users who have established connections to edge server j. In actuality, we evaluate the performance of our system using the total rewards r_t accumulated across all agents, which are the edge servers.

$$r_t = \sum_{j}^{J} r_{t,j} \tag{9}$$

4) The Detailed Algorithm: Algorithm 1 details our proposed Q2D3 algorithm to solve the user-level QoE preference and fairness problem. At each time slot t, agent j either randomly explores actions or directly selects the currently optimal action $a_{j,t}$ and then executes this action. Subsequently, according to Eqs. (7) and (8), we can derive the reward $r_{t,j}$ corresponding to the edge service j, and finally get the

Algorithm 1 algorithm of Q2D3

Initialize: Initialize each agent's evaluation network Q and target network Q' with random parameters; Initialize the replay buffer M;

for $episode = 1, 2, ..., E$ **do**
  Reset the environment and get an initial observation state s_0;
  for $t = 1, 2, 3, ..., T$ **do**
    for $j = 1, 2, 3, ..., J$ **do**
      Cache video chunks based on the user's location and speed information;
      Select actions $a_{j,t}$ based on current state and random exploration probability;
      Executes $a_{j,t}$, that is, executes corresponding transcoding actions;
    for $i = 1, 2, 3, ..., N$ **do**
      Move and request video chunks from local edge servers;
      Calculate the value of $QoE_{i,t}$ and the value of F_t, and then get the corresponding reward $r_{t,i}$ according to Eq. (7);
    The environment return the state of the next time slot s_{t+1} and reward r_t according to Eq. (9);
    Store sample (s_t, a_t, r_t, s_{t+1}) in M;
    Sample a random minbatch of K samples from M;
    for $j = 1, 2, 3...J$ **do**
      Update the evaluation network parameters θ with loss function by Eq. (10);
      Update the target network parameters θ' with loss function by Eq. (11);

total reward r_t of the system according to Eq. (9). After obtaining the reward r_t and the state s_{t+1} for the next time slot, the transition (s_t, a_t, r_t, s_{t+1}) is stored in the experience replay buffer M. Then, each edge server updates its network parameters based on K randomly selected transitions from the replay buffer. For each agent, the loss function for the evaluation network is shown in Eq. (10), and the parameter update for the target network uses Eq. (11) where θ and θ' are the parameters for the evaluation network Q and target network Q' respectively, with $a^* = argmaxQ(s_{t+1}, a; \theta)$, and τ = 0.5 in our simulation experiments.

$$L(\theta) = \frac{1}{K} \sum_{i=1}^{K} \left[r_t + \gamma Q' \left(s_{t+1}, a^*; \theta' \right) - Q \left(s_t, a_t; \theta \right) \right]^2 \tag{10}$$

$$\theta' = \tau\theta + (1 - \tau)\theta' \tag{11}$$

V. EVALUATION

A. Simulation Settings

In this section, we evaluate the proposed Q2D3 algorithm, which is based on collaborative edge servers and considers user QoE preferences and fairness under vehicular mobility scenarios. The system is simulated on a PC platform (Intel(R) Xeon(R) Silver 4310 CPU @ 2.10GHz, RTX 3080TI). In the system, five edge servers are configured and a random number of vehicles are generated, with the velocity of these vehicles ranging between 30km/h and 120km/h. The cloud has information about all video chunks. The end-to-end latency for users to fetch the same bitrate video chunks from the edge server, neighbor edge server, and the cloud are randomly assigned following a uniform distribution within the range of [5, 10] (ms), [20, 50] (ms), and [100, 200] (ms) [26], respectively. We assume that the roads are clear, that vehicles can move normally, and that there is no network congestion due to vehicle congestion.

The neural network used in our proposed Q2D3 algorithm consists of two fully connected hidden layers, with each fully connected layer containing 1024 neurons. We set the size of the experience replay buffer to 10000 and the batch size to 256. The discount factor γ in the reward function is set to 0.95, and the network update parameter τ is set to 0.005. To ensure that the algorithm operates in a valid action space, the input of the neural network is defined as the size of the state space, and the output is defined as the size of the action space, avoiding the generation of invalid actions. We have performed a total of 30,000 training iterations, and finally the average reward of the proposed algorithm progressively increases and converges at about the 3000th episode.

B. Performance Comparison

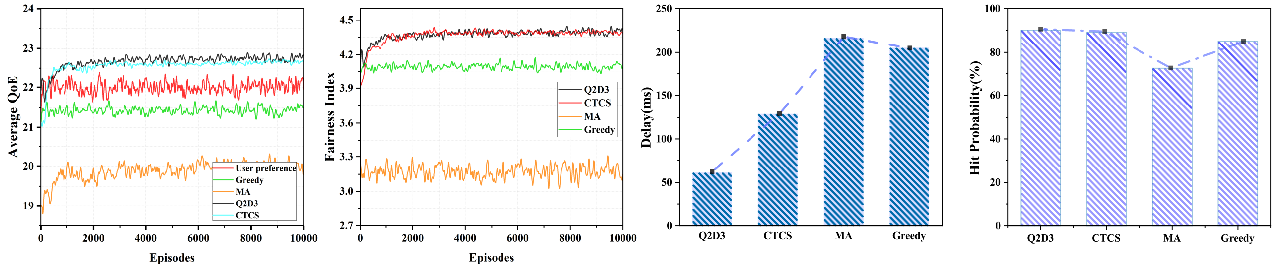
In this section, we evaluate the performance of Q2D3, comparing it with three alternative methods.

DRL-CTCS [24]: This approach consists of a cooperative transcoding and caching strategy at the edge cluster. However, it does not take into account user mobility, as users are static in this scenario with fixed edge server connections. The focus of this work remains on improving user QoE, without considering personalized user preferences and fairness. In contrast, our system comprehensively considers user preferences, mobility, and fairness.

MA [10], [19]: Mobile-based algorithm, which considers user mobility, but overlooks user preferences and fairness issues. Additionally, this work does not consider the potential for collaboration between edge servers. We incorporate this algorithm into our system and compare its performance with our proposed Q2D3 algorithm.

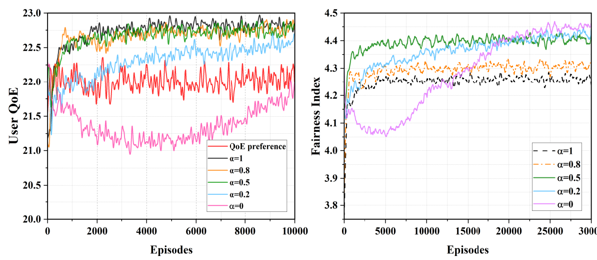
GreedyMSMC [23]: This work considers the fairness issue arising from connections between multiple users and different servers. It adopts a proportional fairness approach to allocate base station resources, thereby achieving coordinated resource allocation between multiple users and servers. However, it is based on maximizing individual QoE to achieve proportional fairness and does not consider the perspective of user preference. We implement this algorithm in our proposed collaborative system and compare its performance on QoE and fairness.

As shown in Fig. 2 (a), the average QoE index of our proposed Q2D3 algorithm eventually converges to 22.7. The QoE index of the DRL-CTCS algorithm finally converges to 22.5, which is slightly lower than our proposed Q2D3



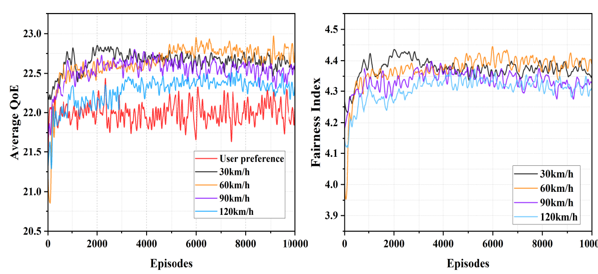
(a) QoE under different methods (b) Fairness under different methods (c) Delay under different methods (d) Hit probability

Fig. 2: Performance metrics under different methods.



(a) QoE under different α (b) Fairness under different α

Fig. 3: Performance Comparison: Parameters.



(a) QoE under different speed (b) Fairness under different speed

Fig. 4: Performance Comparison: Speed.

algorithm. It can be observed that the user's QoE preference fluctuates around 22, clearly demonstrating that our Q2D3 algorithm can satisfy the user's preferences well. However, under the Greedy algorithm, the user's QoE index converges to 21.5, and under the MA algorithm, the user's QoE index converges to 19.8, both of which fail to meet the user's QoE preference. In summary, our proposed Q2D3 algorithm can satisfy the user's QoE preference under the edge collaboration scenario.

Furthermore, Fig. 2 (b) clearly shows the fairness indices of the four algorithms. Our proposed Q2D3 algorithm demonstrates outstanding performance, eventually converging to 4.3 after 10,000 rounds of training iterations. The performance of the DRL-CTCS algorithm is comparable to Q2D3. However, the fairness index of the GreedyMSMC algorithm is only 4.0, which is slightly better than the worst-performing MA algorithm at 3.2. Evidently, our algorithm also exhibits remarkable performance in terms of fairness. In terms of user access latency, as shown in Fig. 2 (c), the Q2D3 algorithm has the lowest latency, averaging 61ms. The MA algorithm has the highest latency, averaging 220ms. We see that the latency performance of the Q2D3 algorithm based on the edge collaboration server is much higher than that of the other three algorithms. Coincidentally, under the characteristic of edge server collaboration, our proposed Q2D3 algorithm also exhibits excellent performance in terms of cache hit rate, with an average cache hit rate as high as 88%. The average cache hit rates of the other three algorithms are 85%, 71%, and 82% respectively, all lower than our proposed algorithm. It is evident that our proposed Q2D3 algorithm demonstrates good performance in four aspects - user preference, fairness, latency, and cache hit rate.

C. Parameter comparison of Q2D3

The optimization objective of our proposed Q^2oE model, as shown in Eq. (6), includes two important parameters α and β in this optimization objective. The size of parameter α determines the user QoE while the size of parameter β determines the fairness metric. In the work of this paper, we use $\alpha = 0.5$ and $\beta = 0.5$, which best values after comparison. In order to select the optimal parameters, we respectively set the values of α as 1, 0.8, 0.5, 0.2, and 0, while maintaining $\alpha + \beta = 1$. As shown in Fig. 3, we conducted experiments based on two key aspects: the user preference-based QoE metric and fairness. In Fig. 3 (a), the user QoE (22.7) reaches its maximum when the value of α is 1. However, the fairness index (4.26) is lower for the values of 0.5 and 0.8. When the value is 0.8, the user QoE (22.6) decreases slightly from α = 1, but the fairness index (4.31) increases slightly. When α value of 0.5, performance is excellent in both QoE and fairness. It can be observed that $\alpha = 0.5$ strikes the best balance between user QoE and fairness index. Therefore, we set $\alpha = 0.5$ and $\beta = 0.5$ as the optimal parameter for the proposed Q^2oE model.

Moreover, vehicle speed is an important benchmark in vehicular networks. Therefore, we conducted simulation experiments with different average vehicle speeds, specifically 30km/h, 60km/h, 90km/h, and 120km/h. Through the above experiments, we verified the performance of our system at different vehicle speeds. Fig. 4 (a) illustrates the QoE metric convergence process under different vehicle speeds. It can be observed that when the average speed is 30km/h, the curve eventually converges to 22.6. When the average speed is 60km/h, the curve converges to 22.75. At 90km/h average speed, the final convergence is 22.5. And at 120km/h average speed, the curve converges to 22.25. Although the user QoE metric decreases slightly with the increase of average vehicle speed, it remains notably higher than the user's QoE preference value (22.0) under all speed conditions. So we demonstrate that, our system can satisfy the user's preferences well under different vehicle speeds. Fig. 4 (b) presents the performance curves of the fairness metric under different vehicle speeds. Under the four scenarios of 30km/h, 60km/h, 90km/h and 120km/h, the fairness metric converges to 4.35, 4.39, 4.32 and 4.30, respectively. As the vehicle speed increases, the user fairness metric shows a slight decrease but remains significantly higher than the three benchmark algorithms compared above. In summary, our system adapts well under different vehicle speeds, demonstrating favorable performance in both user preference and fairness metrics.

VI. CONCLUSION

User-level QoE preference and QoE fairness are important factors to consider in multi-user scenarios. We designed a novel QoE model, named the Q^2oE model, which successfully combines user preference and fairness metrics. Finally, we deployed the Q^2oE model and Q2D3 algorithm on collaborative edge servers with our simulation results showing that, our method improves QoE fairness while guaranteeing user-level QoE preference.

REFERENCES

[1] U. Cisco, "Cisco annual internet report (2018–2023) white paper," *Cisco: San Jose, CA, USA*, vol. 10, no. 1, pp. 1–35, 2020.

[2] L. Sun, F. Duanmu, Y. Liu, Y. Wang, Y. Ye, H. Shi, and D. Dai, "Multi-path multi-tier 360-degree video streaming in 5g networks," in *Proceedings of the 9th ACM multimedia systems conference*, 2018, pp. 162–173.

[3] D. Liu, X. Tan, Y. Liu, S. Wang, Q. Zheng, and Q. Shi, "Joint upload-download transmission scheme for low-latency mobile live video streaming," in *2023 IEEE/ACM 31st International Symposium on Quality of Service (IWQoS)*, 2023, pp. 1–10.

[4] W. Robitza, M.-N. Garcia, and A. Raake, "A modular http adaptive streaming qoe model—candidate for itu-t p. 1203 ("p. nats")," in *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2017, pp. 1–6.

[5] N. Eswara, S. Ashique, A. Panchbhai, S. Chakraborty, H. P. Sethuram, K. Kuchi, A. Kumar, and S. S. Channappayya, "Streaming video qoe modeling and prediction: A long short-term memory approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 661–673, 2019.

[6] Y. Yuan, W. Wang, Y. Wang, S. S. Adhatarao, B. Ren, K. Zheng, and X. Fu, "Vsim: Improving qoe fairness for video streaming in mobile environments," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2022, pp. 1309–1318.

[7] K. Spiteri, R. Urgaonkar, and R. K. Sitaraman, "Bola: Near-optimal bitrate adaptation for online videos," *IEEE/ACM transactions on networking*, vol. 28, no. 4, pp. 1698–1711, 2020.

[8] E. Baccour, A. Erbad, K. Bilal, A. Mohamed, and M. Guizani, "Pccp: Proactive video chunks caching and processing in edge networks," *Future Generation Computer Systems*, vol. 105, pp. 44–60, 2020.

[9] H. Zhao, Q. Wang, J. Wang, B. Wan, and Z. Wu, "Popularity-based and version-aware caching scheme at edge servers for multi-version vod systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 3, pp. 1234–1248, 2020.

[10] J. Cai, M. Xiao, H. Sun, J. Shao, Y. Zhao, and T. Zhang, "Video streaming caching and transcoding for heterogeneous mobile users," in *2023 IEEE 29th International Conference on Parallel and Distributed Systems (ICPADS)*, 2023, pp. 2243–2250.

[11] X. Zuo, J. Yang, M. Wang, and Y. Cui, "Adaptive bitrate with user-level qoe preference for video streaming," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2022, pp. 1279–1288.

[12] L. Huo, Z. Wang, M. Xu, Y. Li, Z. Ding, and H. Wang, "A meta-learning framework for learning multi-user preferences in qoe optimization of dash," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 9, pp. 3210–3225, 2020.

[13] J. Kim, S.-C. Kwon, and G. Choi, "Performance of video streaming in infrastructure-to-vehicle telematic platforms with 60-ghz radiation and ieee 802.11 ad baseband," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 10 111–10 115, 2016.

[14] F. Y. Yan, H. Ayers, C. Zhu, S. Fouladi, J. Hong, K. Zhang, P. Levis, and K. Winstein, "Learning in situ: a randomized experiment in video streaming," in *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*, 2020, pp. 495–511.

[15] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson, "A buffer-based approach to rate adaptation: Evidence from a large video streaming service," in *Proceedings of the 2014 ACM conference on SIGCOMM*, 2014, pp. 187–198.

[16] C. Qian, D. Liu, and H. Jiang, "Harmonizing energy efficiency and qoe for brightness scaling-based mobile video streaming," in *2022 IEEE/ACM 30th International Symposium on Quality of Service (IWQoS)*. IEEE, 2022, pp. 1–10.

[17] S. Cicalò, N. Changuel, V. Tralli, B. Sayadi, F. Faucheux, and S. Kerboeuf, "Improving qoe and fairness in http adaptive streaming over lte network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 12, pp. 2284–2298, 2015.

[18] A. Mehrabi, M. Siekkinen, and A. Ylä-Jääski, "Joint optimization of qoe and fairness through network assisted adaptive mobile video streaming," in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2017, pp. 1–8.

[19] W. J. Yun, D. Kwon, M. Choi, J. Kim, G. Caire, and A. F. Molisch, "Quality-aware deep reinforcement learning for streaming in infrastructure-assisted connected vehicles," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 2, pp. 2002–2017, 2021.

[20] K. Tan, D. Bremner, J. Le Kernec, Y. Sambo, L. Zhang, and M. A. Imran, "Intelligent handover algorithm for vehicle-to-network communications with double-deep q-learning," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 7, pp. 7848–7862, 2022.

[21] H. Sun, M. Xiao, Y. Xu, G. Gao, and S. Zhang, "Privacy-preserving stable crowdsensing data trading for unknown market," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*, 2023, pp. 1–10.

[22] ——, "Crowdsensing data trading for unknown market: Privacy, stability, and conflicts," *IEEE Transactions on Mobile Computing*, pp. 1–16, 2024.

[23] A. Mehrabi, M. Siekkinen, and A. Ylä-Jääski, "Joint optimization of qoe and fairness through network assisted adaptive mobile video streaming," in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2017, pp. 1–8.

[24] T. Yang, Z. Tan, Y. Xu, and S. Cai, "Collaborative edge caching and transcoding for 360° video streaming based on deep reinforcement learning," *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 25 551–25 564, 2022.

[25] V. Nathan, V. Sivaraman, R. Addanki, M. Khani, P. Goyal, and M. Alizadeh, "End-to-end transport for video qoe fairness," in *Proceedings of the ACM Special Interest Group on Data Communication*, 2019, pp. 408–423.

[26] X. Li, X. Wang, S. Xiao, and V. C. Leung, "Delay performance analysis of cooperative cell caching in future mobile networks," in *2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 5652–5657.