This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

EDGE360: Edge-Enabled Multi-Agent DRL for Region-Aware Rate Adaptation Solution to Enhance Quality of 360° Video Streaming

Fazal E Subhan*, Abid Yaqoob*, Cristina Hava Muntean§, and Gabriel-Miro Muntean*
*School of Electronic Engineering, Dublin City University, Dublin, Ireland
§School of Computing, National College of Ireland, Dublin, Ireland
fazal.subhan2@mail.dcu.ie, abid.yaqoob@dcu.ie, cristina.muntean@ncirl.ie, gabriel.muntean@dcu.ie

***Abstract*—Optimal tile-based bitrate allocation improves the Quality of Experience (QoE) for adaptive 360° video streaming across multiple clients in heterogeneous network environments; however, it is challenging as it implies accurate viewport prediction, finest tile-based bitrate reservation, and maintaining QoE fairness, particularly under constrained network conditions. This paper proposes a strategy named EDGE360, that employs an edge-driven Multi-Agent Deep Reinforcement Learning (MADRL) solution for rate adaptation to improve the joint QoE in DASH-based rich media content delivery based on adaptive viewport prediction and Video Multi-method Assessment Fusion (VMAF) corresponding tiling granularity selection. Cooperative strategies among agents in the central critic network are crucial for addressing the complexity of network instances at the edge and optimizing media streaming bitrate assignment in multiple-client scenarios. Therefore, EDGE360 aims to implement the Counterfactual Multi-Agent Policy Gradients (COMA) based on 5G network traces to train agents in policies that optimize individual client QoE and fairness among clients, resulting in an improved rich streaming experience. At the edge, a tile-based quality monitor evaluates viewport trajectories, buffer status, and network throughput, employing deep learning to forecast optimal tile bitrate allocation, which is formulated as an MDP and solved with MADRL. Based on extensive experimentation, EDGE360 surpasses state-of-the-art adaptive bitrate algorithms by achieving the highest average reward, outperforming RAPT360, 360SRL, and BOLA360 by 8.12%, 11.86%, and 18.00%, respectively, demonstrating superior convergence and refinement.**

***Index Terms*—MPEG DASH, QoE Fairness, Bitrate adaptation, Edge Computing, Deep Reinforcement Learning**

I. Introduction

THE rapid growth of immersive multimedia technologies, notably 360° virtual reality (VR) video streaming, has influenced how individuals interact with rich media content [1]. These applications necessitate an extremely high Quality of Experience (QoE), since they immerse consumers in fully interactive worlds with a wide field of view. However, providing such high-fidelity VR video across numerous clients with varying network circumstances presents considerable constraints. Providing continuous streaming, controlling bandwidth, and ensuring adequate quality distribution across clients become critical considerations, especially in bandwidth-constrained contexts. Furthermore, the widespread implementation of HTTP-based adaptive streaming (HAS) with sophisticated adaptive bitrate adaptation (ABR) techniques has substantially enhanced individual client QoE [2]. However, client-centric ABR solutions overlook the larger context of shared bandwidth constraints, resulting in competitive situations between clients and accompanying disparities, unfairness, and ineffective bandwidth utilization. The growing world of latency-sensitive rich media content delivery across 5G and beyond networks complicates guaranteeing fair QoE distribution across multiple clients.

Edge-based ABR has become a vital enabler for efficient real-time 360° VR video delivery. By processing data closer to users, edge computing reduces latency and offloads computation from centralized clouds, addressing the high resource demands of VR streaming. It facilitates localized, dynamic rate adaptation essential for bandwidth-efficient and real-time QoE optimization in 360° video systems. Unlike traditional cloud centers, edge nodes reside near end users and execute advanced DRL-based algorithms [3]–[5] to predict network conditions and manage client requests for precise QoE control. This paradigm reduces transmission redundancy and bandwidth bottlenecks [6], [7]. However, current edge solutions still struggle with accurate network prediction, viewport forecasting, and client state awareness for optimal ABR decisions, especially under multi-client scenarios.

Furthermore, adaptive 360° video streaming divides each frame into smaller units or tiles that can be delivered at varying bitrates depending on the user's viewport. Accurate viewport prediction is critical because it allows the system to dedicate more resources to tiles in the user's FoV while lowering the quality of tiles in peripheral regions and, hence, bandwidth usage. In [8], the authors proposed a user-aware viewport prediction technique that combines tile-based adaptation and QoE measurements for more precise bandwidth allocation in 360° video streaming. This tile-based rate adaptation is critical for providing a seamless and immersive experience, but it brings challenges such as efficient bitrate allocation, error-prone long-term prediction, managing numerous clients, and guaranteeing fair quality distribution. Multi-user immersive VR experiences present additional complexities in synchronization, latency, and bandwidth contention, particularly within wireless and edge-assisted environments [9].

To address the aforementioned challenges, EDGE360 proposes to use edge-assisted Multi-Agent Deep Reinforcement

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

Learning (MADRL) to perform localized decision-making and rate adaptation in order to improve individual as well as collective QoE. EDGE360's key development is the application of Counterfactual Multi-Agent Policy Gradients (COMA) [10] to train agents in policies that optimize bitrate allocation based on viewport prediction, tile quality assessment with Video Multi-method Assessment Fusion (VMAF) scores [11], and overall client fairness. COMA is well-suited for this role because it addresses the multi-agent credit assignment challenge by analyzing each agent's value to the global reward, as opposed to standard DRL systems such as A3C [12] or DDQN [13], which are based on single-agent conditions. Furthermore, COMA lowers agent interference by focusing on counterfactual incentives, making it more successful in settings with complex interdependence, such as optimizing joint QoE in multi-client settings. This cooperative, multi-agent technique is well-suited for complicated, multi-client settings in heterogeneous 5G and Beyond networks.

Furthermore, standard ABR algorithms, which are usually centralized, have limits when dealing with highly dynamic and resource-intensive media 360° video. They frequently fail to compensate for the various network circumstances and device capabilities in current video streaming situations. EDGE360 overcomes these restrictions by evaluating viewport feedback and network information in real-time, resulting in much lower latency and improved bitrate allocation efficiency. This study also describes a tile-based quality monitor at the edge that dynamically uses client buffer status, network performance, and other QoE variables to anticipate and allot appropriate bitrates to distinct tiles. EDGE360's deep reinforcement learning architecture reformulates the adaptive bitrate selection problem in 360° VR video streaming as a Markov Decision Process (MDP), enabling context-aware decisions that are highly responsive to fluctuating network conditions and dynamic user viewport behaviours.

The main contributions of this paper are as follows:

- We introduce EDGE360, a novel edge-assisted MADRL framework designed for adaptive tile-wise bitrate management in multi-user 360° VR video streaming. Unlike previous approaches, EDGE360 incorporates fairness as a primary optimization goal by integrating both individual deviation-based penalties and the global Jain's fairness index into the COMA policy gradient algorithm. This ensures equity in both personalized and system-wide QoE. Through the use of COMA's counterfactual advantage estimation, EDGE360 effectively isolates the impact of each agent on global performance. This enables more accurate credit assignment and cooperative behavior, leading to substantial improvements in fairness when compared to traditional independent actor-critic DRL frameworks.
- We employ a VMAF-guided video quality monitoring approach that incorporates a tile-size granularity-aware dynamic adjustment of region-specific bitrate allocations based on long-short-term memory (LSTM)-driven viewport prediction. This method of region-aware adaptation enhances bandwidth efficiency and perceptual significance by assigning fine, medium, and coarse tile granularities to the viewport, marginal, and background regions, respectively. This approach ensures optimal video quality delivery within real-time operational constraints.
- We model the streaming problem as a cooperative MDP, with distributed actor networks generating parallelised, client-specific actions while communicating with a centralised critic. This decentralised execution architecture enables scalability, low-latency inference, and resilience in high-density multi-client environments, as demonstrated by comprehensive ablation, sensitivity, and scalability evaluation.

II. Related Works

Extensive research has been undertaken in the realm of adaptive rich media streaming and client QoE, emphasizing improving multimedia transmission efficiency by employing machine learning-based ABR techniques. A lot of research investigates the challenges of multi-client bandwidth congestion and suggests customized bitrate adaptation techniques for different network scenarios. These works focus on three key deployment paradigms for bitrate optimization agents intended to improve QoE: server-side, client-side, and network-assisted solutions. Each of these paradigms offers valuable insights into improving transmission efficiency for multimedia transmission. This section highlights the most recent advances in each of these three techniques.

A. Server-Side ABR Streaming Approaches

Multimedia content can be sent by servers with multiple bitrate versions and encoded so that clients can request the requisite chunks. Zou et al. [14] addressed streaming tile-based 360° videos to several viewers who share network resources using server-side rate adaptation. The study optimizes tile rate allocation using a non-linear optimization problem to minimize visual distortion while taking into account transmission constraints and user-specific requirements. They applied a CNN-based viewpoint prediction framework and a Laplace probability for error modelling to achieve near-optimal QoE performance. In order to maximize effectiveness and QoE fairness, Altamimi et al. [15] developed and implemented a client-server side DRL-based ABR solution for adaptive video streaming. Also, Yuan et al. [16] presented an end-to-end method that divides bottleneck bandwidth among users according to mobility and QoE characteristics to maximize mobile video streaming and QoE fairness in heterogeneous networks. Currently, server-side infrastructure is not widely used for collaborative QoE augmentation techniques, mainly because estimating bottleneck capacities across many access networks is challenging. Moreover, the implementation of real-time traffic management for a significant number of clients has serious computing costs at the server end, making these practices not viable from an economic standpoint and low latency multimedia applications.

B. Client-Side ABR Streaming Approaches

Clients in 360° video streaming dynamically adapt tile quality based on viewport prediction and network conditions,

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

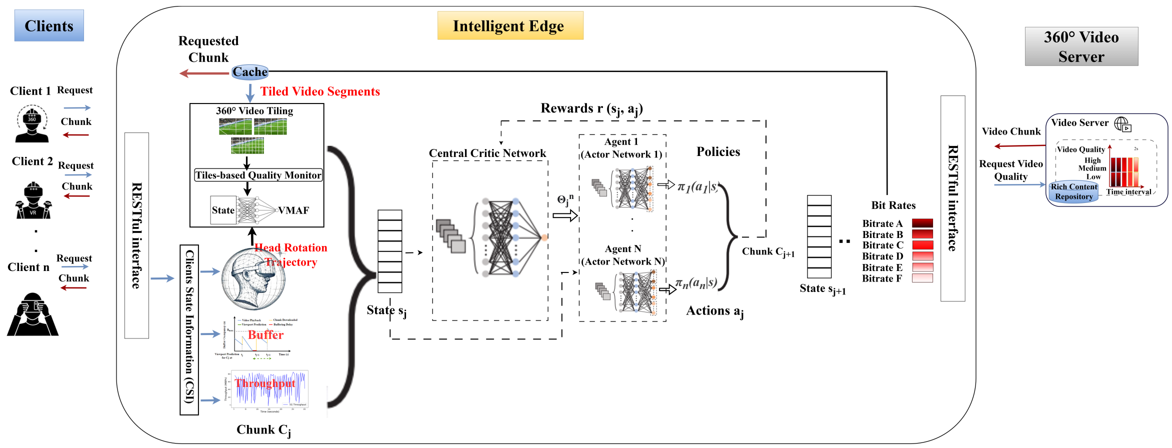


Fig. 1: EDGE360 video streaming architecture which comprises a CDN server, smart edges, and multiple rich media clients.

making tile-based rate adaptation attractive for its scalability and deployment ease. Client-side tile-based streaming with viewport adaptation has been extensively explored, notably in heuristic designs like FLARE [17], which allocate tile bitrates based on predicted viewports to enhance bandwidth efficiency. However, this design lacks multi-user coordination and resilience under variable 5G conditions, limiting its scalability and QoE stability. 360SRL [18], which serves as a baseline approach here, introduces a sequential RL approach using DDQN to tackle the exponential action space in tile-based 360° streaming. While it improves tile decision efficiency, it lacks a region-prioritized strategy with an effective perceptual quality VMAF metric and does not integrate collaborative multi-agent reward structures for fairness. RAPT360 [2], a baseline approach here, utilizes an A3C-based RL framework with LSTM to capture temporal features and predict user head movement. RAPT360 makes independent bitrate decisions for each tile without enforcing spatial consistency leads to incoherent visual quality across adjacent tiles, resulting in noticeable quality flicker and reduced perceptual smoothness. It lacks multi-user coordination, limiting its effectiveness in optimizing QoE in fairness-sensitive streaming environments. Li et al. [19] proposed a multi-agent reinforcement learning strategy for fair bandwidth distribution among concurrent 360° video users, incorporating rebuffering, viewport trajectory, and video quality preferences. To address head motion uncertainty, Wang et al. [20] introduced a MADRL-based ABR framework using a multi-modal spatio-temporal attention transformer for viewpoint prediction, significantly improving QoE but remaining sensitive to abrupt motion changes. BOLA360. [21] is a heuristic baseline buffer-driven online ABR algorithm model, aiming to maximize QoE by optimizing tile downloads within the user's field of view (FoV). However, it lacks predictive viewport modeling, perceptual quality metrics, region-aware tiling, and multi-user coordination, limiting its performance in dynamic streaming scenarios. Although client-side solutions are easily deployable without server or network modifications, they lack global context across users, making them suboptimal for complex, latency-sensitive content. These limitations underline the need for scalable approaches that jointly consider multi-client dynamics and avoid preference biases that may compromise QoE fairness.

C. Network-Assisted ABR Streaming Approaches:

Network-assisted techniques are essential for scalable 360° video streaming, addressing high bandwidth demands and multi-user dynamics. Such strategies enhance user engagement and delivery efficiency by leveraging network insights. Bi et al. [22] proposed a two-stage ABR scheme with FoV tile prediction and pre-caching in MEC environments, using a dual-agent DRL framework to adapt bitrates against wireless fading and prediction errors. Similarly, [23] introduced a reinforcement learning-based MEC framework integrating time-series FoV prediction and bitrate adaptation to reduce bandwidth costs and improve streaming reliability. However, these methods primarily target single-user scenarios, limiting scalability in multi-user contexts. Shi et al. [24] developed a meta-RL-based edge analytics framework to address dynamic QoE trends, but it also lacks multi-user coordination. More recent work [25]–[28] explored MADRL-based edge ABR strategies to jointly optimize QoE and fairness. Despite promising results, these approaches often underperform under high user loads and do not incorporate rich media characteristics. In contrast, our edge-based solution employs a COMA MADRL framework and VMAF-driven tile adaptation to support scalable, real-time 360° streaming for multiple users, delivering superior QoE and fairness.

III. Proposed EDGE360 System Design

A. Overview

Fig. 1 illustrates the EDGE360 architecture for adaptive 360° VR streaming to multiple clients via an intelligent edge node. Using a multi-agent DRL-based ABR algorithm, the edge leverages client-specific data viewport prediction, buffer status, and bandwidth to ensure seamless playback and optimal video quality. Acting as an intermediary to the central 360° video server, the edge dynamically responds to network and client variations to maintain high QoE. The system comprises three core components: clients, the intelligent edge, and the VR video server.

1) Clients: Each client in the EDGE360 system represents an end-user device that can stream and request VR content, such as a 360° viewer or VR headset. Leveraging a RESTful interface, the client requests particular segments of the 360°

video from the smart edge. To optimize the streaming experience, each client periodically provides status information to the edge by employing the Client State Information (CSI) module. This client state data comprises clients' throughput, buffer status, and viewport trajectories. Throughput data provides dynamic bitrate adaptation based on network circumstances, resulting in smoother viewing. Buffer status allows modifying video quality to avoid disruptions, whereas viewport trajectories enable the edge to prioritize high-quality tiles inside the client's field of view. This real-time status data is critical for the edge of optimizing resource allocation and maximizing QoE across multiple clients.

2) EDGE360: In the EDGE360 architecture, the intelligent edge serves as the central processing unit that handles client requests, performs adaptive bitrate streaming, and delivers optimized 360° video content. It leverages client-specific variables, such as viewport prediction, buffer status, and bandwidth availability, to make latency-aware, resource-efficient quality decisions. A core function of the edge is 360° video tiling and quality monitoring. The video is divided into spatial tiles, enabling selective quality adaptation: higher resolution is assigned to tiles within the client's viewport, and lower quality to peripheral tiles. To assess tile quality, the edge employs a VMAF-based quality monitor, which evaluates perceptual metrics like sharpness, contrast, and structural fidelity. This tile-specific VMAF-driven adaptation ensures high visual quality in the field of view while conserving bandwidth by reducing quality in non-visible regions.

The COMA policy network is a specialized MADRL model implanted at the edge that manages cooperative decision-making for adaptive streaming. To carry out this task, the EDGE360 system generates a state $\mathcal{S}_{\mathrm{j}}$ for each client session based on real-time data. In the COMA framework, each client serves as a separate agent with its actor-network that learns a policy $\pi_n(a_n \mid s_n)$, where a_n denotes actions like bitrate and tile quality selection for the current video chunk $\mathcal{C}_{\mathrm{j}}$. After selecting the action a_j, the system updates the state to $\mathcal{S}_{\mathrm{j+1}}$ based on an updated CSI module and a tiles-based quality monitor. The actor-network then repeats the process with this updated state for the next chunk $\mathcal{C}_{\mathrm{j+1}}$.

This strategy iteratively tries to maximize reward $r(s_j, a_j)$ in given current network circumstances. A central critic network assesses collective agent performance by computing the reward $r(s_j, a_j)$, incorporating measures such as video optimal streaming, tiled-chunk quality, and re-buffering minimization. The critic network computes an Advantage Function $\theta = A_n(s, a_n) = U_j(s, a_n) - \sum_{a_n'} \pi_n(a_n' \mid s) U_n(s, a_n')$, where $U_j(s, a_n)$ is the value-action function of actor n, and $\sum_{a_n'} \pi_n(a_n' \mid s) U_n(s, a_n')$ is a baseline to identify the expected value of alternative actions resulting in isolating the influence of action a_n on overall performance. This function θ evaluates the efficiency of each agent's actions in terms of rewards, leading the system toward optimal resource allocation and adaptability across multiple clients.

Moreover, one of the edge's principal functions is to maintain a cache of video chunks. This cache momentarily caches frequently visited video items, reducing the need to send multiple queries to the central 360° video server. Whenever a client requests a video chunk, the edge node initially determines if it is accessible in the cache. If it is, the chunk is provided directly to the client, which results in quicker response times and less network demand for the central server. This caching approach is convenient when multiple viewers request overlapping content, since it allows for an effective reprise of cached video segments.

3) 360° Video Server: The rich VR 360° video server is the primary source of 360° video material and includes a rich content repository with multiple quality levels (high, medium, and low) for each video tile. Whenever the desired video chunk cannot be located in the edge's cache, the edge node retrieves it from the server, suggesting the quality required based on the COMA policy recommendations. The server dynamically reacts to client requests by supplying the appropriate video quality to ensure each client achieves the highest degree of detail to enhance the viewing experience. The server allows the edge to scale effectively by transmitting only the required quality levels for each tile rather than an entire video with maximum quality. The server optimizes the bandwidth consumption and minimizes redundant data transmission by keeping a repository with split quality levels.

IV. EDGE360-Driven Bitrate Adaptation and QoE Computation

A. QoE Model Design

EDGE360's adaptive 360° streaming across multiple clients optimises fluctuating QoE factors while ensuring fairness. Clients request tiles based on expected viewports, and the network must use bandwidth effectively to reduce rebuffering and latency. EDGE360 integrates tile-level utility based on viewport relevance and content complexity, inspired by perceptual insights provided in [29], which highlight saliency-aware quality adaptation. ABR decisions are influenced by COMA, which was chosen over alternatives such as A3C and DDQN for its ability to provide counterfactual advantages that segregate each agent's contribution to the overall reward. This enables fine-grained credit assignment and coordinated policy learning, both of which are crucial when utilising shared bandwidth and cache constraints. Decentralised actor networks determine per-client bitrates, while a centralised critic assesses the global state. EDGE360 calculates QoE for a multi-client scenario in steps, beginning with the network throughput distribution and moving through tile granularity, video quality switches, buffer dynamics, and delay, eventually combining these aspects into a comprehensive QoE metric. A detailed breakdown of the QoE model components and derivations is provided in Appendix A.

B. QoE Calculation

In EDGE360, client-specific QoE estimation is essential for assessing streaming performance. The QoE model integrates factors such as effective tile quality, viewport enhancement, quality switches, rebuffering, and latency, key to ensuring a smooth and immersive experience under network variability. These metrics support fairness and effectiveness in multi-client scenarios. For client n at time t, QoE is estimated as follows:

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

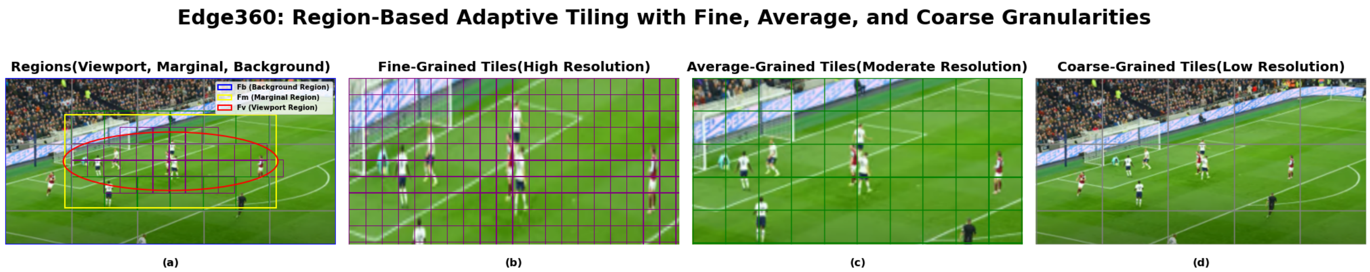


Fig. 2: (a) shows the segmentation of a 360° video frame into three spatial regions: *Viewport* ($\mathcal{F}_v$) with fine-grained tiles, *Marginal* ($\mathcal{F}_m$) with average-grained tiles, and *Background* ($\mathcal{F}_b$) with coarse-grained tiles. (b)–(d) illustrate region-specific tiling, where finer tiles are allocated to high-priority regions for perceptual quality, and coarser tiles to low-priority areas for bitrate efficiency. This demonstrates EDGE360's hierarchical tiling strategy for optimizing 360° VR streaming.

$$\mathcal{Q}oE_n(t) = \mathcal{Q}_v(t) - \alpha \cdot \Delta\mathcal{Q}_{j,k}(t) - \beta \cdot \rho_n(t)' - \zeta \cdot \mathcal{D}_n(t) \quad (1)$$

Here, $\mathcal{Q}_v(t)$ represents the aggregated perceptual quality for the predicted viewport tiles, as introduced in Eq. A.11. Eq. A.13, $\Delta\mathcal{Q}_{j,k}(t)$ tracks the variance in tile quality between segments and is penalized by α. Quality stability reduces visual interruptions, improving the viewing experience [30].

As defined in Eq. A.15, $\rho_n(t)$ indicates a rebuffering event for client n at time t, with its duration given by $\delta_n(t)$. Their product models the stall penalty in QoE, weighted by β to discourage frequent and prolonged playback interruptions. Latency $\mathcal{D}_n(t)$ was introduced in Eq. A.17 as the sum of transmission delay and processing delay and is weighted by ζ. This term highlights the significance of response time in adaptive streaming, particularly in a multi-client immersive environment. To evaluate the system-wide QoE level, the aggregated QoE was defined as $\mathcal{Q}\hat{o}E_n(t)$, measuring the overall QoE across all clients i.e., $\mathcal{Q}\hat{o}E_n(t) = \frac{1}{N}\sum_{n=1}^{N} \mathcal{Q}oE_n(t)$. This metric gives a system-wide (global) perspective on video quality and adaptation efficiency.

C. Fairness Assessment

To ensure fairness in resource allocation among multiple clients, Jain's Fairness Index is used as a well-established metric to quantify the equality of QoE across all clients. For $\mathcal{N}$ clients, the fairness index at time t is defined as:

$$\hat{\mathcal{F}}_{\text{Jain}}(t) = \frac{\left(\sum_{n=1}^{\mathcal{N}} \mathcal{Q}oE_n(t)\right)^2}{\mathcal{N} \cdot \sum_{n=1}^{\mathcal{N}} \mathcal{Q}oE_n^2(t)} \quad (2)$$

Here, $\mathcal{Q}oE_n(t)$ denotes the perceived quality of experience of client n at time t, which incorporates viewport-aware video quality, rebuffering impact, delay penalties, and quality switches. A Jain's index value close to 1 indicates an equitable QoE distribution among clients, whereas lower values reflect increasing disparity.

This formulation aligns with EDGE360's goal to not only optimize individual client performance but also ensure system-wide fairness in perceived video quality, avoiding resource bias and improving overall user satisfaction across diverse network and client conditions [31], [32].

D. EDGE360 Reward Function

The EDGE360 reward function is introduced to help optimise each client's QoE while preserving fairness across multiple clients in a resource-shared network. The reward function is preferable for adaptive streaming in a multi-client setting as it takes into account the QoE of each client, the consistency of QoE among clients, and fairness. The compact reward function for client n at time t is as follows:

$$\mathcal{R}_n(t) = \mathcal{Q}oE_n(t) - |\varepsilon| \cdot \frac{\left|\mathcal{Q}oE_n(t) - \mathcal{Q}oE_n(t)^k{}_{\text{avg}}\right|}{\mathcal{Q}oE_{\max,n}(t) - \mathcal{Q}oE_{\min,n}(t)} + \lambda \cdot \hat{\mathcal{F}}_{\text{Jain}}(t), \forall k = 1, 2, \ldots, N \quad (3)$$

In this context, ε functions as a local fairness penalty term, which quantifies the divergence of client n's instantaneous QoE from the moving average QoE of other clients, denoted as $\mathcal{Q}oE_n(t)^k{}_{\text{avg}}$. This term imposes penalties for significant discrepancies between individual clients within short temporal intervals, and is standardized using the observed maximum and minimum QoE values for the client, represented by $\mathcal{Q}oE_{\max,n}(t)$ and $\mathcal{Q}oE_{\min,n}(t)$, respectively. Furthermore, a global fairness measure across all clients is imposed via the term $\hat{\mathcal{F}}_{\text{Jain}}(t)$, specifically Jain's Fairness Index, which is computed based on system-wide QoE distributions at time t and weighted according to λ. While the local penalty promotes fairness among proximate clients within a rolling time window, the global term ensures prolonged equity in bandwidth distribution across all clients.

Together, these two fairness components allow EDGE360 to achieve both short-term fairness responsiveness and long-term global balance, maintaining efficient and fair system operation under dynamic network conditions.

E. Adaptive Bitrate Allocation for Predicted Regions in EDGE360

In EDGE360, efficient bitrate allocation is essential for delivering high-quality 360° video streaming while optimizing bandwidth usage. To accomplish this, each video frame is partitioned into three predicted regions: the viewport region ($\mathcal{F}v$), marginal region ($\mathcal{F}m$), and background region ($\mathcal{F}b$), prioritized based on their relevance to the user's FoV. The adaptive tiling mechanism assigns optimized bitrates $\mathcal{R}j, k^v$, $\mathcal{R}j, k^m$, and $\mathcal{R}j, k^b$ to the respective tiles in each region, balancing visual quality with bandwidth efficiency.

The available bitrate set, $\mathcal{R} = \mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_M$, provides discrete options for allocation across the three regions. The complete tile set is defined as $T = T^v, T^m, T^b$, where each subset corresponds to a specific region. Tiles in T^v (viewport) are assigned the highest priority and bitrate to preserve fine visual quality. Tiles in T^m (marginal) lie adjacent to the viewport and receive intermediate bitrates to maintain contextual continuity. Lastly, T^b (background) tiles are outside the immediate FoV and are allocated the lowest bitrates to conserve bandwidth. This hierarchical structure enables EDGE360 to focus resources where they matter most enhancing the user's perceived quality in the FoV while efficiently managing network load across peripheral regions.

As shown in Fig. 2, the viewport region $\mathcal{F}_v$ (red ellipse) receives fine-grained, high-resolution tiles to prioritize user focus. The marginal region $\mathcal{F}_m$ (yellow rectangle) is streamed with moderate granularity, while the background $\mathcal{F}_b$ uses coarse tiles to conserve bandwidth. Fig. 4 illustrates EDGE360's bitrate allocation for three test videos (*Sunset*, *Alaska*, and *Beach*), assigning 18–20 Mbps to F_v, 8–12 Mbps to F_m, and 2.5–4 Mbps to F_b, balancing visual quality and bandwidth efficiency across regions.

Algorithms 1, parts 1 and 2, present the detailed procedure of the proposed viewport-aware adaptive tiling scheme in EDGE360. The algorithm returns the tile set $T = \{T^v, T^m, T^b\}$, where each subset corresponds to a different viewport-aware region: T^v (viewport tiles) forms the predicted user viewport, T^m (marginal tiles) covers adjacent areas with moderate granularity, and T^b (background tiles) consists of lower-priority tiles beyond the predicted FoV. These selected tiles collectively reconstruct a complete video chunk, ensuring optimized visual quality and efficient bandwidth utilization in EDGE360.

This systematic approach ensures efficient bandwidth utilization while maintaining high-quality streaming for the user's viewport. By dynamically adjusting bitrates and calculating total file sizes based on tile regions and priorities, EDGE360 achieves an optimal balance between visual quality and resource constraints. This method enables the system to adapt to varying network conditions and user demands dynamically, ensuring improved QoE.

V. EDGE360 Rate Adaptation and Problem Formulation

In the EDGE360 adaptive streaming framework, each client selects optimal tiled-region bitrates per adaptation span to maximize aggregated QoE and ensure immersive 360° VR experiences. Leveraging the COMA framework [33], EDGE360 jointly optimizes rate adaptation across multiple clients by balancing individual demands with system-wide fairness under dynamic network and viewport conditions. As shown in Fig. 3, COMA integrates both local (e.g., throughput, bitrate, viewport probability, quality scores) and global (e.g., predicted viewport maps, average QoE, chunk size, fairness, delay, edge cache usage) state inputs within a MADRL setup to learn effective streaming actions.

The EDGE360 optimization problem may be represented quantitatively as follows:

Problem P1:

$$\arg\max \sum_{n\in[1,N]} \mathcal{Q}oE_n$$

EDGE360's aim for problem **P1** is to determine the optimum bitrate and tile quality selection approach with optimal policy $\pi^* : S \cdot A \to [0, 1]$, that maximizes the projected long-term discounted QoE while taking into consideration the limitations of view-port and tiled-based 360° video streaming in multi-client scenarios.

A. Foundation of MDP

A discrete-time finite Markov Decision Process (MDP) represented by $\mathcal{M}$, is defined by a 7-tuple $(\mathcal{S}, \{\mathcal{O}_n\}_{n\in\mathcal{N}}, \{\mathcal{A}_n\}_{n\in\mathcal{N}}, f, \{r_n\}_{n\in\mathcal{N}}, \gamma, \rho_0, H)$, which serves as the foundation for modeling decision-making in dynamic environments where multiple agents engage with both the environment and each other. Here, $\mathcal{N}$ represents the set of agents, and each agent makes individual decisions while interacting with a shared environment. This formulation indicates the action space $\mathcal{A}_n$ of agent n and the state space of the environment by $\mathcal{S}$ as they encompass both global and local information that influences decision-making. However, in multi-agent systems each agent usually have access to partial observations of this state space which is denoted by $\{\mathcal{O}_n\}_{n\in\mathcal{N}}$, where $\mathcal{O}_n$ represents the specific set of observations available to agent n. Such partial accessibility translates the MDP into a Partially Observable Markov Decision Process (POMDP), which reflects the actual constraints that agents encounter in real-world systems. $\mathcal{A}_n = \{a_n^v, a_n^m, a_n^b\}$ represents the set of actions the agent n can take and a_n^v, a_n^m, a_n^b are actions for viewport, marginal, and background tiles, respectively. The transition dynamics are defined by the deterministic function $f : \mathcal{S} \times A \to S$, which specifies the next state s' to which the environment will transition when an action a is performed in the current state s. The f accounts for joint actions of all agents (clients) and actions that affect the global state, especially regarding network throughput, buffer states, and tile quality switches. The reward for taking an action a in state s is given by the bounded reward function $r : S \times A \to \mathbb{R}$. It incorporates the discount factor $\gamma (0 \leq \gamma \leq 1)$ to balance the importance of current and future rewards. The initial state distribution can be determined using the formula $\rho_0 : S \to \mathbb{R}^+$, it reflects the initial conditions of network throughput, buffer states, and video quality across all clients, where H represents the horizon length or the length of the decision-making trajectory. Furthermore, since each EDGE360 agent perceives a partial view of the global state $\mathcal{S}$, the framework is a POMDP [34]. COMA's centralized critic calculates counterfactual baselines to influence individual agents' actor strategies, maximizing their value to the shared global reward. The objective of the POMDP framework for the agent n is to find the optimal policy $\pi_n^* : S \to A$, that maximizes the cumulative long-term discounted reward. This can mathematically be represented as $R = \sum_{t=0}^{H} \gamma^t \cdot r_t$, where r_t is the reward at time t and γ^t is

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

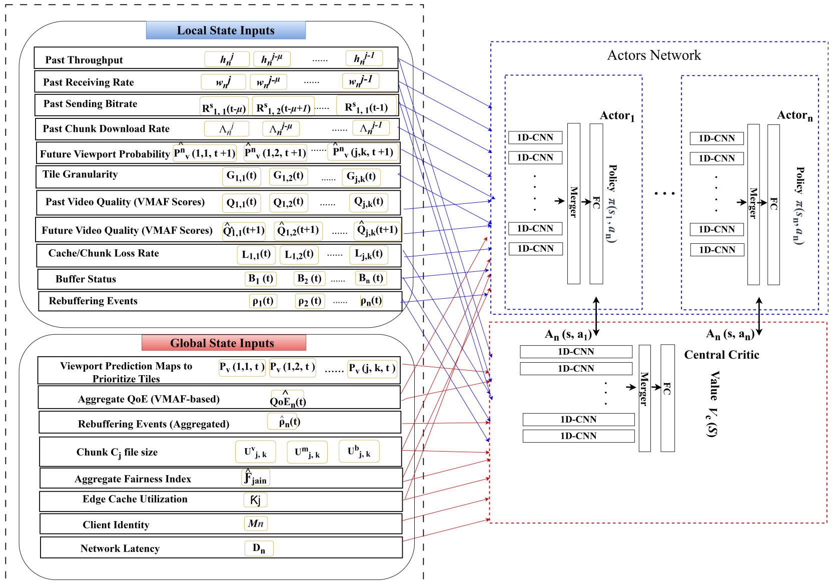


Fig. 3: Neural Network Architecture for EDGE360 Adaptive 360° VR Streaming Using COMA

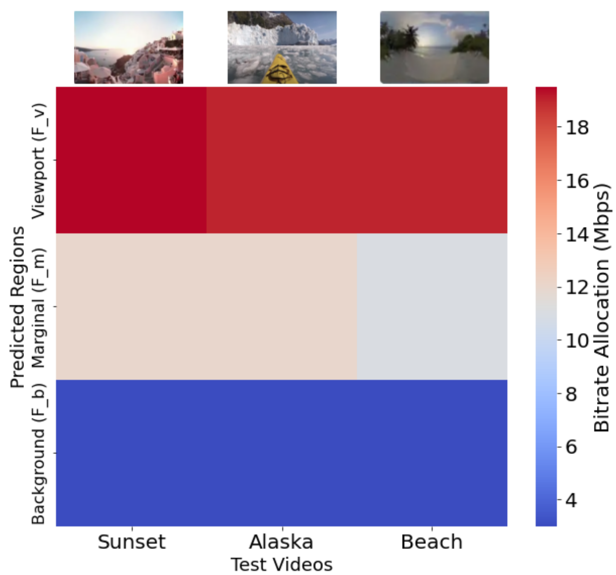


Fig. 4: EDGE360's adaptive bitrate distribution among predicted regions (viewport, marginal, and background) for three test videos.

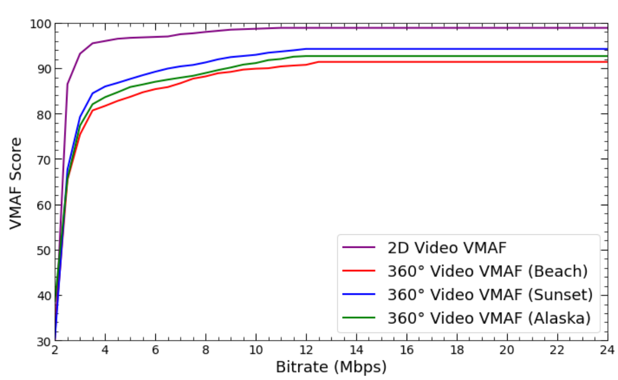


Fig. 5: Bitrate Vs VMAF of 2D Video and 360° videos (*Sunset*, *Alaska*, and *Beach*)

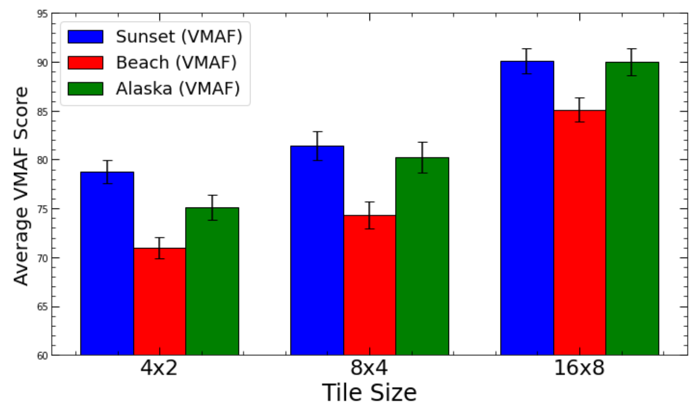


Fig. 6: Average VMAF score Vs Tile size for *Sunset*, *Alaska*, and *Beach* 360° Videos

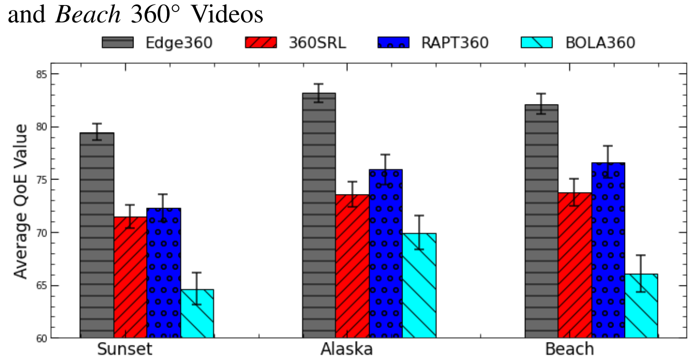


Fig. 7: Comparison of average chunk QoE values for different adaptive streaming strategies across three test videos, with error bars representing ± one standard deviation.

a discount factor that balances the significance of immediate versus future rewards. Consequently, the edge-based optimal bitrate selection problem can be redesigned as:
Problem **P2:**

$$\max_{\{\pi_n\}_{n\in\mathcal{N}}} \mathbb{E}\left[\sum_{t=0}^{H} \gamma^t \cdot R_t \mid \{\pi_n\}_{n\in\mathcal{N}}\right] \tag{4}$$

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

Algorithm 1: Part 1: Input/Output Processing and Hierarchical Tile Selection with Bitrate Allocation

Input : Predicted Head Rotation $\mathcal{F}$, dividing the frame into three regions:
$\mathcal{F}_v$: Viewport Region (high priority),
$\mathcal{F}_m$: Marginal Region (moderate priority),
$\mathcal{F}_b$: Background Region (low priority),
Network Throughput $W_n(t)$, Buffer Status $B_n(t)$, Set of Available Bitrates $\mathcal{R} = \{R_1, R_2, \dots, R_M\}$, and Neural Network Inputs.
Output: Tile set $T_{j,k} \in T = \{T^v, T^m, T^b\}$ with optimized bitrates $\mathcal{R}^{\chi}_{j,k}, \forall \chi \in \{v, m, b\}$.

Initialization: $T^v = \emptyset$, $T^m = \emptyset$, $T^b = \emptyset$.;

Hierarchical Tile Selection with Bitrate Allocation:
foreach *tile* $T_{j,k}$ **do**
  if $T_{j,k} \cap \mathcal{F}_v \neq \emptyset$ **then**
    $T_{j,k} \mapsto T^v, \quad \mathcal{R}^v_{j,k} \leftarrow R_v \in \mathcal{R}$;
  end if
  if $T_{j,k} \cap \mathcal{F}_m \neq \emptyset$ **then**
    if $\mathcal{G}_{j,k}(t) = a$ **then**
      $T_{j,k} \mapsto T^m, \quad \mathcal{R}^m_{j,k} \leftarrow R_m \in \mathcal{R}$;
    end if
    if $\mathcal{G}_{j,k}(t) = c$ **then**
      Subdivide$(T^b_{j,k}) \mapsto T^m_{j,k}$;
    end if
  end if
  if $T_{j,k} \cap \mathcal{F}_b \neq \emptyset$ **then**
    if $\mathcal{G}_{j,k}(t) = c$ **then**
      Subdivide$(T^b_{j,k}) \mapsto T^m_{j,k}$;
    end if
    if $\mathcal{G}_{j,k}(t) = a$ **then**
      Subdivide$(T^m_{j,k}) \mapsto T^v_{j,k}$;
    end if
    $T_{j,k} \mapsto T^b, \quad \mathcal{R}^b_{j,k} \leftarrow R_b \in \mathcal{R}$;
  end if
end foreach

Tile Merging for Compression:
if *all fine-grained viewport tiles* $T_{j,k} \in T^v$ *within a region belong to the same subset* **then**
  Merge $T_{j,k} \in T^v$ into a medium-grained tile $T^m_{j,k}$.;
end if
else if *all medium-grained marginal tiles* $T_{j,k} \in T^m$ *within a region belong to the same subset* **then**
  Merge $T_{j,k} \in T^m$ into a background tile $T^b_{j,k}$.;
end if
else
  Maintain the current tiling structure.;
end if

Algorithm 1: Part 2: State Preparation, EDGE360 Evaluation, and Cache Management

State Preparation for EDGE360:
Local States: Throughput history $W_n(t)$ (Eq. A.1),
VMAF scores $V_{j,k}(t)$ with tile granularity
$G_{j,k}(t)$ (Eq. A.5),
Quality transitions $\Delta\mathcal{Q}_{j,k}(t)$ (Eq. A.13),
Buffer status $B_n(t)$ (Eq. **??**).
Global States: Viewport maps $P_v(j,k,t)$ (Eq. A.4),
aggregated QoE $\mathcal{Q}o\bar{E}_n(t)$ (Eq. 1), rebuffering events
$\rho_n(t)$ (Eq. A.15), Latency $D_n(t)$ (Eq. A.17)

EDGE360 Evaluation:
Combine local and global states in the central critic to evaluate $Q(s,a)$ (Eq. 8).;
Each actor network selects actions (tile bitrates $\mathcal{R}^{\chi}_{j,k}$) for T^v, T^m, T^b to maximize QoE.;

Dynamic Bitrate Adaptation:
if $W_n(t) < W_{th,min}$, *where* ΔR: *bitrate reduction step for non-viewport tiles* **then**
  if $\mathcal{R}^b_{j,k} > R_{min}$ **then**
    Reduce bitrate: $\mathcal{R}^b_{j,k} = \mathcal{R}^b_{j,k} - \Delta R$;
  end if
  if $\mathcal{R}^m_{j,k} > R_{min}$ **then**
    Reduce bitrate: $\mathcal{R}^m_{j,k} = \mathcal{R}^m_{j,k} - \Delta R$;
  end if
  if $\mathcal{R}^v_{j,k} < R_{max}$ **then**
    Maintain highest bitrate: $\mathcal{R}^v_{j,k} = R_{\max}$;
  end if
end if

Edge Cache Integration:
foreach $T_{j,k} \in T = \{T^v, T^m, T^b\}$ **do**
  if $T_{j,k}$ *is available in the edge cache* **then**
    Retrieve $T_{j,k}$ with its assigned quality and corresponding data size $U_{j,k}$ (Eq. A.8);
  end if
  else
    Request $T_{j,k}$ from the video server based on its region:
    - **Viewport** (T^v): High quality, data size $U^v_{j,k}$ (Eq. A.8)
    - **Marginal** (T^m): Moderate quality, data size $U^m_{j,k}$ (Eq. A.8).
    - **Background** (T^b): Low quality, data size $U^b_{j,k}$ (Eq. A.8).

    Store the retrieved $T_{j,k}$ in the edge cache.;
  end if
end foreach
return $T = \{T^v, T^m, T^b\}$ *with bitrates* $\mathcal{R}^{\chi}_{j,k}$.

The P2 goal is to collaboratively optimize the policies of all agents in the EDGE360 framework to maximize the predicted cumulative discounted global reward across all clients in a multi-agent scenario.

B. State Formulation for EDGE360 Adaptive 360° Streaming

In the multi-client EDGE360 architecture, the MDP is modified into a Markov game [35] to handle interactions among multiple agents and the environment. Each client behaves as an independent actor, requesting video tiles, and their collective decisions impact global state changes. The complicated dynamics of adaptive 360° video streaming, such as viewport prediction, tile-based adaptation, and limited network resources, require a precise state definition. The state of a client n requesting a video chunk C_i is defined as follows:

$$s^n_j = \left\{h^n_j, \omega^n_j, \Lambda^n_j, u^n_{j,k}, d^n_{j,k}, \mathcal{V}^n_{j-1}, B^n_j, \Delta\mathcal{Q}^n_{j,k}, P^n_v(j,k,t)\right\} \tag{5}$$

This state represents numerous facets of the client's interaction with the system. In a multi-client scenario, each client's local state s^n_j serves the joint global state $S_j = \{s^1_j, s^2_j, \dots, s^N_j\}$. COMA's centralized critic utilizes S_j to optimize resource allocation, guarantee fairness, and maximize the client's QoE. The historical network throughput, presented as $h^j_n = \{h^{j-\mu}_n, \dots, h^{j-1}_n\}$, incorporates the average throughput measurements over the past μ chunks, representing recent network performance to assist bitrate adaptation. Similarly, $\omega^j_n = \{\omega^{j-\mu}_n, \dots, \omega^{j-1}_n\}$ contains the downloading instances of the previous μ chunks, providing insights regarding transmission efficiency and network variability. The historical data retrieval rate is expressed as $\Lambda^n_j = \{\Lambda^n_{j-\mu}, \dots, \Lambda^n_{j-1}\}$, indicating the pace at which prior chunks were downloaded.

This metric helps to monitor network consistency and guide bitrate adaptation, resulting in smooth playback while

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

preventing unnecessary rebuffering. The required chunk C_i tile sizes are represented as $u^n_{j,k} = [u^v_{j,k}, u^m_{j,k}, u^b_{j,k}]^T$, where $u^v_{j,k}$, $u^m_{j,k}$, and $u^b_{j,k}$ correspond to the viewport, marginal, and background regions, respectively. Each tile size is set for multiple bitrate levels, such as $u^v_{j,k} = \{u^v_{j,k}(R_1), \ldots, u^v_{j,k}(R_M)\}$, which enables the solution to change quality granularity swiftly. Similarly, $d^n_{j,k} = [d^v_{j,k}, d^m_{j,k}, d^b_{j,k}]^T$ records spherical distortions associated with these tiles, indicating spatial quality drop between regions.

The past chunk's video quality denoted as V_n^{j-1}, serves as a reference point to ensure quality consistency. The buffer occupancy metric B_n^j assesses the client's playback buffer during a chunk request C_i and is crucial for smooth playing and avoiding re-buffering occurrences. The quality switching penalty $\Delta\mathcal{Q}^n_{j,k}$ adjusts for disruptive quality variations, resulting in seamless transitions between video quality levels. The viewport probability mapping model $P^v_n(j,k,t)$ anticipates the likelihood of tile (j,k) being viewed based on the client's viewport trajectory, allowing for prioritized resource allocation.

C. Actor-Central Critic Policy Training for EDGE360

In the EDGE360 architecture, the actor and central critic network leverage the COMA approach to handle the challenges of multi-client adaptive streaming in a partially observable setting. The technique merges client-specific decisions with global optimization via a POMDP framework, resulting in an effective resource allocation and improved QoE for all rich users.

1) Actor Policy Approximation: The actor policy for each client is parameterized by φ_n, representing the weights of the actor network. This network produces a probability distribution over possible actions, such as bitrate allocation and tile quality selection. The distribution is computed using a softmax function:

$$\pi_n(a_n \mid s) = \text{Softmax}(\phi(s; \varphi_n)) \tag{6}$$

where, $\phi(\cdot)$ is the feature extraction function that transforms the observable state s into a latent representation, while φ_n denotes the trainable parameters of the actor network for client n. This allows the policy to dynamically adapt to client-specific requirements, such as viewport movement, network fluctuations, and buffer conditions, while being influenced by the shared global environment. global environment.

2) Central Critic State-Value Function: The state-value function which determines the predicted cumulative reward setting up from a particular global state s is assessed by the centralized critic. It is represented as:

$$V_c(s) = \mathbb{E}\left[\sum_{t=0}^{H} \gamma^t R(s,a)\right], \tag{7}$$

where, $R(s,a)$ is the global reward aggregating QoE tracks across all clients, γ is the discount factor harmonising current and future rewards, and H is the decision horizon. This function provides a global view by taking into account all agents' combined states and actions, coordinating their actions with system objectives.

3) Advantage Function for Central Critic: In order to enhance actor training, the central critic evaluates the advantage function, which compares the effectiveness of an agent's action to a counterfactual baseline. This is stated as follows:

$$A_n(s,a_n) = Q(s,a) - \sum_{n=1}^{N} \pi_n(a_n \mid s) Q(s, a^{-n}) \tag{8}$$

Here, a denotes the joint action vector of all agents, i.e., $a = (a_1, a_2, \ldots, a_N)$, while a^{-n} represents the set of actions taken by all agents except agent n. The state-action value function is represented by $Q(s,a)$, and the counterfactual baseline by $Q(s,a^{-n})$ which minimizes the contribution of agent n by marginalizing over its actions. The counterfactual baseline $Q(s,a^{-n})$ estimates the global reward by marginalizing agent n's action while keeping other agents' actions fixed. This formulation is crucial for EDGE360 with multiple agents, as the advantage function enables each agent to focus on optimizing its unique contribution to the global reward, guaranteeing fairness and efficient resource allocation for each client.

4) Actor Policy Gradient Update: In EDGE360, the actor policy gradient update is critical for enhancing each client's actor network. The policy gradient approach updates the actor networks, which promotes actions that result in higher expected rewards. The actor policy gradient update can be expressed as:

$$\nabla_{\varphi_n} J(\varphi_n) = \mathbb{E}\left[A_n(s,a_n) \nabla_{\varphi_n} \log \pi_n(a_n \mid s)\right] \tag{9}$$

Here, function $A_n(s,a_n)$ measures the comparative advantage of an action compared to its baseline. This allows the actor network to maximize its distinct value to the shared global reward. This recurrent refinement guarantees that the policy responds dynamically to variations in network circumstances, such as bandwidth fluctuations and viewport trajectories, while also optimizing the QoE for each client. This strategy combines local decision-making with global system objectives by using COMA's multi-agent framework.

5) Entropy Regularization: to stimulate exploration while avoiding early convergent to suboptimal policies, the actor objective incorporates entropy regularisation as follows:

$$H(\pi_n) = -\sum_{a_n} \pi_n(a_n \mid s) \log \pi_n(a_n \mid s) \tag{10}$$

where, $H(\pi_n)$ measures the randomness of the policy's action decision. A higher entropy number implies a more exploratory perspective, whereas a lower entropy indicates a more predictable behavior. Entropy regularization encourages diversity in action choices, ensuring that agents explore different bitrate allocation and tile quality ways across the viewport, marginal, and non-viewport regions. The updated actor objective, which includes entropy regularization becomes:

$$\nabla_{\varphi_n} J'(\varphi_n) = \mathbb{E}\left[\nabla_{\varphi_n}\left(A_n(s,a_n) \log \pi_n(a_n \mid s) + \eta H(\pi_n)\right)\right] \tag{11}$$

The updated actor objective gradient, denoted by $\nabla_{\varphi_n} J'(\varphi_n)$, incorporates an entropy regularization term scaled by the tunable parameter η, which balances exploration and exploitation. This regularized formulation is especially

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

important in EDGE360, where agents must adapt to dynamic network states and diverse viewport trajectories in a multi-client, tile-based streaming environment.

6) Critic Network Loss Function: The critic network is trained to reduce temporal difference (TD) errors, which improves the accuracy of value assessments. The loss function is stated as follows:

$$L(\Psi) = \mathbb{E}\left[\left(V_c(s) - (r + \gamma V_c(s'))\right)^2\right], \tag{12}$$

Here, $V_c(s)$ is the predicted value of the current state s, r is the immediate reward obtained following action, and $V_c(s')$ is the estimated value of the next state s'. The discount factor (γ) determines the relevance of future rewards compared to immediate ones. By minimizing this loss function, the critic network learns to deliver precise value estimations, allowing the centralized critic in EDGE360's COMA system to develop effective counterfactual advantage functions.

EDGE360's actor central critic system integrates these components, i.e. $\pi_n(a_n \mid s)$, $V_c(s)$, $A_n(s, a_n)$, $\nabla_{\varphi_n} J(\varphi_n)$, $H(\pi_n)$, $\nabla_{\varphi_n} J'(\varphi_n)$, $L(\Psi)$, aligning client-specific edge-based adaptations with global QoE optimization. The COMA-based centralized critic ensures fairness, while actor networks dynamically adapt to local conditions, allowing for efficient and adaptable 360° video streaming in multi-agent settings.

VI. Implementation and Evaluation

A. Simulation Settings

To evaluate the performance of the proposed EDGE360 algorithm, we built a Python-based simulator with the PyTorch framework [36]. The simulator was built on an Ubuntu 24.04 machine with a 64-bit Intel Core i7-8550 U CPU (1.8 GHz quad-core, turbo boost up to 4GHz) and 16 GB RAM.

EDGE360's COMA DRL-based rate adaptation technique employs a single centralized critic network as well as several independent actor networks, with each client in a multi-client streaming situation acting as an autonomous agent. Unlike independent DRL approaches like A3C and DDQN, which rely on decentralized learning, COMA combines centralized training and decentralized execution to provide optimum collaborative policy acquisition for viewport-based bitrate selection while remaining fair between different clients. To simulate realistic multi-client streaming scenarios aligned with 5G network conditions, EDGE360 is evaluated with increasing number of clients: 16, 32, and 64 concurrent users, respectively. Each actor network is mapped to a unique client and independently performs tile-wise bitrate allocation decisions. The architecture supports decentralized execution, allowing each actor to operate with local observations that include viewport probability maps, buffer states, and past bitrate actions. Simultaneously, the client's observation space inherently captures the viewport, marginal, and background regions, enabling fine-grained control over spatial adaptation. This region-aware, modular structure not only supports accurate per-user decisions but also facilitates scalable operation, as validated in our scalability experiments. Even with 64 clients, EDGE360 maintains acceptable levels of QoE, fairness, and rebuffering, demonstrating its robustness in high-density streaming environments.

Each network of client-specific actors selects bitrates based on local information such as viewport probability visualizations, network performance, buffer occupancy, and history bitrate decisions, where the actor policy gradient update $\nabla_{\varphi_n} J(\varphi_n)$ Eq. (9) optimizes policy learning by adjusting actions that result in higher expected rewards. These actor networks consist of an input layer and two fully connected (FC) layers, with the first layer comprising 128 neurons and the second containing 64 neurons, both with ReLU activation. The output layer implements a Softmax activation function $\phi(s; \varphi_n)$ Eq. (6) to generate action probabilities that decide the bitrate of each client's tiles in the viewport, marginal, and background regions. The critic examines the state action value function $Q(s, a)$ and compares the impact of each agent's action with a counterfactual baseline $Q(s, a^{-n})$ using an advantage function $A_n(s, a_n)$ Eq. (8) rather than directly calculating a Q value in a linear activation output layer. The counterfactual baseline marginalizes the contribution of the agent n's to global reward by marginalizing its activities. Taking into account future rewards over a broader horizon, the discount factor set to $\gamma = 0.99$ ensures long-term optimization. Using the Adam optimizer for effective convergence, the learning rate is set at 1×10^{-4} for actor networks and 5×10^{-3} for critic networks. To avoid early convergence to suboptimal policies, the entropy regularisation factor $H(\pi_n)$ Eq. (10) is initialized at 0.99 and decays exponentially at a rate of 0.995 every 10^3 iterations in the COMA DRL framework for exploration.

B. Viewport Dataset

To model the user viewport dynamics in EDGE360, we used a publicly available viewport dataset that includes head rotation traces from 153 volunteers while watching various 360° videos [37]. These traces, sampled at 10 Hz (i.e., 10 samples per second), offer high temporal resolution to support fine-grained head movement prediction.

EDGE360 adopts a LSTM-based deep learning model for viewport prediction due to its ability to learn long-range temporal dependencies inherent in user head motion. In contrast to CNN-based methods that often struggle with temporal modeling [38], the LSTM network captures sequential user interactions effectively and enables forecasting of the future viewport center using historical rotation data. Additionally, EDGE360 integrates a Laplace-based probabilistic viewport expansion scheme, which estimates tile-wise probability around the predicted center, exponentially decaying with distance. This allows for perceptually guided bitrate allocation, favoring high-probability tiles.

Although attention-based and transformer-based architectures have recently shown strong predictive capabilities in such domains [39], however, LSTM was chosen for its balance of accuracy and computational efficiency in real-time applications. Transformers, while potentially more accurate, often involve higher inference latency and parameter overhead, making them less practical for edge-based scenarios with resource constraints.

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

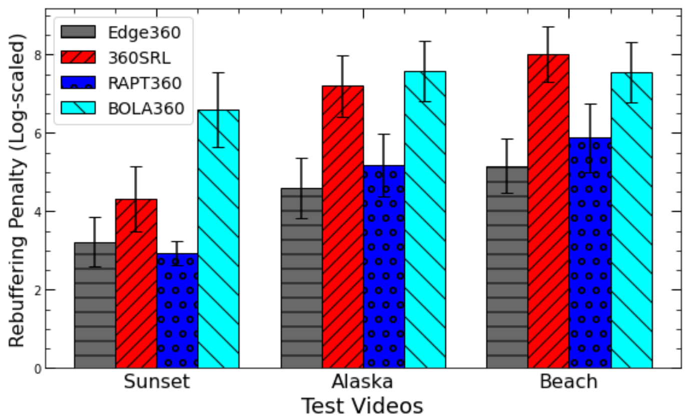


Fig. 8: Log-scaled rebuffering penalty for three test videos (*Sunset*, *Alaska*, and *Beach*) for different adaptive streaming techniques (EDGE360, 360SRL, RAPT360, and BOLA360), with error bars representing ± one standard deviation.

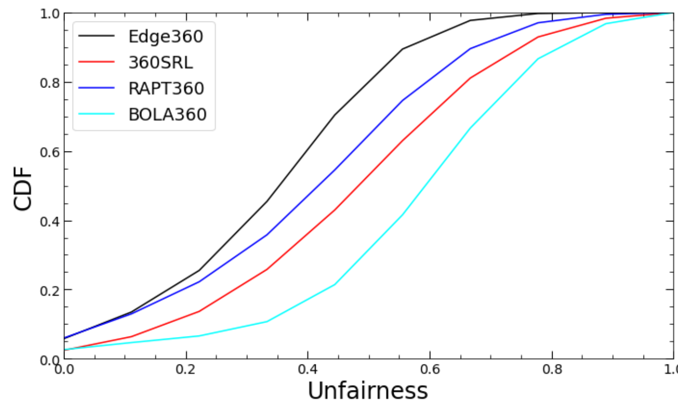


Fig. 9: CDF vs. unfairness for different streaming strategies ensuring balanced resource allocation across clients.

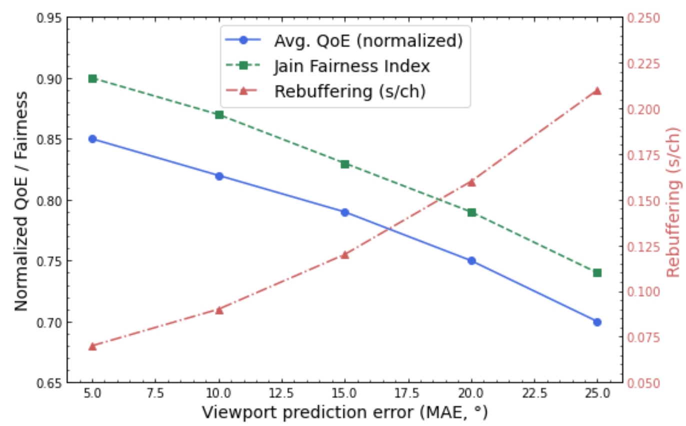


Fig. 10: Sensitivity analysis of EDGE360 for viewport prediction error, showing the impact on normalized QoE, Jain's fairness index, and average rebuffering time per chunk.

The prediction model was quantitatively evaluated using the Mean Absolute Error (MAE) and viewport hit rate metrics. The average MAE across all test users was 5.0°, and the average viewport hit rate exceeded 90%, indicating strong predictive accuracy. These metrics validate the use of the LSTM model in combination with the Laplace-based expansion strategy employed in EDGE360.

Furthermore, Fig. 10 quantifies the system's sensitivity to viewport prediction error through an ablation study. To simulate degraded prediction accuracy, we injected controlled Gaussian noise into the predicted viewport coordinates and evaluated the resulting impact on EDGE360 performance. The LSTM-based viewport predictor used in our framework achieved a MAE of 5.1°, a hit rate within 5° of 89%, and an F1-score of 0.92 for FoV classification. As the injected MAE increases from 5° to 25°, the normalized average QoE steadily declines from 0.85 to 0.70, indicating an 18% loss in perceptual quality due to misallocated bitrate and increased stall events. Jain's fairness index similarly falls from 0.90 to 0.74, demonstrating imbalanced quality across users, as the COMA reward function dynamically reallocates bandwidth to clients with poor predictions via the deviation term $|\varepsilon|$. Rebuffering emerges as the dominant penalty beyond 15° MAE, where stall time per chunk increases from 0.07 s to 0.21 s, surpassing the penalty from quality switches.

This comprehensive degradation across all three dimensions QoE, fairness, and rebuffering, highlights the pivotal role of accurate, low-latency viewport prediction. EDGE360 maintains high QoE and fairness as long as the prediction MAE remains below 20°, validating the choice of an efficient LSTM + Laplace PDF model over heavier attention-based alternatives in real-time VR scenarios.

C. Rich Video Dataset and Network Traces

To ensure robustness across diverse 360° content, we employ the public dataset [40], comprising videos of varying durations (37–668s) and spatio-temporal complexity. The COMA-MADRL model was trained on the full dataset. For evaluation, we selected three representative videos *Beach*, *Sunset*, and *Alaska* and exhibiting varying degrees of motion, texture, and scene complexity corresponding to low, average, and high VMAF profiles, covering a broad range of motion and visual detail to validate EDGE360's adaptability. Each video presents unique challenges: *Beach* features rapid motion and dynamic transitions, *Sunset* contains static-to-moderate scene shifts, and *Alaska* captures high spatial fidelity with minimal movement. This selection ensures that performance is tested across a diverse set of real-world conditions, validating the robustness of EDGE360's adaptation strategy. To provide adaptive 360° video streaming, we deploy the DASH360 tool, which divides each video into spatially adaptable tiled representations while optimizing bandwidth allocation based on viewport likelihood. DASH360 separates videos into three levels of tile granularity: coarse-grained (1024 × 1024 pixels), medium-grained (512 × 512 pixels), and fine-grained (256 × 256 pixels) to ensure effective bitrate distribution across regions. After tiling, the videos are encoded in HEVC/H.265 employing FFmpeg [41] and transformed into 2 seconds MPEG-DASH chunks with MP4Box [42]. Tile of each granularity is then pre-transcoded into VMAF-optimized bitrates based on its probability of being viewed, ensuring perceptual quality is maintained in high-priority viewing regions. The encoded content comprises 11 unique bitrates, i.e., 2.5, 4.0,

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

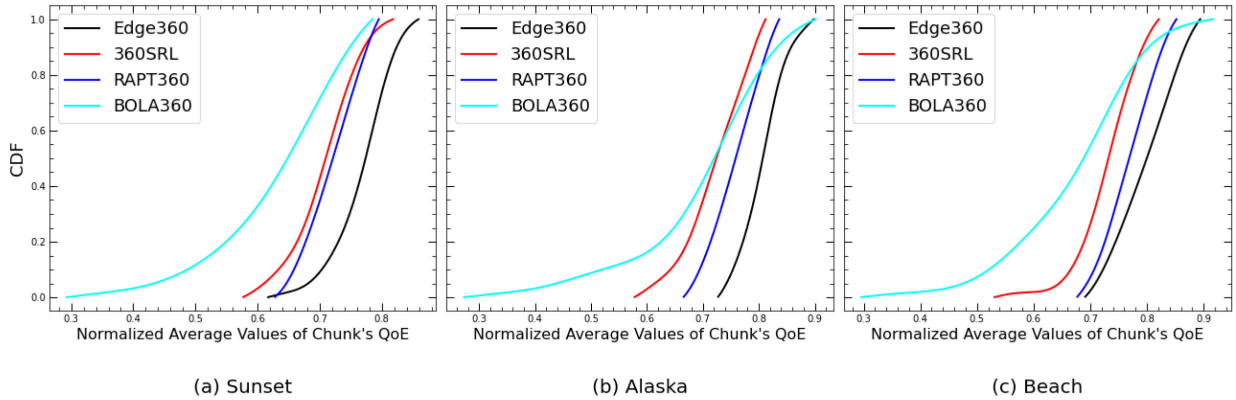


Fig. 11: Comparison of the CDF vs. normalized average chunk QoE for sessions simulated on three test videos employing different adaptive streaming strategies.

5.0, 8.0, 10.0, 12.5, 15.0, 17.5, 20.0, 22.5, 25.0 Mbps, which essentially aligns with the 5.0-20.0 Mbps range proposed by 3GPP for 360° video streaming [43]. To measure perceptual video quality, we use the VMAF-360° models [44]. All three test videos, *Sunset*, *Alaska*, and *Beach*, have a resolution of 4096 × 2160 (4K) and a duration of 89, 234, and 138 seconds, respectively.

We employ a public 5G network trace dataset [45] gathered from operating 5G networks, mimicking different degrees of bandwidth fluctuation, to assess adaptive multi-agent 360° video streaming under realistic network conditions. These traces show three main network modes: disturbed (660 Mbps, moderate variations), unstable (585 Mbps, large fluctuations), and stable (734 Mbps, minimum changes). Each client in the EDGE360 multi-agent architecture functions as a distinct RL agent, using real-time throughput observations to make adaptive decisions regarding bitrate selection and viewport tile prioritization. The 5G traces function as a shared network environment that influences the fairness of bandwidth distribution among multiple clients and dynamically influences the agents' adaptation approaches. Fine-grained viewport tiles are guaranteed high bitrates (14–20 Mbps) through collaborative decision-making, whereas background (2.5–8 Mbps) and marginal (8–12 Mbps) tiles are resource-efficiently assigned lower bitrates. EDGE360 uses 5G transmission to reduce rebuffering, improve multi-user QoE, and dynamically adjust to fluctuating network circumstances.

D. Results and Analysis

This section compares EDGE360 with other adaptive streaming techniques and thoroughly studies EDGE360's performance across various assessment measures. Evaluations include VMAF scores for different bitrate levels and tile sizes, chunk-wise QoE switches, rebuffering penalties, resource allocation fairness, and training convergence trends. This shows how EDGE360's optimization architecture impacts adaptive learning, rebuffering, and video quality switches, demonstrating its ability to achieve high QoE and network efficiency across various 360° video content.

Fig. 5 shows the relationship between bitrate (Mbps) and VMAF scores for both 2D and 360° video content in three test videos (*Sunset*, *Alaska*, and *Beach*). The 2D video shows a clear trend, with rising bitrate resulting in an increasing rise in VMAF scores, implying improved video quality. This trend is predicted since greater bitrates improve encoding efficiency, reduce compression artefacts, and improve perceived quality. Notwithstanding the trend of rising bitrate, 360° videos continually have lower VMAF scores than 2D videos. This disparity comes from the intrinsic complexity of 360° details, which requires substantially more data to reach the same visual quality as 2D video.

The three 360° test videos exhibit different VMAF trends across bitrate values due to their distinct spatial and temporal complexity. *Sunset*, with the highest spatial complexity content, requires a higher bitrate to maintain a good VMAF score, placing it ahead of *Beach* and *Alaska* at higher bitrates owing to rich visuals that involve more encoding resources. *Alaska*, with the highest temporal complexity, imposes challenges to adaptation since fast motion demands precise viewport prediction for persistent video quality. *Beach*, with low temporal and moderate spatial complexity, has a smoother quality trend but falls behind *Sunset* in VMAF scores. The compression inefficiencies of 360° videos, attributed to vast FoV and diverse environments, underline the need for proficient adaptive streaming such as EDGE360 that optimizes quality variation dynamically while minimizing rebuffering.

Fig. 6 illustrates the relationship between tile size and average VMAF score for the three 360° test videos: *Sunset*, *Alaska*, and *Beach*. A rise in tile granularity from 4×2 to 16×8 improves VMAF scores for all videos, demonstrating the advantages of finer-grained tiling for higher video quality. *Sunset* typically gets the best VMAF ratings because of its high spatial complexity and moderate temporal complexity, which benefits from higher bitrate allocation at finer tile resolutions. Despite its higher temporal and moderate spatial complexity, *Alaska* follows a similar pattern, although with a bit lower scores due to motion-induced compression inefficiencies. *Beach*, with moderate spatial complexity and little temporal variation, has the lowest VMAF growth but still benefits from finer tile segmentation. The error bars demonstrate that quality fluctuates and narrows as tile resolutions grow, indicating improved consistency at higher granularity.

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

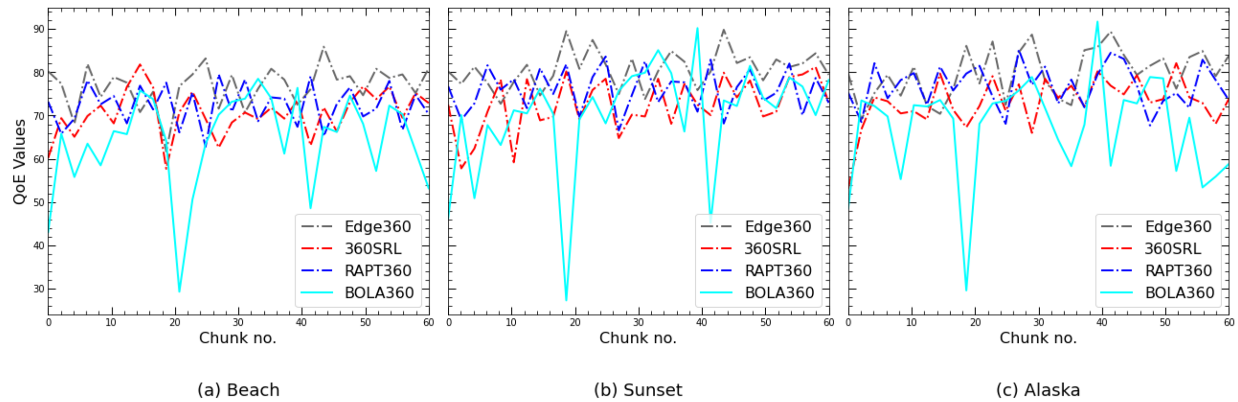


Fig. 12: Variations in chunk-wise QoE values for adaptive streaming strategies, randomly selected from three test video session simulations.

Employing adaptive streaming optimization algorithms, the graph properly represents the impact of tile size on perceived video quality.

Fig. 7 demonstrates that EDGE360 outperforms other tile-aware viewport-based adaptive streaming strategies by consistently achieving the greatest QoE across all test videos. The greatest improvement over BOLA360 is shown with EDGE360, which offers greater QoE for *Sunset*, *Alaska*, and *Beach* by 22.97%, 18.94%, and 24.33%, respectively. Lower QoE results from BOLA360's buffer-based adaptation's inability to effectively optimize viewport-aware streaming. EDGE360's QoE for *Sunset*, *Alaska*, and *Beach* is 11.16%, 13.03%, and 11.34% higher than 360SRL, respectively, demonstrating its resilience when handling high-motion and spatially diverse video content. Against RAPT360, EDGE360 stays ahead, particularly in *Beach* (7.18%), *Alaska* (9.50%), and *Sunset* (9.90%), demonstrating its efficiency in adapting to varying video characteristics, dynamic viewport trajectories, and network conditions. EDGE360's improved performance is primarily due to its adaptive viewport-aware tiling approach, which allocates bandwidth appropriately depending on viewport prediction and content complexity.

EDGE360 dynamically prioritizes tiles in the user's viewport, guaranteeing efficient bitrate allocation while retaining excellent perceptual quality. This is particularly pertinent for videos like *Sunset*, which have high spatial complexity and require high bitrates. *Alaska* also has high temporal complexity and requires swift adaptation to motion changes. EDGE360 provides an improved balanced trade-off between quality enhancement and rebuffering minimization by utilizing the MARL approach, resulting in considerably enhanced QoE across different test videos compared to state-of-the-art solutions. To reinforce generalizability and comprehensiveness, we extend our evaluation to include nine 360° videos with diverse spatial-temporal characteristics (see Appendix B, Fig. B.1) and assess performance under three real-world 5G network conditions: stable, disturbed, and unstable (see Appendix B, Fig. B.2). These extended test video results demonstrate the robustness of EDGE360 across content diversity and dynamic bandwidth fluctuations.

Fig. 8 presents the log-scaled rebuffering penalty for EDGE360, 360SRL, RAPT360, and BOLA360 over three test videos. EDGE360 consistently attains the lowest rebuffering penalties by utilising predictive adaptation based on COMA reinforcement learning, which allows for efficient bitrate allocation and viewport-aware tile prioritization. In contrast, BOLA360 suffers the most penalty, notably in *Sunset* and *Alaska*, where its entirely buffer-based adaptability struggles with significant spatial and temporal complexity, resulting in frequent rebuffering occurrences. It is worth noting that reducing rebuffering does not always result in higher video quality. Some techniques, such as RAPT360, provide lower rebuffering penalties, especially in *Sunset*, but struggle to adequately optimize perceptual quality, whereas 360SRL performs ineffectively concerning rebuffering reduction and video quality enhancement. EDGE360 finds a balance by minimizing rebuffering while maintaining excellent visual quality via MARL-based decision-making that adjusts to both network fluctuations and viewport variations. In *Beach*, wherein temporal complexity is lower, each approach faces less cost, but BOLA360 continues to perform poorly owing to its reactive nature. EDGE360's ability to strike intelligent trade-offs between improvement in quality and rebuffering minimization makes it a more effective adaptive 360° streaming solution in terms of delivering a seamless viewing experience.

Fig. 9 demonstrates EDGE360's improved fairness in resource allocation, with the curve hitting higher values at lower unfairness levels. This displays its MARL-based optimization, which balances QoE across clients. 360SRL and RAPT360 are moderately fair, while BOLA360 is the most unfair owing to its buffer-based adaptation, which prioritizes rebuffering avoidance over fair resource allocation. These findings are consistent with Jain's Fairness Index (Eq. 2) concepts presented in the above section. EDGE360 significantly reduces QoE disparities, guaranteeing consistent and proportionate quality adaptation across each client. EDGE360 (COMA) outperforms RAPT360 (A3C) and 360SRL (DDQN) by employing a centralized critic to ensure enhanced fairness and effective utilization of resources among clients. Dynamically optimizing viewport-based tile-aware streaming avoids severe QoE shifts, making it excellent for multi-client adaptive streaming use cases. To precisely explain the difference in performance, we present the cumulative distribution function (CDF) vs. normalized average chunk QoE curves in Fig. 11.

The CDF curves for EDGE360 proceed more to the right than other techniques, showing that a more significant proportion of session chunks retain high QoE levels. In contrast, 360SRL and RAPT360 have intermediate QoE distributions, with slopes ranging between EDGE360 and BOLA360. This implies that while these strategies can attain reasonable QoE levels, their learning-based adaptations underperform in optimising quality under evolving network and content scenarios. BOLA360 has the steepest and most leftward-shifted CDF curves, indicating an inability to maintain high QoE. These results support EDGE360's COMA-based MARL optimisation, which improves viewport-aware adaptation and bitrate allocation, providing consistent, high-quality streaming across a range of video characteristics.

Fig. 12 shows the chunk-wise QoE transitions over multiple adaptive streaming techniques (EDGE360, 360SRL, RAPT360, and BOLA360) in three test videos: *Beach*, *Sunset*, and *Alaska*. These results highlight how each technique controls video quality over time, demonstrating its capacity to maintain stability while adapting to network changes. All testing evaluations were conducted under unstable 5G network trace conditions to emulate real-world dynamic bandwidth scenarios. EDGE360 displays strong QoE performance, delivering excellent and consistent chunk-wise quality. This strengthens its advantage over other adaptive streaming strategies, particularly in dealing with the challenges of viewport-based tile-aware 360° video streaming under dynamic network circumstances. With very little variation, EDGE360 consistently maintains the most constant QoE across all videos, remaining above 75–85 in *Beach* and 80 above in *Sunset*. On the other hand, RAPT360 and 360SRL struggle with bitrate fluctuations and rapid adjustments to video motions, seeing sporadic decreases (such as chunks 20, 35, and 50 in *Beach* and chunks 15 and 42 in *Sunset*). With significant dips at chunks 5, 23, and 45 in *Beach*, 10, 27, and 50 in *Sunset*, and 12, 32, and 48 in *Alaska*, BOLA360 exhibits the lowest performance, underscoring its incapacity to manage varying situations. Although every approach is challenged by *Alaska*, which has the most temporal complexity, EDGE360 still performs better, exhibiting only slight deviations at chunks 30 and 45, whilst RAPT360 and 360SRL have difficulties with motion-heavy regions.

Fig. 13 depicts the learning process of different adaptive streaming techniques across 50 epochs, emphasizing their potential for optimizing reward values. EDGE360 has the highest final average reward, surpassing 360SRL, RAPT360, and BOLA360 by 11.86%, 8.12%, and 18.00%, respectively, demonstrating EDGE360's higher level of convergence and refinement. EDGE360's COMA-based MARL framework employs a centralized critic with an advantage function, allowing for more accurate credit assignment to individual agents. This alleviates the multi-agent credit assignment challenge by discriminating between beneficial joint actions and suboptimal actions. Unlike DDQN, which encounters Q-value networks collapse to effectively approximate the true action-value function, and A3C, which is struggling with partial observability and scalability issues, COMA uses an actor-critic architecture with policy gradients to ensure stable learning and effective

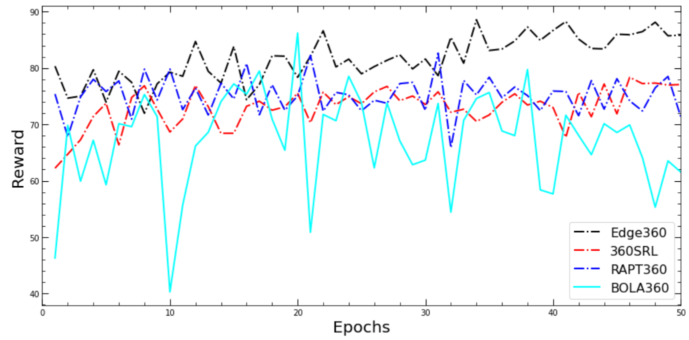


Fig. 13: Average training reward (QoE) achieved by EDGE360, 360SRL, RAPT360, and BOLA360 for approximately 50 epochs.

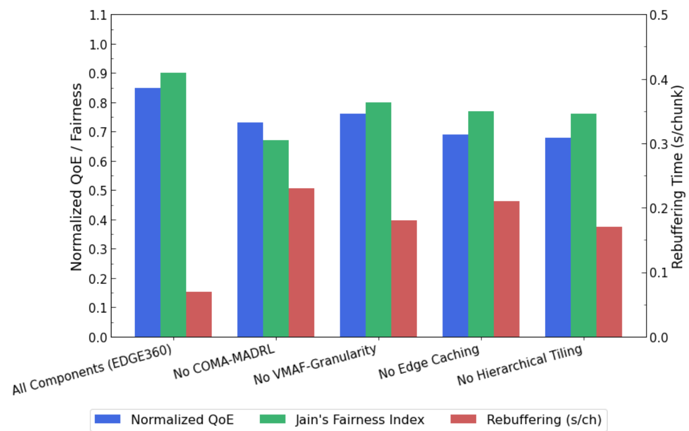


Fig. 14: Ablation study showing the impact of disabling each EDGE360 component individually on normalized QoE, Jain's fairness index, and rebuffering time.

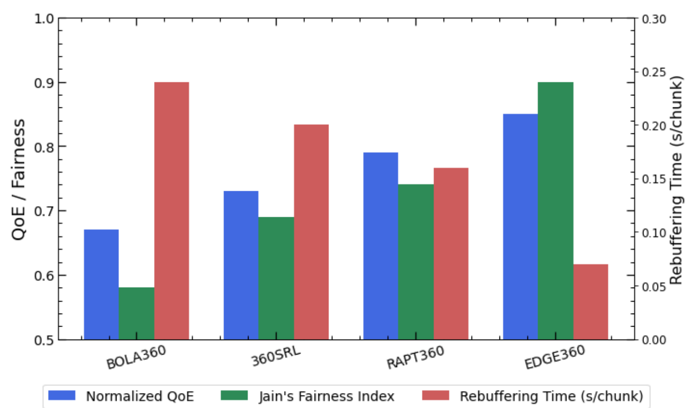


Fig. 15: Comparison of QoE, fairness, and rebuffering time across state-of-the-art ABR streaming schemes.

adaptation to dynamic network circumstances. The centralized critic in COMA allows agents to make coordinated decisions, avoiding the solitary suboptimal updates. This results in more rapid convergence, less inconsistent policy updates and ultimately superior decision-making in a multi-client adaptive streaming environment.

1) Ablation Study and Comparative Evaluation: To evaluate the individual impact of each core module in EDGE360, an ablation study was performed by selectively disabling one

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

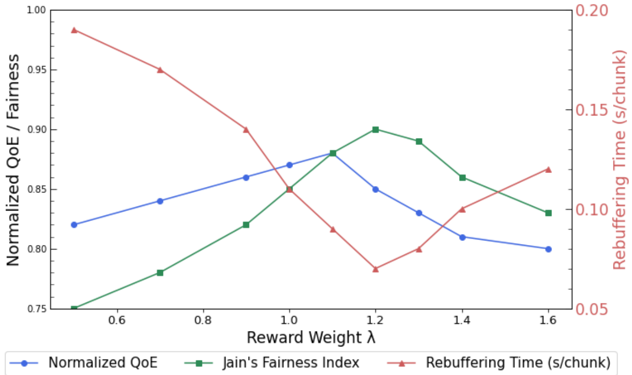


Fig. 16: Sensitivity of EDGE360 to reward weight λ, showing optimal QoE, fairness, and rebuffering near $\lambda = 1.2$.

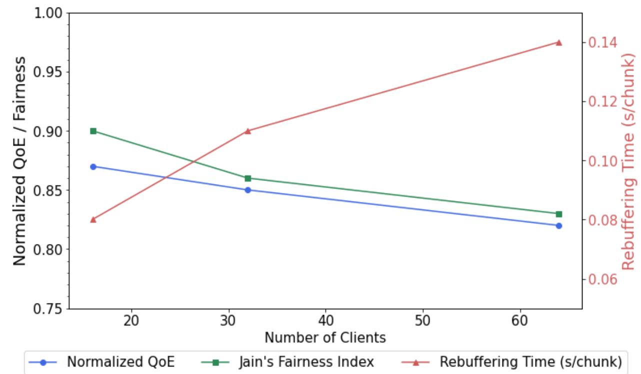


Fig. 17: Scalability of EDGE360 under increasing client load: slight QoE and fairness decline due to bandwidth contention, with low rebuffering maintained by decentralized actors and region-aware tiling.

component at a time and measuring changes in normalized QoE, Jain's fairness index, and rebuffering time per chunk. As shown in Fig. 14, the full EDGE360 configuration achieves the best performance across all metrics.

The deactivation of *COMA-MADRL* leads to a substantial deterioration of performance, resulting in a reduction in fairness from 0.90 to 0.67, which constitutes a 23% decrease, and an increase in rebuffering time from 0.07 s/chunk to 0.26 s/chunk. This underscores the fundamental importance of COMA's counterfactual advantage function in facilitating cooperative bitrate decisions and equitable resource allocation in multi-agent environments.

Disabling the *VMAF-driven granularity* reduces fairness from 0.90 to 0.81 and QoE from 0.86 to 0.76, due to perceptual quality misalignment and inefficient tile prioritization. The removal of this module also increases rebuffering time to 0.18 s/chunk, reflecting wasted bandwidth on exchanging data associated with less relevant regions. Disabling *edge caching* leads to network-induced stalls and delivery delays, lowering QoE to 0.69 and fairness to 0.77, with rebuffering increasing to 0.21 s/chunk due to the reliance on remote content sources. Lastly, removing the *hierarchical tiling* strategy and applying uniform granularity results in fairness dropping to 0.77 and QoE to 0.68, with rebuffering rising to 0.17 s/chunk. This indicates degraded spatial bandwidth efficiency and poorer adaptation to viewport importance.

These results validate EDGE360's modular design: each component contributes significantly to maintaining high-quality, fair, and stall-resilient 360° video streaming.

Fig. 15 provides a comparative analysis of EDGE360 and benchmark methodologies (BOLA360, 360SRL, RAPT360) with respect to three principal performance metrics: normalized QoE, Jain's fairness index, and the average rebuffering duration per segment. EDGE360 demonstrates superior performance over state-of-the-art benchmarks (BOLA360, 360SRL, RAPT360) across all principal metrics, achieving a 26.8% enhancement in QoE, a 55.2% increase in fairness, and a 70.8% reduction in rebuffering time compared to BOLA360; a 16.4% advancement in QoE, a 30.4% improvement in fairness, and a 65% decrease in rebuffering relative to 360SRL; as well as a 7.6% elevation in QoE, a 21.6% betterment in fairness, and a 56.3% diminution in rebuffering time than RAPT360. These findings underscore the effectiveness of the COMA-driven multi-agent coordination, hierarchical region-based tiling, and decentralized actor execution, affirming the framework's real-time adaptability and fairness awareness under various streaming conditions. The implementation of the COMA framework plays a crucial role, as it addresses the multi-agent credit assignment challenge by calculating per-agent counterfactual baselines that elucidate the influence of individual actions on the global reward, substantially enhancing fairness among clients and promoting more stable and equitable learning among agents. In contrast, other methodologies such as 360SRL and RAPT360 lack cooperative coordination among agents, leading to uneven QoE distribution and increased rebuffering.

2) Hyperparameter Tuning and Sensitivity Analysis: To ensure the optimal performance of the COMA-MADRL framework within EDGE360, a systematic hyperparameter tuning process was employed. The actor and critic networks were trained utilizing the Adam optimizer, with learning rates of 1×10^{-4} and 5×10^{-3} respectively. To manage the trade-off between exploration and exploitation, the entropy regularization coefficient was initialized at 0.99 and subjected to an exponential decay by a factor of 0.995 every 10^3 iterations. Furthermore, early stopping was implemented to mitigate overfitting and to guarantee generalization across a variety of client conditions.

We evaluated the sensitivity of EDGE360 to the reward weight λ, which governs the trade-off between maximizing user QoE and promoting fairness. Fig. 16 shows the variation in normalized QoE, Jain's fairness index, and rebuffering time as λ ranges from 0.5 to 1.6. Increasing λ consistently improves fairness, peaking at $\lambda = 1.2$ with a fairness index of 0.90. At this point, normalized QoE also reaches 0.87, and average rebuffering drops to 0.07 s/chunk. Beyond this point, however, overemphasis on fairness leads to QoE degradation and increased buffering.

While the fairness index exceeds normalized QoE numerically, this does not imply superior perceptual quality. Jain's index measures the equity of QoE distribution, while QoE itself reflects the average user-perceived experience, which

TABLE I: Optimized per-decision runtime breakdown of EDGE360 at the edge server under varying client loads

Module	Runtime (ms)
16 Clients	
LSTM-based Viewport Prediction	1.8
VMAF-based Tile Prioritization	2.0
QoE and Utility Evaluation	1.2
Actor Network Inference	1.9
Total Runtime	**6.9**
32 Clients	
Actor Network Inference	3.8
Total Runtime	**8.8**
64 Clients	
Actor Network Inference	7.7
Total Runtime	**12.7**

may still vary across clients due to video complexity or network heterogeneity. This distinction emphasizes the need for balanced optimization.

This analysis substantiates the design of the reward function and underscores the significance of adjusting λ to maintain a balance between fairness and QoE. Additional parameters, such as $\alpha = 0.9$, $\beta = 1.5$, $\zeta = 3.5$, and $|\varepsilon| = 0.8$, were empirically optimized through a grid search to guarantee training stability and dynamic robust learning.

3) Scalability and Real-Time Feasibility: Fig. 17 illustrates the scalability performance of EDGE360 as the number of clients increases from 16 to 64, reflecting typical high-density 5G streaming scenarios. Three critical performance metrics are evaluated: normalized average QoE, Jain's fairness index, and average rebuffering time per chunk.

As depicted in Fig. 17, an increase in the client count from 16 to 64 results in a moderate deterioration in both normalized QoE and fairness. Specifically, the normalized QoE declines from 0.87 to 0.81, eventually reaching 0.77, while the fairness index reduces from 0.92 to 0.84 and subsequently to 0.80. This degradation is predominantly attributed to elevated bandwidth contention among clients utilizing shared wireless access points and edge resources. As a greater number of users concurrently stream high-resolution 360° video content, adaptive mismatches and compromises in bitrate occur with increased frequency.

Despite these challenges, EDGE360 maintains stable rebuffering behavior. The average rebuffering time increases from 0.076s/chunk at 16 clients to 0.10s and 0.14 s at 32 and 64 clients, respectively. This resilience is enabled by two architectural enablers: (i) decentralized actor networks, where each client adapts independently based on its local observations, eliminating centralized inference delays, and (ii) region-aware hierarchical tiling, which intelligently assigns finer granularity to viewports and coarser tiles to peripheral regions, optimizing bandwidth use across the multi-client environment.

Table I, presents the average per-decision runtime breakdown of EDGE360, measured on an edge server. The overall processing pipeline comprises viewport prediction via LSTM, tile prioritization using pre-computed VMAF scores, QoE utility evaluation, and inference by parallel actor networks.

While the first three stages of the EDGE360 pipeline viewport prediction, tile prioritization, and QoE evaluation exhibit fixed latency regardless of the number of clients, their cumulative runtime is significantly optimized, totaling approximately 5.0 ms. In contrast, the actor network inference latency scales linearly with the number of clients but remains lightweight at 0.12 ms per client due to model quantization and efficient design. It is important to note that the centralized critic network in COMA introduces computational overhead only during training and is not active during inference. As a result, the final per-decision runtime at the edge is further reduced, as only the decentralized actor networks are executed. Owing to this modular and parallel architecture, the total per-decision latency remains well within real-time constraints even under the heaviest load of 64 concurrent clients, the overall decision latency stays below 13 ms. This is significantly lower than the typical 2-second video chunk duration, ensuring compliance with ultra-low latency requirements for real-time VR streaming over 5G networks.

This responsiveness enables scalable and real-time bitrate adaptation for high-resolution 360° VR streaming in 5G edge environments. Notably, real-time VR systems require motion-to-photon latency below 20 ms and ideally under 15 ms to prevent discomfort and maintain immersion [46]. Our optimized EDGE360 architecture achieves an end-to-end per-client decision latency consistently under 13 ms, even when supporting up to 64 concurrent clients. This meets the ultra-low latency demands of interactive VR experiences. Moreover, prior research shows that maintaining latency below 15–20 ms can enable up to a fivefold reduction in bitrate for foveated VR video streaming, whereas latencies exceeding 45 ms severely reduce such efficiency gains [47]. Thus, the designed low-latency pipeline of EDGE360 ensures perceptual fidelity and real-time responsiveness, which are necessary for delivering immersive media over next-generation networks.

VII. Conclusions and Future Work

This study presents EDGE360, a COMA-based MADRL framework for edge-driven, viewport-aware, tile-adaptive 360° video streaming. Designed to maximize both client QoE and fairness under heterogeneous network conditions, EDGE360 leverages VMAF-based granularity selection and viewport prediction to intelligently distribute bitrates across tiles, resulting in a more effective DASH-based streaming experience. Unlike prior schemes, RAPT360 (A3C-based), 360SRL (DDQN-based), and BOLA360 (buffer-based) EDGE360 utilize a centralized critic with decentralized actors to coordinate bitrate adaptation across multiple clients, enhancing learning stability, fairness, and efficiency. Its integration of edge caching and deep-learning-based viewport prediction further reduces latency and rebuffering, improving adaptability in dynamic 5G conditions. Extensive evaluations demonstrate that EDGE360 achieves 8.12%, 11.86%, and 18.00% higher average rewards than RAPT360, 360SRL, and BOLA360, respectively.

Future work will explore multi-step COMA learning, enabling the centralized critic to model long-term temporal dependencies and improve proactive bitrate selection in response to network and viewport variations. We also aim to extend EDGE360 with hierarchical COMA, creating a multi-level decision architecture where the edge server optimizes rate adaptation between the video server and multiple clients. Finally, cost-efficiency will be addressed by dynamically prioritizing and reducing tile transmission based on predicted viewports, making adaptive 360° streaming more scalable in diverse network environments.

Acknowledgment

This work is funded by the European Regional Development Fund through the Research Ireland Centres Programme under Grant Number SFI/12/RC/2289_P2 (Insight Centre for Data Analytics) and 21/FFP-P/10244 (FRADIS).

References

[1] H. Xiao, C. Xu, Z. Feng, R. Ding, S. Yang, L. Zhong, J. Liang, and G.-M. Muntean, "A transcoding-enabled 360 vr video caching and delivery framework for edge-enhanced next-generation wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 5, pp. 1615–1631, 2022.

[2] N. Kan, J. Zou, C. Li, W. Dai, and H. Xiong, "Rapt360: Reinforcement learning-based rate adaptation for 360-degree video streaming with adaptive prediction and tiling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1607–1623, 2021.

[3] Z. Hu, J. Niu, T. Ren, and M. Guizani, "Achieving fast environment adaptation of drl-based computation offloading in mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 23, no. 5, pp. 6347–6362, 2024.

[4] X. Ma, Q. Li, Y. Jiang, G.-M. Muntean, and L. Zou, "Learning-based joint qoe optimization for adaptive video streaming based on smart edge," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1789–1806, 2022.

[5] X. Ma, Q. Li, L. Zou, J. Peng, J. Zhou, J. Chai, Y. Jiang, and G.-M. Muntean, "Qava: Qoe-aware adaptive video bitrate aggregation for http live streaming based on smart edge computing," *IEEE Transactions on Broadcasting*, vol. 68, no. 3, pp. 661–676, 2022.

[6] L. Zhang, X. Wu, F. Wang, A. Sun, L. Cui, and J. Liu, "Edge-based video stream generation for multi-party mobile augmented reality," *IEEE Transactions on Mobile Computing*, vol. 23, no. 1, pp. 409–422, 2024.

[7] F. E. Subhan, A. Yaqoob, C. H. Muntean, and G.-M. Muntean, "A survey on artificial intelligence techniques for improved rich media content delivery in a 5g and beyond network slicing context," *IEEE Communications Surveys & Tutorials*, 2024.

[8] Y. Wang, J. Li, Z. Li, S. Shang, and Y. Liu, "Synergistic temporal-spatial user-aware viewport prediction for optimal adaptive 360-degree video streaming," *IEEE Transactions on Broadcasting*, 2024.

[9] X. Liu, C. Vlachou, F. Qian, C. Wang, and K.-H. Kim, "Firefly: Untethered multi-user {VR} for commodity mobile devices," in *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, 2020, pp. 943–957.

[10] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[11] Z. Jiang, Y. Xu, J. Sun, J.-N. Hwang, Y. Zhang, and S. C. Appleby, "Tile-based panoramic video quality assessment," *IEEE Transactions on Broadcasting*, vol. 68, no. 2, pp. 530–544, 2021.

[12] J. Zou, T. Hao, C. Yu, and H. Jin, "A3c-do: A regional resource scheduling framework based on deep reinforcement learning in edge scenario," *IEEE Transactions on Computers*, vol. 70, no. 2, pp. 228–239, 2020.

[13] C. Lee, J. Jung, and J.-M. Chung, "Intelligent dual active protocol stack handover based on double dqn deep reinforcement learning for 5g mmwave networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 7, pp. 7572–7584, 2022.

[14] J. Zou, C. Li, C. Liu, Q. Yang, H. Xiong, and E. Steinbach, "Probabilistic tile visibility-based server-side rate adaptation for adaptive 360-degree video streaming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 1, pp. 161–176, 2019.

[15] S. Altamimi and S. Shirmohammadi, "Qoe-fair dash video streaming using server-side reinforcement learning," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 2s, pp. 1–21, 2020.

[16] Y. Yuan, W. Wang, Y. Wang, S. S. Adhatarao, B. Ren, K. Zheng, and X. Fu, "Joint optimization of qoe and fairness for adaptive video streaming in heterogeneous mobile environments," *IEEE/ACM Transactions on Networking*, pp. 1–15, 2023.

[17] F. Qian, B. Han, Q. Xiao, and V. Gopalakrishnan, "Flare: Practical viewport-adaptive 360-degree video streaming for mobile devices," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, 2018, pp. 99–114.

[18] J. Fu, X. Chen, Z. Zhang, S. Wu, and Z. Chen, "360srl: A sequential reinforcement learning approach for abr tile-based 360 video streaming," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2019, pp. 290–295.

[19] Z. Li, P. Zhong, J. Huang, F. Gao, and J. Wang, "Achieving qoe fairness in bitrate allocation of 360° video streaming," *IEEE Transactions on Multimedia*, 2023.

[20] H. Wang, Z. Long, H. Dong, and A. El Saddik, "Madrl-based rate adaptation for 360° video streaming with multi-viewpoint prediction," *IEEE Internet of Things Journal*, 2024.

[21] A. Zeynali, M. H. Hajiesmaili, and R. K. Sitaraman, "Bola360: Near-optimal view and bitrate adaptation for 360-degree video streaming," in *Proceedings of the 15th ACM Multimedia Systems Conference*, 2024, pp. 12–22.

[22] S. Bi, H. Chen, X. Li, S. Wang, Y. Wu, and L. Qian, "A two-stage deep reinforcement learning framework for mec-enabled adaptive 360-degree video streaming," *IEEE Transactions on Mobile Computing*, 2024.

[23] S. Kumar, A. Franklin A, J. Jin, and Y.-N. Dong, "Seer: Learning-based 360° video streaming for mec-equipped cellular networks," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 6, pp. 3308–3319, 2023.

[24] X. Shi, S. Zhang, M. Liu, L. Meng, L. Wei, Y. Gu, K. Liu, H. Cheng, Y. Song, L. Tang, A. Zhu, N. Chen, and Z. Qian, "Mystique: User-level adaptation for real-time video analytics in edge networks via meta-rl," *IEEE Transactions on Mobile Computing*, vol. 24, no. 5, pp. 3615–3632, 2025.

[25] Y. Ma, C. Xu, Z. Wu, R. Ding, H. Xiao, L. Zhong, Y. Zhuang, and G.-M. Muntean, "Ca-live360: Crowd-assisted transcoding and delivery for live 360-degree video streaming," *Computer Networks*, vol. 251, p. 110618, 2024.

[26] B. Hou, S. Yang, F. Li, L. Zhu, X. Chen, Y. Wang, and X. Fu, "Nova: Neural-optimized viewport adaptive 360-degree video streaming at the edge," *IEEE Transactions on Services Computing*, 2024.

[27] R. Deng, "Qoe optimized adaptive streaming method for 360° virtual reality videos over mec-assisted network," *Multimedia Tools and Applications*, vol. 83, no. 17, pp. 52 737–52 761, 2024.

[28] J. Zeng, X. Zhou, and K. Li, "Madrl-based joint edge caching and bitrate selection for multicategory 360° video streaming," *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 584–596, 2023.

[29] Y. Guan, C. Zheng, X. Zhang, Z. Guo, and J. Jiang, "Pano: Optimizing 360 video streaming with a better understanding of quality perception," in *Proceedings of the ACM Special Interest Group on Data Communication*. ACM, 2019, pp. 394–407.

[30] R. G. d. A. Azevedo, N. Birkbeck, F. De Simone, I. Janatra, B. Adsumilli, and P. Frossard, "Visual distortions in 360° videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2524–2537, 2019.

[31] U. U. Khan, N. Dilshad, M. H. Rehmani, and T. Umer, "Fairness in cognitive radio networks: models, measurement methods, applications, and future research directions," *Journal of Network and Computer Applications*, vol. 73, pp. 12–26, 2016.

[32] A. Balcı and R. Sokullu, "Fairness aware deep reinforcement learning for grant-free noma-iot networks," *Internet of Things*, vol. 25, p. 101079, 2024.

[33] Y. Wang, Y. Wan, C. Zhang, L. Bai, L. Cui, and P. Yu, "Competitive multi-agent deep reinforcement learning with counterfactual thinking," in *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019, pp. 1366–1371.

[34] H. Zhou, T. Lan, and V. Aggarwal, "Pac: Assisted value factorization with counterfactual predictions in multi-agent reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 15 757–15 769, 2022.

[35] X. Liu and K. Zhang, "Partially observable multi-agent rl with (quasi-) efficiency: the blessing of information sharing," in *International Conference on Machine Learning*. PMLR, 2023, pp. 22 370–22 419.

[36] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

This article has been accepted for publication in IEEE Transactions on Mobile Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMC.2025.3605849

[37] Y. Bao, H. Wu, T. Zhang, A. A. Ramli, and X. Liu, "Shooting a moving target: Motion-prediction-based transmission for 360-degree videos," in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 1161–1170.

[38] H.-N. Hu, Y.-C. Lin, M.-Y. Liu, H.-T. Cheng, Y.-J. Chang, and M. Sun, "Deep 360 pilot: Learning a deep agent for piloting through 360° sports videos," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1396–1405.

[39] Y. Guo, M. Xu, L. Jiang, X. Deng, J. Zhou, G. Chen, and L. Sigal, "Proposal with alignment: A bi-directional transformer for 360° video viewport proposal," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11 423–11 437, 2024.

[40] "A public database of 360 videos with corresponding ratings of arousal and valence," in *Virtual Human Interaction Lab. Stanford University*. [Online]. Available: http://vhil.stanford.edu/360-video-database/

[41] Ffmpeg: A complete, cross-platform solution to record, convert and stream audio and video. [Online]. Available: http://ffmpeg.org/

[42] "Fragmentation, segmentation, splitting and interleaving." [Online]. Available: https://gpac.wp.imt.fr/mp4box/

[43] 3rd Generation Partnership Project, "3gpp tr 26.918: Virtual reality (vr) media services over 3gpp, 2017." [Online]. Available: http://www.3gpp.org/ftp/tsg_sa/WG4_CODEC/TSGS4_94/Docs/S4-170752.zip

[44] M. Orduna, C. Díaz, L. Muñoz, P. Pérez, I. Benito, and N. García, "Video multimethod assessment fusion (vmaf) on 360vr contents," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 1, pp. 22–31, 2020.

[45] 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," ser. MMSys '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 162–173. [Online]. Available: https://doi.org/10.1145/3204949.3204978

[46] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Network*, vol. 32, no. 2, pp. 78–84, 2018.

[47] L. Hsiao, B. Krajancich, P. Levis, G. Wetzstein, and K. Winstein, "Towards retina-quality vr video streaming: 15ms could save you 80% of your bandwidth," *ACM SIGCOMM Computer Communication Review*, vol. 52, no. 1, pp. 10–19, 2022.

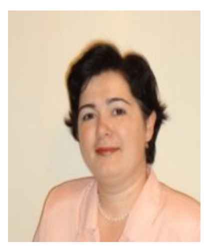
[48] Y. Chen, S. Omoma, H. Kwon, H. Inaltekin, and M. Gorlatova, "Quantifying and exploiting vr frame correlations: An application of a statistical model for viewport pose," *IEEE Transactions on Mobile Computing*, 2024.

[49] L. Xie, Z. Xu, Y. Ban, X. Zhang, and Z. Guo, "360probdash: Improving qoe of 360 video streaming using tile-based http adaptive streaming," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 315–323.

[50] J. Chakareski, R. Aksu, X. Corbillon, G. Simon, and V. Swaminathan, "Viewport-driven rate-distortion optimized 360º video streaming," *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:4118274

Fazal E Subhan received a B.Sc. degree in Electrical Engineering and an M.S.c degree in Renewable Energy Engineering from University of Engineering and Technology, Peshawar, Pakistan, in 20015 and 2019, respectively. He's currently pursuing his PhD degree in Electronic Engineering from Dublin City University (DCU), Ireland. In 2018, he worked as a Research Exchange Scholar in the Holman Research Group at the School of Electrical, Computer and Energy Engineering, Arizona State University (ASU), USA.

Abid Yaqoob (M'22) received a B.Sc. degree in computer systems engineering from the Islamia University of Bahawalpur, Pakistan, in 2014, and an M.Sc. degree in network and information security from Northwestern Polytechnical University, Xi'an, China, in 2018. In 2022, he completed his Ph.D. degree with the Performance Engineering Laboratory and the Insight Centre for Data Analytics, School of Electronic Engineering, Dublin City University, Ireland. His research interests include mobile wireless communication, priority-aware differentiated video streaming, as well as immersive multimedia processing and delivery solutions.



Cristina Hava Muntean is a Professor and Vice-Dean with the School of Computing, National College of Ireland. She holds a Ph.D. degree awarded by Dublin City University, Ireland in 2005. She has been constantly involved in various research related activities over the past 16 years fostering and promoting research, leading research projects, supervising PhD and MSc students and publishing over 100 publications in international peer-reviewed books, journals and conferences. Adaptive and personalised e/m- learning, user modelling, technology enhanced learning, game-based learning, self-directed learning, consumer behaviour, end-user Quality of Experience, adaptive multimedia, energy saving solutions are the main research areas she is involved in. She was the NCI coordinator of the EU Horizon 2020 project NEWTON.

Gabriel-Miro Muntean (F'22) is a Professor with the School of Electronic Engineering, Dublin City University (DCU), Ireland, and the co-Director of the DCU Performance Engineering Laboratory. His research interests include quality, performance, and energy-saving issues related to multimedia and multiple sensorial media delivery, technology-enhanced learning, and other data communications over heterogeneous networks. He is an Associate Editor of the IEEE TRANSACTIONS ON BROADCASTING, the Multimedia Communications Area Editor of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, and a reviewer for important international journals, conferences, and funding agencies. He was the Project Coordinator for the EU-funded project NEWTON and led the DCU team in the EU projects TRACTION and HEAT.