VR-GX: an Attention-aware QoE-based resource allocation model for VR-Cloud Gaming

Gabriel M. Almeida[1], **João Paulo Esper**[2], **Luiz A. DaSilva**[3], **Kleber V. Cardoso**[1]

[1] Instituto de Informática - Universidade Federal de Goiás - Brazil,
[2]Departamento de Ciência da Computação - Universidade Federal de Minas Gerais - Brazil,
[3]Commonwealth Cyber Initiative - Virginia Tech - USA.

gabrielmatheus@inf.ufg.br; joaopauloesper@dcc.ufmg.br;
ldasilva@vt.edu; kleber@inf.ufg.br.

Abstract. *Virtual Reality Cloud Gaming (VR-CG) applications demand high computational and network resources due to their immersive nature and user-specific needs. To address these demands, we propose VR-GX, a mathematical formulation for resource allocation that incorporates user attention levels toward virtual objects within their Field of View (FoV) to optimize the Quality of Experience (QoE). Leveraging 3GPP specifications, VR-GX adjusts object resolutions based on attention levels, minimizing unnecessary data transmission and enhancing network efficiency. We compare VR-GX with a state-of-the-art model, demonstrating that our formulation consistently achieves higher QoE and fairness across various scenarios, particularly as user numbers increase. A heuristic algorithm is introduced to approximate solutions efficiently while maintaining QoE and latency fairness without exceeding computational limits. Our findings underline the significance of integrating user-centric features in VR-CG environments to ensure resource-efficient and high-quality user experiences in scalable, real-world immersive applications.*

1. Introduction

Sixth generation (6G) networks are set to transform connectivity, delivering unprecedented support for dense user environments and ultra-reliable communication. Among the enabling technologies of 6G networks are the Semantic Communication (SC) and the Computing and Network Convergence (CNC) architecture [Shokrnezhad et al. 2024]. SC aims to minimize the transmission load of applications by focusing on the meaning of the information rather than merely transmitting raw data [Chaccour et al. 2024], ultimately reducing bandwidth consumption, and improving network performance. CNC brings computation capabilities closer to the user, by deploying Computing Nodes (CNs) closer to users while integrating a holistic allocation of computing and network resources. These CNs handle resource-intensive tasks, such as rendering video frames and reducing latency for latency-sensitive applications, e.g., immersive applications.

One of the key immersive applications defined by the 3rd Generation Partnership Project (3GPP) is Virtual Reality Cloud Gaming (VR-CG) [3GPP 2019,3GPP 2022,3GPP 2023]. VR-CG allows users to engage in high-fidelity VR games rendered remotely by CNs, spread in the Radio Access Network (RAN) infrastructure, and delivered in real-time via cellular network [Huawei 2018]. A significant advantage of VR-CG lies in its ability to offload game frame rendering, eliminating the need for users to invest in high-performance hardware and increasing availability of VR applications [Baena et al. 2024].

The emergence of VR-CG technology introduces a challenging resource allocation problem [3GPP 2022], as high-demand immersive applications create intense resource competition, directly impacting the user experience. Consequently, both academia and industry have recently focused on properly formulating the VR-CG resource allocation problem. However, most of the existing formulations [3GPP 2022, Huawei 2018, Baena et al. 2024] overlook the full dynamic nature of VR-CG environment, often lacking comprehensive wireless communication models, dynamic image resolution selection, semantic communication aspects, and adaptive frame rate. In this context, a comprehensive formulation of the VR-CG resource allocation problem is essential to develop effective solutions that can meet the real-world scenarios [3GPP 2022].

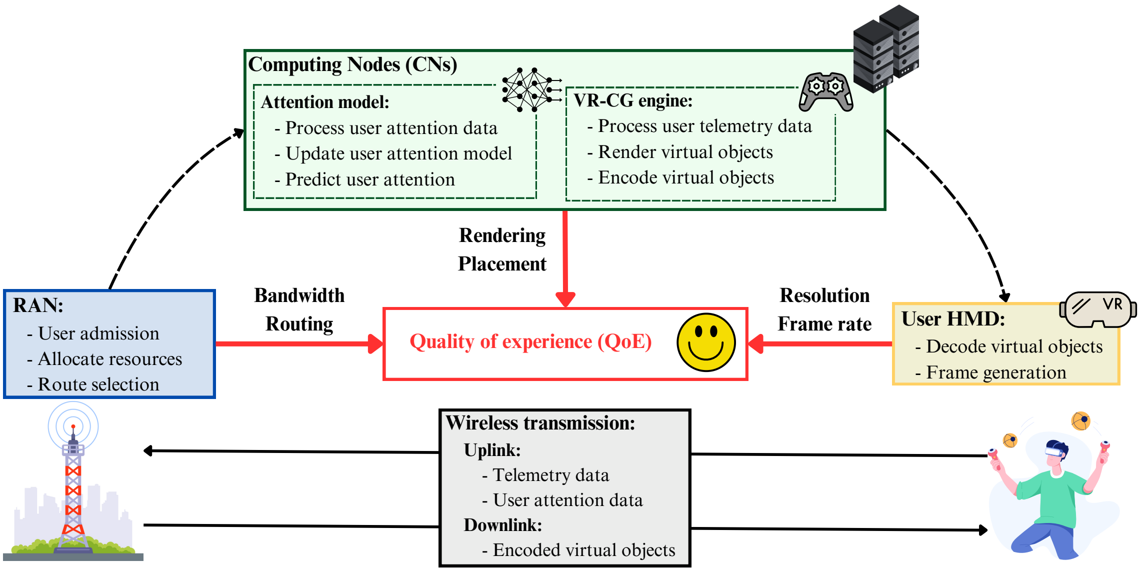


Figure 1. VR-CG resource allocation problem.

Figure 1 illustrates the VR-CG environment. Users connect to the RAN via wireless links with Base Stations (BSs) and Head Mounted Displays (HMDs). BSs, which may be disaggregated, are connected to CNs through a crosshaul. The CNs handle critical tasks such as running the VR-CG engine, rendering and encoding virtual objects, and processing the attention model. The attention model uses data collected from the HMDs to update attention profiles and predict user attention in the next frame. The Quality of Experience (QoE) for users is directly influenced by the image resolution and frame rate, which significantly affect network bandwidth usage, routing latency, and VR-CG applications placement. As a result, VR-CG resource allocation becomes a complex joint optimization problem, requiring the careful selection of image resolution and frame rate while simultaneously optimizing both communication and computing resource allocation.

1.1. Related Work

In [Xia et al. 2023], the authors present *WiserVR*, a framework designed for semantic-driven 360º VR video transmission, minimizing latency and enhancing transmission reliability. They define the Semantic Location Graph (SLG), which captures and represents semantic features of static and dynamic objects. This SLG model optimizes resource usage and improves user QoE by prioritizing essential information. However, while SLG semantic features effectively transmit multiple frames simultaneously, well-suited for 360º VR video applications, it does not align with the VR-CG applications need for frame-by-frame transmission [3GPP 2019, 3GPP 2023, 3GPP 2022].

In [Baena et al. 2024], authors explore how cloud gaming user experience correlates with network quality, defining minimum throughput and maximum latency requirements. They discuss the impact of the transmission load derived from the ultra-high resolution, and show how frame rates depend on network latency and throughput. However, their work lacks a mathematical formulation of the resource allocation problem, and does not account for users connected to different BSs, as outlined by 3GPP [3GPP 2022].

The authors of [Alhilal et al. 2024] introduce *FovOptix*, an algorithm that leverages foveated rendering and video encoding to reduce bitrate demands in VR-CG video transmission. Although effective in minimizing bitrate, their work fails to consider the guaranteed maximum latency and minimum throughput defined by 3GPP [3GPP 2019, 3GPP 2023]. Instead, their algorithm focuses on minimizing these metrics without ensuring specific thresholds.

In [Du et al. 2023a], the authors present a mathematical formulation for resource allocation in VR video transmission within the Metaverse[1]. The model employs users' attention level to determine the rendering resolutions of virtual objects. However, the model assumes all users are connected to the same BS, disregarding the possibility of multiple BSs, and do not consider offloading, therefore assuming all users have powerful hardware to process and render VR frames locally. It also overlooks latency in its QoE model, and lacks support for adaptive frame rates, making it incompatible with 3GPP standards definitions [3GPP 2019, 3GPP 2022, 3GPP 2023].

In summary, while these studies contribute to the resource allocation problem for immersive applications, none of the presented formulations fully adhere to 3GPP standards. They all overlook critical elements such as: dynamic resolution and frame rate, proper QoE formulations, and multi-BS scenarios. Table 1 presents a comparison between the literature and our work, considering some relevant aspects.

Reference	3GPP compliant	Multi-BS	Adaptative	Semantic Communication	QoE metrics
[Xia et al. 2023]	×	×	×	✓	Only latency
[Baena et al. 2024]	×	×	×	×	Resolution and latency
[Alhilal et al. 2024]	×	×	Resolution and frame rate	✓	Only resolution
[Du et al. 2023a]	×	×	Only resolution	✓	Resolution and attention
This work	✓	✓	**Resolution, frame rate and routing**	✓	**Resolution, frame rate, and attention**

Table 1. Related work comparison.

[1] https://about.meta.com/metaverse/

1.2. Our Contributions and Paper Organization

In this work, we address the described gaps by introducing VR-GX, a mathematical formulation aligned with the 3GPP standards, optimizing resource allocation for VR-CG applications in a dynamic multi-BS RAN environment while considering semantic features to transmit rendered frames from remote CNs to users. Our main contributions are presented below.

- We formulate the VR-CG resource allocation problem, considering dynamic adaptation of video resolution, frame rate, and bandwidth allocation based on users' channel quality. We consider a multi-BS RAN environment with CNs spread across the network to render the VR-CG video frames.
- We incorporate the communication system characteristics defined by 3GPP for VR-CG that influence user experience, leading to the development of a novel QoE model based on concepts of the *Weber-Fechner* law of satisfaction.
- We address a multi-BS RAN topology, integrating SC features to minimize the impact of the high transmission load generated by VR-CG frames.
- The entire implementation of the optimization model, heuristic, and the data used in the evaluation experiments are publicly available in a GitHub repository.

This paper is organized as follows. Section 2 presents the system model of our formulation. Section 3 formulates the VR-CG resource allocation problem. Section 4 outlines a heuristic approach to solving the formulated problem. Evaluation results are presented and discussed in Section 5, and finally, we conclude with our final considerations and future work in Section 6.

2. System Model

In our RAN system, we define a set of BS, denoted by $\mathcal{B} = \{b_1, \ldots, b_{|\mathcal{B}|}\}$, and a set of CNs, denoted by $\mathcal{C} = \{c_1, \ldots, c_{|\mathcal{C}|}\}$, responsible for processing the game engines and rendering VR-CG applications video frames. Each CN is characterized by its capacities for processing, memory, network, and rendering, denoted as c_m^{CPU}, c_m^{Mem}, c_m^{Net}, and c_m^{GPU}, respectively. The CNs are distributed across the RAN topology and are interconnected via transport nodes, represented by $\mathcal{T} = \{t_1, \ldots, t_{|\mathcal{T}|}\}$. Additionally, the core network, c_0, is connected to the BSs through transport nodes.

We model RAN topology as a graph $G = \{V, E\}$, where $V = \{c_0\} \cup \mathcal{T} \cup \mathcal{B} \cup \mathcal{C}$ represents the set with all nodes, and $E = \{e_{ij} : v_i, v_j \in \mathcal{V}\}$ denotes the set of edges. Each edge is defined by its capacity and latency, denoted by e_{ij}^{Cap} and e_{ij}^{Lat}, respectively. We consider a set of paths $\mathcal{P}_l^m$ from each CN c_m to the corresponding BS $b_l \in \mathcal{B}$. These paths transport VR-CG application data from CNs to the BSs, which then deliver it to users via wireless links. As illustrated in Figure 1, the VR-CG environment includes a set of users $\mathcal{U} = \{u_1, \ldots, u_{|\mathcal{U}|}\}$. Each user is characterized by their Signal-to-Interference plus Noise Ratio (SINR) from their device to a BS. Each user $u_i \in \mathcal{U}$ is also equipped with a Head-Mounted Display (HMD) defined by its terminal resolution (e.g., 1080p, 2K, or 4K) for each eye and its frame rate (e.g., 30 Hz, 60 Hz, or 90 Hz), denoted by $Res(u_i)$ and $FPS(u_i)$, respectively.

In this work, we propose a dynamic VR-CG environment considering attention-aware resource allocation. Our formulation incorporates user attention levels as a semantic feature, adjusting object rendering resolution within the Field of View (FoV) based on attention, i.e., high-attention objects are rendered in higher resolution, while lower-attention objects are rendered in a reduced resolution. This adaptive strategy minimizes transmission load, optimizing network resources for VR-CG applications. In this way, we define a set of virtual objects $\mathcal{O}_i = \{o_1, \ldots, o_{|\mathcal{O}_i|}\}$ that compose the VR-CG virtual environment within the FoV of user $u_i \in \mathcal{U}$. Each virtual object in the scene must be rendered and transmitted to the user's HMD at a resolution from the set $\mathcal{R} = \{r_1, \ldots, r_{|\mathcal{R}|}\}$, and the video image must be rendered in frame rate from the set $\mathcal{F} = \{f_1, \ldots, f_{|\mathcal{F}|}\}$.

Finally, we define the attention level, denoted as $\lambda_i^j \in \mathbb{R}$, for each user $u_i \in \mathcal{U}$ towards each object $o_j \in \mathcal{O}_i$, which is estimated independently based on user preferences, historical data, eye-tracking, and interactions [Du et al. 2023b].

3. Problem Formulation

In this section, we introduce the **V**irtual **R**eality cloud **G**aming resource allocation based on Quality of e**X**perience (VR-GX) problem formulation aligned with 3GPP standards [3GPP 2019, 3GPP 2023, 3GPP 2022]. VR-GX selects the rendering resolution for each virtual object and the appropriate video frame rate for each user to maximize their QoE while obeying infrastructure capacity limits. For the allocation of communication resources, we define two sets of decision variables: $y_l^i \in \{0, 1\}$ indicating if user $u_i \in \mathcal{U}$ is admitted by BS $b_l \in \mathcal{B}$, and $x_l^i \in \mathbb{Z}$ representing the bandwidth allocated to user $u_i \in \mathcal{U}$ in BS $b_l \in \mathcal{B}$. For computing resources, we define three decision variables: $w_{i,j}^k \in \{0, 1\}$, for selecting the resolution $r_k \in \mathcal{R}$ to object $o_j \in \mathcal{O}_i$ viewed by user $u_i \in \mathcal{U}$, variable $v_i^f \in \{0, 1\}$ for selecting frame rate $f \in \mathcal{F}$ to user $u_i \in \mathcal{U}$, and variable $z_i^m \in \{0, 1\}$ to determine if the VR-CG application of user $u_i \in \mathcal{U}$ must run on CN $c_m \in \mathcal{C}$.

To formulate the VR-GX QoE model, we account for the user's attention level to virtual objects within the user's FoV, the resolution of each virtual object, and the video frame rate. Furthermore, we incorporate concepts from the *Weber-Fechner* satisfaction law, as discussed in [Du et al. 2023a]. We represent the QoE of user u_i as follows:

$$\zeta(i) = \sum_{f \in \mathcal{F}} v_i^f \ln(f) + \sum_{o_j \in \mathcal{O}_i} \sum_{r_k \in \mathcal{R}} \left(w_{i,j}^k \ln(\tau(j,k)\lambda_i^j) \right), \tag{1}$$

where $\tau(j, k)$ represents the resolution quality coefficient and λ_i^j represents the user u_i attention level to the object o_j.

The objective function of our formulation aims to maximize the sum of users' QoE, which we define as follows:

$$\text{maximize} \sum_{u_i \in \mathcal{U}} \zeta(i). \tag{2}$$

In the following sections, we introduce the constraints that define VR-CG application requirements, the wireless transmission model, user throughput, and latency needs. Additionally, we present constraints for user admission control and VR-CG placement in CNs. These constraints are grouped into two parts: the first covers computational resources, while the second addresses communication resources and the wireless transmission model.

3.1. Computational resource constraints

Each virtual object $o_j \in \mathcal{O}_i$ in the FoV of user $u_i \in \mathcal{U}$ must be rendered at exactly one resolution $r_k \in \mathcal{R}$, which is represented by the following constraint:

$$\sum_{r_k \in \mathcal{R}} w_{i,j}^k = 1, \qquad \forall u_i \in \mathcal{U}, o_j \in \mathcal{O}_i. \tag{3}$$

Exactly one video frame rate $f \in \mathcal{F}$ must be selected for each user $u_i \in \mathcal{U}$, which is represented by the following constraint:

$$\sum_{f \in \mathcal{F}} v_i^f = 1, \qquad \forall u_i \in \mathcal{U}. \tag{4}$$

Each user $u_i \in \mathcal{U}$ plays a single VR-CG application that must run on one $c_m \in \mathcal{C}$, which hosts the game engine and delivers the related VR-CG data. Each VR-CG application must run on at most one CN, ensuring that users connect to a single game engine and receive VR-CG data from the same game instance they are interacting with. This constraint is defined as:

$$\sum_{c_m \in \mathcal{C}} z_i^m = 1, \qquad \forall u_i \in \mathcal{U}. \tag{5}$$

Our formulation considers a multiplayer game scenario where users share the same game instance, as in co-op multiplayer games. Thus, users must connect to the same VR-CG application to ensure that VR-CG video frames remain consistent, allowing actions from one user to impact the virtual environment of all players and enabling real-time interaction across locations. The following constraint represents this scenario:

$$z_i^m \Phi(u_i, u_j) \leq z_j^m, \qquad \forall u_i, u_j \in \mathcal{U}, c_m \in \mathcal{C}, \tag{6}$$

where $\Phi(u_i, u_j) \in \{0, 1\}$ indicates if users u_i and $u_j \in \mathcal{U}$ are playing together. If $\Phi(u_i, u_j) = 1$, then $z_i^m = z_j^m$, i.e., the VR-CG application runs on the same CN $c_m \in \mathcal{C}$. If $\Phi(u_i, u_j) = 0$, z_i^m and z_j^m may differ. Additionally, if $\Phi(u_i, u_j) = 1$ and $z_i^m = 0$, Equation (6) ensures $z_j^m = 0$.

Finally, we define the capacity constraints for CNs, covering rendering resource usage, based on object resolutions, processing usage from active VR-CG engines, memory usage to meet VR-CG demands, and network capacity to ensure transmission bandwidth. These constraints are formulated as follows:

$$\sum_{u_i \in \mathcal{U}} z_i^m \mathbf{G}(\mathcal{O}_i, \mathbf{w}_{i,j}^k) \leq c_m^{GPU}, \qquad \forall c_m \in \mathcal{C}, \tag{7}$$

$$\sum_{u_i \in \mathcal{U}} z_i^m \mathbf{C}(\mathcal{O}_i, \mathbf{w}_{i,j}^k) \leq c_m^{CPU}, \qquad \forall c_m \in \mathcal{C}, \tag{8}$$

$$\sum_{u_i \in \mathcal{U}} z_i^m \mathbf{M}(\mathcal{O}_i, \mathbf{w}_{i,j}^k) \leq c_m^{RAM}, \qquad \forall c_m \in \mathcal{C}, \tag{9}$$

$$\sum_{u_i \in \mathcal{U}} z_i^m \mathbf{N}(\mathcal{O}_i, \mathbf{w}_{i,j}^k) \leq c_m^{Net}, \qquad \forall c_m \in \mathcal{C}, \tag{10}$$

where $\mathbf{w}_{i,j}^k$ represents the set of all decision variables $w_{i,j}^k$. $\mathbf{G}(\mathcal{O}_i, \mathbf{w}_{i,j}^k)$ denotes the rendering load in flops per pixel, $\mathbf{C}(\mathcal{O}_i, \mathbf{w}_{i,j}^k)$ represents the processing workload, $\mathbf{M}(\mathcal{O}_i, \mathbf{w}_{i,j}^k)$ is the memory load for game engine drivers, and $\mathbf{N}(\mathcal{O}_i, \mathbf{w}_{i,j}^k)$ indicates the transmission load of the rendered VR-CG frame.

3.2. Wireless communication resource constraints

This part outlines all communication-related constraints, the RAN environment, and the transmission model for the VR-CG use case [3GPP 2019, 3GPP 2023, 3GPP 2022]. First, we establish that each user $u_i \in \mathcal{U}$ must be associated with exactly one BS to ensure they receive data streams exclusively from that BS, which is formalized as:

$$\sum_{b_l \in \mathcal{B}} y_l^i = 1, \qquad \forall u_i \in \mathcal{U}. \tag{11}$$

BSs can dynamically allocate bandwidth to users to optimize resource usage. However, given their finite capacity, the total bandwidth allocated to users must not exceed its maximum capacity, denoted by b_l^{BW}. This is formulated as:

$$\sum_{u_i \in \mathcal{U}} x_l^i \leq b_l^{BW}, \qquad b_l \in \mathcal{B}. \tag{12}$$

Each user $u_i \in \mathcal{U}$ has a throughput demand that is defined by two factors: (i) the transmission load from the selected rendering resolution and (ii) the wireless channel quality. We calculate the transmission load for user $u_i \in \mathcal{U}$ as follows:

$$Load(u_i) = \sum_{o_j \in \mathcal{O}_i} \sum_{r_k \in \mathcal{R}} \left(w_{i,j}^k \Psi(j,k) \mathbf{C}(\mathcal{O}_i) \right), \tag{13}$$

where $\mathbf{C}(\mathcal{O}_i)$ is the image compression rate in bits per pixel, and $\Psi(j,k)$ is the size of the object $o_j \in \mathcal{O}$, measured in pixels, according to the selected rendering resolution $r_k \in \mathcal{R}$.

As defined by 3GPP [3GPP 2022], the wireless communication model for the VR-CG use case is a 4×4 Single User Multiple-Input Multiple-Output (SU-MIMO) system in Frequency Range 1 (FR1), covering sub-6 GHz frequencies. This model features 4 antennas each at the transmitter and receiver, enabling the transmission of four distinct data streams to one user. We calculate the user's offered throughput using Shannon's capacity equation based on the allocated bandwidth, treating each stream independently regarding SINR with adequately spaced antennas [Tse and Viswanath 2005]. The user's throughput capacity is defined as follows:

$$T_{offer}(u_i) = \sum_{s \in \mathbb{S}} \sum_{b_l \in \mathcal{B}} x_l^i \log_2 \left(1 + SINR(u_i, s)\right), \tag{14}$$

where $\mathbb{S}$ represent the number of data streams transmitted to the user $u_i \in \mathcal{U}$, and $SINR(u_i, s)$ denotes the SINR of user $u_i \in \mathcal{U}$ in stream $s \in \mathbb{S}$, calculated as follows:

$$SINR(u_i, s) = \frac{\mathbf{h}_s^H \mathbf{P}_s \mathbf{h}_s}{\sum_{t \neq s} \mathbf{h}_t^H \mathbf{P}_t \mathbf{h}_t + \sigma^2}, \tag{15}$$

where $\mathbf{h}_s$ is the channel vector for the stream s, $\mathbf{P}_s$ is the transmit power matrix for the stream s, σ^2 is the additive white noise, and $\mathbf{h}_s^H$ is the Hermitian of the channel vector $\mathbf{h}_s$.

We define the user's minimum throughput constraint based on its transmission load and offered throughput. This ensures the model consistently meets the user's demand by allocating sufficient resources. This constraint is formulated as follows:

$$Load(u_i) \leq T_{offer}(u_i), \qquad \forall u_i \in \mathcal{U}. \tag{16}$$

The end-to-end (E2E) latency is critical for VR-CG applications [3GPP 2019]. The latency threshold must adapt to the video frame rate, as higher rates require more frames to be transmitted in a given time. While latency is a common bottleneck in many 6G applications, VR-CG is unique since uplink flows are classified as Ultra Reliable Low Latency Communication (URLLC) due to user game inputs, while downlink flows are classified as enhanced Mobile Broadband (eMBB) considering the transmission of high-resolution image data [3GPP 2023]. Given the importance of E2E latency in the VR-CG, we formulate a comprehensive calculation that includes all components between users and CNs. We represent the wireless propagation latency of VR-CG frame data between users' HMDs and BSs by considering L as the speed of light and $\mathbf{d}(u_i, b_l)$ as the distance between user $u_i \in \mathcal{U}$ and BS $b_l \in \mathcal{B}$. The wireless propagation latency is computed as:

$$\mathbf{P}(i) = \sum_{b_l \in \mathcal{B}} \left[y_l^i \frac{\mathbf{d}(u_i, b_l)}{LS} \right]. \tag{17}$$

Next, we define the transmission latency of VR-CG frame packets, where $\mathbf{s}(p)$ represents the video frame size in packets:

$$\mathbf{T}(i) = \left(\frac{\sum_{p \in \mathbf{p}(u_i)} \mathbf{s}(p)}{T_{Offer}(u_i)} \right). \tag{18}$$

We formulate the processing latency at the BSs, where $\rho(b_l)$ denotes the processing of BS $b_l \in \mathcal{B}$, as follows:

$$\mathbf{C}(i) = \sum_{b_l \in \mathcal{B}} \left[y_l^i \left(\frac{\sum_{p \in \mathbf{p}(u_i)} \mathbf{s}(p)}{\rho(b_l)} \right) \right]. \tag{19}$$

We calculate the queue latency at the BSs' buffers, where $\mathbf{a}_j$ represents the packet arrival rate of user $u_j \in \mathcal{U}$, as follows:

$$\mathbf{Q}(i) = \sum_{b_l \in \mathcal{B}} \left[y_l^i \left(\rho(b_l) - \sum_{u_j \in \mathcal{U}} \mathbf{a}_j \right)^{-1} \right]. \tag{20}$$

So far, the latencies discussed involve the wireless communication between users' HMDs and BSs. However, considering the CNs distributed in the network, we also need to account for the wired communication. Thus, we calculate the routing latency from BSs to CNs in the RAN topology, where $\mathbf{l}(p)$ is the latency of path $p \in \mathcal{P}_l^m$:

$$\mathbf{R}(i) = \sum_{b_l \in \mathcal{B}} \sum_{c_m \in \mathcal{C}} \left[y_l^i z_i^m \max_{p \in \mathcal{P}_l^m} (\mathbf{l}(p)) \right]. \tag{21}$$

Next, we consider the processing time used by the CN to render all objects in the user's FoV. We consider $\mathbf{r}(c_m)$ as the CN $c_m \in \mathcal{C}$ rendering speed in pixel per second (pps). The video frame rendering latency is calculated as:

$$\mathbf{M}(i) = \sum_{f \in \mathcal{F}} \sum_{c_m \in \mathcal{C}} \left[v_i^f z_i^m \left(\frac{\sum_{o_j \in \mathcal{O}} \Psi(j,k) f}{\mathbf{r}(c_m)} \right) \right] \tag{22}$$

We define the E2E latency requirement of users based on their video frame rate, similar to [Baena et al. 2024]. This E2E latency requirement represents the upper bound threshold for the video transmission of the user's VR-CG application:

$$E_i = \sum_{f \in \mathcal{F}} \left[v_i^f \left(\frac{1}{f} \right) \right]. \tag{23}$$

Finally, we define the E2E latency constraint of our optimization model as follows:

$$\mathbf{P}(i) + \mathbf{T}(i) + \mathbf{C}(i) + \mathbf{Q}(i) + \mathbf{R}(i) + \mathbf{M}(i) \leq E_i, \quad \forall u_i \in \mathcal{U}. \tag{24}$$

The VR-GX problem formulation is NP-hard, classified as a mixed-integer nonlinear programming (MINLP) problem. It includes binary decision variables for resource selection, such as assigning users to BSs and routing data to CNs. Nonlinear relationships, particularly in latency calculations involving summations and ratios, add to the complexity. This complexity demands a non-exact approach to quickly finding efficient solutions for practical scenarios.

Table 2 summarizes all the decision variables, sets, and data parameters used throughout the formulation.

Parameters	
$\mathcal{B}, \mathcal{C}, \mathcal{T}$ and E	Set of BSs, CNs, transport nodes and links
c_m^{CPU}, c_m^{Mem}, c_m^{Net} and c_m^{GPU}	CPU, memory, network, and GPU capacities of CNs
e_{ij}^{Cap}, e_{ij}^{Lat}	Transmission capacity and latency of link e_{ij}
$\mathcal{P}_l^m$ and $\mathcal{U}$	Set of k-shortest paths and set of users
$Res(u_i)$ and $FPS(u_i)$	Indicates users' HMD resolution and frame rate
$\mathcal{O}_i$, $\mathcal{R}$ and $\mathcal{F}$	Set of virtual objects, resolutions and frame rate
λ_i^j	Attention level of user $u_i \in \mathcal{U}$ to object $o_j \in \mathcal{O}_i$
Decision Variables	
$y_l^i \in \{0,1\}$	If user $u_i \in \mathcal{U}$ is admitted by BS $b_l \in \mathcal{B}$
$x_l^i \in \mathbb{Z}$	Amount of bandwidth allocated to user $u_i \in \mathcal{U}$ in BS $b_l \in \mathcal{B}$
$w_{i,j}^k \in \{0,1\}$	If resolution $r_k \in \mathcal{R}$ is selected to object $o_j \in \mathcal{O}_i$ for user $u_i \in \mathcal{U}$
$v_i^f \in \{0,1\}$	If frame rate $f \in \mathcal{F}$ is selected to user $u_i \in \mathcal{U}$
$z_m^i \in \{0,1\}$	If VR-CG application of user $u_i \in \mathcal{U}$ is runnint at CN $c_m \in \mathcal{C}$

Table 2. Model parameters and variables.

4. VR-GX Heuristic Algorithm

As discussed in Section 3, the VR-GX problem formulation is NP-hard, meaning that finding optimal solutions becomes computationally intractable as the problem size increases. While optimal solutions can be found for smaller instances, this type of solution is often not achievable for real-world scenarios where the number of users, network resources, and complexity of interactions can be much higher. In this context, searching for optimality is time-consuming and often impractical. Thus, the need for a fast approach to deliver efficient solutions in a reasonable amount of time becomes crucial.

To address this, we propose a heuristic algorithm that significantly reduces computation time while still providing high-quality solutions. Our heuristic effectively captures all relevant aspects of the VR-GX problem, ensuring that the key constraints and objectives are respected. It does so by strategically balancing between solution quality and computational efficiency, making it well-suited for practical scenarios where real-time decision-making is necessary. Moreover, by focusing on the most critical factors and avoiding exhaustive searches, our heuristic provides a robust solution that scales well with increasing user demands and network complexities. This makes it a valuable tool for scenarios with many users or limited resources, where achieving optimal solutions would otherwise be unfeasible.

Algorithm 1: VR-GX heuristic algorithm

```
1  Input: $\mathcal{B}, \mathcal{C}, \mathcal{P}_l^m, \mathcal{U}, \mathcal{O}_i, \mathcal{R}, \mathbb{S}, \lambda_i^j$
2  Output: VR-GX formulation feasible solution.
3  $\mathcal{R} \leftarrow \mathcal{R}$ in ascending order of quality;
4  $\mathcal{O}_i \leftarrow \mathcal{O}_i$ in descending order of attention;
5  $f_0 \leftarrow$ minimum frame rate;
6  Attributes for all objects of all users the minimum $r_k$ and $f_0$;
7  Associate users to BSs according to their SINR;
8  Position VR-CG applications of users in each BS at the closest CN;
9  for $u_i \in \mathcal{U}$ do
10     for $r_k \in \mathcal{R}$ do
11         for $o_j \in \mathcal{O}(u_i)$ do
12             $BW_{f_0} \leftarrow$ bandwidth demand to assign $r_k$ to $o_j$;
13             Calculate $\mathbf{G}(\mathcal{O}_i, r_k)$, $\mathbf{C}(\mathcal{O}_i, r_k)$, $\mathbf{M}(\mathcal{O}_i, r_k)$;
14             if $BW_{f_0}$, $\mathbf{G}(\mathcal{O}_i, r_k)$, $\mathbf{C}(\mathcal{O}_i, r_k)$, $\mathbf{M}(\mathcal{O}_i, r_k)$ is feasible then
15                 Assign new resolution $r_k$ to object $o_j$;
16                 Update available resources at $b_l \in \mathcal{B}$ and $c_m \in \mathcal{C}$;
17                 Update user served bandwidth to $BW_{u_i}$;
18             end
19         end
20     end
21     for $f \in \mathcal{F}$ do
22         $BW_f \leftarrow$ bandwidth demand to assign $f$ to $u_i$;
23         if $BW_f$ is feasible then
24             Assign new frame rate $f$ to user $u_i$;
25             Update available resources at $b_l \in \mathcal{B}$;
26             Update user served bandwidth to $BW_{u_i}$;
27         end
28     end
29 end
```

Algorithm 1 outlines our heuristic for efficiently solving the VR-GX problem. It takes as input the data defined in Section 2, and outputs a feasible resource allocation. Initially, available resolutions are sorted in ascending order, and objects in users' FoV are sorted by attention level (lines 3-4) to prioritize critical objects. Users are assigned to BSs based on SINR (line 7), and the VR-CG application is placed on the nearest CN (line 8).

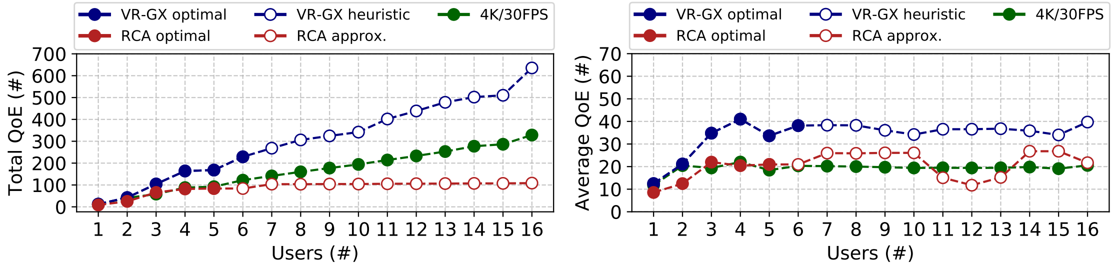
From line 9, the algorithm iteratively increases resolutions of each objects in user's

FoV, calculating bandwidth demand, based on Equations (13) - (24), and computational demand (line 13). If feasible, the new resolution is applied (lines 15-17). Then, starting at line 21, frame rates are adjusted. For each user, the algorithm checks bandwidth availability to increase frame rates (line 22-23) and updates accordingly (lines 24-26), optimizing object resolutions and frame rates.

The heuristic algorithm has a polynomial time complexity of $\mathcal{O}(|\mathcal{U}| \times |\mathcal{R}| \times |\mathcal{O}|)$, making it suitable for practical scenarios as it scales efficiently with the number of users, resolutions, and objects. This ensures computational feasibility even as problem size increases, allowing for quick decision-making in real-time. Unlike NP-hard problems, which become intractable at larger scales, this heuristic offers a practical solution framework adaptable to changing network conditions and user demands.

5. Evaluation

We evaluated the VR-GX problem formulation by comparing its optimal solutions with the approximated solutions from our heuristic algorithm and the solutions from the Rendering Capacity Allocator (RCA) by [Du et al. 2023a]. We simulate the RAN environment as in [Esper et al. 2023], featuring 475 BSs in a 4 km^2 area (2 km × 2 km), each with 100 MHz bandwidth. For evaluation, we selected a subset of four BSs based on user channel quality to represent a focused RAN configuration. The user attention level dataset is publicly accessible[2], and all implementations are available in an open GitHub repository[3]. The experiments were conducted using CPLEX Constraint Programming (version 22.11), which supports solving non-linear constraints. The implementation was executed in Python 3.10.12 with docplex 2.27.239 on a system equipped with an Intel® Core™ i7-1255U processor and 32 GB of RAM.



(a) Comparing total QoE.

(b) Comparing average QoE.

Figure 2. Comparing users' QoE for each scenario.

Figure 2 compares the total and average QoE achieved by each model, including the VR-GX optimal and heuristic solutions, the RCA optimal model [Du et al. 2023a], and a fixed fair allocation baseline proposed in [Huawei 2018], in which all users receive 4K/30 FPS. Due to the NP-hard nature of the problem, we limited the run time to one hour for both the VR-GX and RCA optimal models. Within this time frame, the VR-GX optimal model achieved optimal solutions for up to 6 users, while the RCA optimal model reached optimality for up to 5 users, providing best-approximated solutions (RCA Approx.) for larger scenarios.

[2] https://github.com/HongyangDu/AttentionQoE

[3] https://github.com/LABORA-INF-UFG/paper-GJLK-2025

Figure 2a demonstrates that the VR-GX model consistently outperforms the RCA model and the 4K/30 FPS baseline across all scenarios. This advantage is attributed to the VR-GX models' ability to incorporate video frame rate into QoE calculations, as defined by 3GPP standards [3GPP 2019, 3GPP 2023, 3GPP 2022]. Additionally, the RCA model achieves a lower QoE than the fixed 4K/30 FPS allocation due to its resource prioritization strategy, which favors users with better channel quality at the expense of overall QoE. This resource allocation imbalance negatively impacts the total QoE, further highlighting the advantages of the VR-GX approach in balancing fairness and maximizing QoE.

Figure 2b illustrates the average QoE per user for each model. The VR-GX optimal and heuristic consistently achieve higher average QoE for admitted users across all scenarios, due to its capacity of balance resource allocation while meeting user-specific QoE requirements. In contrast, the RCA model demonstrates lower average QoE, especially in larger scenarios where resource constraints lead to some users being denied admission to the network. Furthermore, due to RCA formulation user prioritization strategy, in some scenarios 4K/30 FPS outperforms RCA model in average QoE. This occurs since the RCA formulation may admit only users with better channel condition, reducing the overall fairness and average QoE of its solutions.

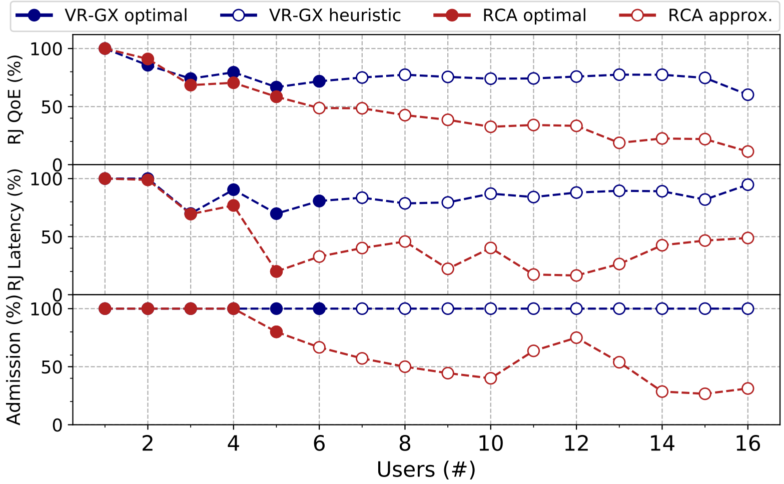


Figure 3. Comparing QoE and latency fairness, and the number os users admitted by each solution.

Figure 3 highlights the significant advantages of our VR-GX optimal and heuristic solutions over the RCA model [Du et al. 2023a] in terms of QoE and latency fairness, and user admission. To assess fairness, we employ *Raj Jain*'s fairness index (RJ QoE and RJ latency), with a user being considered admitted to the network when both computing and communication resources are allocated. In scenarios with up to 4 users, the RCA model successfully admits all users while maintaining fair QoE and latency. However, as the user count increases, the RCA model begins to prioritize users with better channel quality, boosting their QoE and reducing latency, while sidelining others. This trade-off, driven by its focus on minimizing latency, results in a significant degradation in QoE and fairness, particularly when resources become limited. In contrast, both the VR-GX formulation and our heuristic consistently achieve higher QoE and fairness while admitting all users. This demonstrates the superior performance of threshold-based resource allocation in VR-

CG applications, where maintaining fairness and meeting user demands is essential for delivering a high-quality experience.

Figures 4 and 5 present results related to bandwidth resource usage and execution time for each approach, including the VR-GX model and the RCA model [Du et al. 2023a]. Figure 4 illustrates the bandwidth usage of both models across all scenarios. The RCA model [Du et al. 2023a], which prioritizes maximizing user throughput, consistently allocates the maximum available bandwidth to each admitted user. For example, with a single user, the RCA model allocates the entire bandwidth of a BS to one user, resulting in resource over-provisioning. Similarly, when handling 4 users, the RCA model activates all BSs, allocating their full bandwidth to each user. In all scenarios, the RCA model consumes more bandwidth than the VR-GX model. In particular, while our model does not explicitly aim to minimize resource usage, it ensures that only the essential communication thresholds are met. This results in a more efficient resource allocation, avoiding the excessive over-provisioning observed in the RCA model.

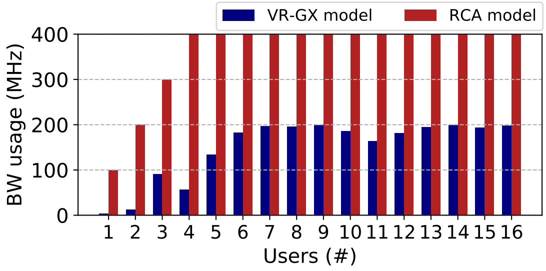


Figure 4. Bandwidth resource usage.

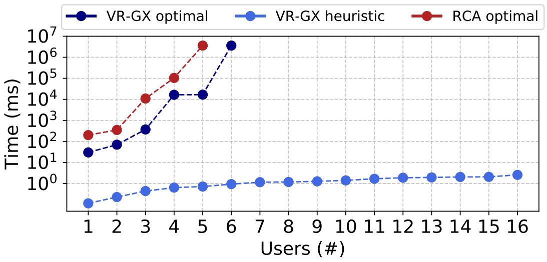


Figure 5. Runtime of each approach.

Figure 5 illustrates the runtime of each approach, the VR-GX optimal model, the VR-GX heuristic, and the RCA optimal model [Du et al. 2023a]. To facilitate comparison, the runtime is plotted on a logarithmic scale. Both optimization models require significant computational time to find solutions, particularly when ensuring the optimality of their output. In contrast, our heuristic algorithm consistently finds high-quality approximate solutions (as shown in Figure 2) in under 4 milliseconds across all scenarios. Additionally, our VR-GX heuristic finds the optimal solution for scenarios with up to 6 users. This result highlights the practical efficiency of our heuristic approach, making it better suited for real-world applications. Meanwhile, the VR-GX optimal solutions provide valuable baselines for assessing the quality of the heuristic solutions.

6. Conclusion and Future Work

This study presents a novel VR-GX formulation that effectively addresses the challenges of resource allocation in VR-CG applications, demonstrating better solutions in terms of QoE and latency fairness compared to existing approaches. Our heuristic algorithm provides a practical solution for managing the complexities of user demands and network constraints, making it suitable for real-time applications with large user bases. Future work will focus on refining the heuristic algorithm, exploring machine learning solutions for dynamic resource allocation, conducting practical experiments to collect Mean Opinion Score feedback for comparison with the proposed QoE model, and extending the framework to account for user mobility and dynamic network conditions.

Acknowledgments

This work was supported by CAPES, MCTIC/CGI.br/FAPESP through the SAMURAI project under Grant 2020/05127-2, RNP/MCTI through the Brasil 6G project under Grant 01245.020548/2021-07, and the OpenRAN@Brasil program.

References

3GPP (2019). "3rd Generation Partnership Project; Technical Specification Group Service and System Aspects; Study on Network Controlled Interactive Service (NCIS) in the 5G System (Release 17)". Technical Report TR 22.842 V17.2.0, 3GPP.

3GPP (2022). "3rd Generation Partnership Project; Technical Specification Group RAN; Study on XR evaluations for NR (Release 17)". Technical Report TR 38.838 V17.0.0, 3GPP.

3GPP (2023). "3rd Generation Partnership Project; Technical Specification Group Service and System Aspects; XR in 5G (Release 18)". Technical Report TR 22.842 V18.0.0, 3GPP.

Alhilal, A., Wu, Z., Tsui, Y. H., and Hui, P. (2024). FovOptix: Human Vision-Compatible Video Encoding and Adaptive Streaming in VR Cloud Gaming. In *ACM Multimedia Systems Conference*, MMSys '24, page 67–77, New York, NY, USA.

Baena, C., Fortes, S., Penaherrera-Pulla, O. S., Baena, E., and Barco, R. (2024). Gaming in the Cloud: 5G as the Pillar for Future Gaming Approaches. *IEEE Communications Magazine*, pages 1–7.

Chaccour, C., Saad, W., Debbah, M., Han, Z., and Poor, H. V. (2024). Less Data, More Knowledge: Building Next Generation Semantic Communication Networks. *IEEE Communications Surveys and Tutorials*, pages 1–1.

Du, H., Liu, J., Niyato, D., Kang, J., Xiong, Z., Zhang, J., and Kim, D. I. (2023a). Attention-aware resource allocation and qoe analysis for metaverse xurllc services. *IEEE Journal on Selected Areas in Communications*, 41(7):2158–2175.

Du, H., Wang, J., Niyato, D., Kang, J., Xiong, Z., Shen, X., and Kim, D. I. (2023b). Exploring Attention-Aware Network Resource Allocation for Customized Metaverse Services. *IEEE Network*, 37(6):166–175.

Esper, J. P. et al. (2023). Impact of User Privacy and Mobility on Edge Offloading. In *IEEE International Symposium on PIMRC*, pages 1–6.

Huawei (2018). "Huawei; iLab; Cloud VR Network Solution White Paper (2018)". Technical report, Huawei.

Shokrnezhad, M., Yu, H., Taleb, T., Li, R., Lee, K., Song, J., and Westphal, C. (2024). Towards a Dynamic Future with Adaptable Computing and Network Convergence (ACNC).

Tse, D. and Viswanath, P. (2005). *Fundamentals of wireless communication*. Cambridge university press.

Xia, L., Sun, Y., Liang, C., Feng, D., Cheng, R., Yang, Y., and Imran, M. A. (2023). WiserVR: Semantic Communication Enabled Wireless Virtual Reality Delivery. *IEEE Wireless Communications*, 30(2):32–39.