Characterizing the Network Traffic of Virtual Reality Cloud Gaming Applications

Vinicius R Oliveira, Daniel Henriques C. M. Soares, Daniel F. Macedo
Computer Science Department
Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte, Brazil
{viniciusr01@ufmg.br, daniel.henriques@dcc.ufmg.br, damacedo@dcc.ufmg.br}

***Abstract*—Cloud Gaming with Virtual Reality (VR) imposes strict requirements on bandwidth and latency to ensure Quality of Experience (QoE). However, current literature often focuses on traditional 2D gaming or older VR hardware. This paper presents a flow-level measurement study of Cloud VR traffic using a Meta Quest 3 headset and the Steam Link platform running a AAA title (*Half-Life: Alyx*). Our analysis reveals that the service utilizes a unified QUIC tunnel for both video and control data. We observed a distinct asymmetry in traffic patterns: the downlink behaves as a bursty flow driven by video fragmentation, while the uplink maintains a steady, periodic flow for telemetry and tracking. Furthermore, we systematically analyzed the impact of environment topology and player behavior on network load. Statistical results confirm that visual complexity (scenario) is the primary determinant of average throughput, whereas the player profile (exploratory vs. objective-oriented) significantly impacts flow variance and stability. These findings provide empirical guidelines for capacity planning and QoS strategies in immersive streaming services.**

***Index Terms*—Cloud Gaming, Virtual Reality, Traffic Characterization, Network Measurement, Meta Quest 3.**

I. Introduction

Virtual Reality (VR) and Augmented Reality (AR) have become widely adopted technologies applied across multiple domains, including movies, gaming, education, healthcare, and tourism [1]–[4]. According to the International Data Corporation (IDC) in Worldwide Augmented and Virtual Reality Spending Guide, global spending on these technologies is expected to reach approximately USD 8.4 billion by 2029 [5].

Among these applications, gaming stands out as one of the main drivers of VR and AR adoption, providing immersive and interactive experiences that demand high computational and network resources [6]. To meet such requirements, the gaming industry has increasingly relied on cloud gaming services, which allow users to play high-end games remotely without the need for local processing power. In this paradigm, the game runs entirely on remote servers, and the rendered frames are streamed to the player's device in real time.

However, despite its advantages, cloud gaming introduces significant challenges in terms of network performance. Parameters such as latency and jitter impact the user's Quality of Experience (QoE) [7]. These issues become even more critical in VR-based cloud gaming, where any degradation in frame rate or delay in motion feedback can lead to discomfort or motion sickness [8]. Consequently, understanding and characterizing the network behavior of VR cloud gaming traffic is essential for improving service reliability and user satisfaction.

Cloud gaming with VR requires a head-mounted display (HMD), rather than traditional joysticks or mouse-keyboard setups, and the literature on this modality remains comparatively sparse given its stricter performance envelope [6], [9], [10]. VR streams must render two high-resolution views, one per eye, at elevated refresh rates (72 Hz or higher) to sustain immersion, which tightens latency bounds well beyond those of conventional cloud gaming [10]. Disruptions in this pipeline increase the motion-to-photon delay (the time from a person moving to seeing the consequence of movement on the HMD) and de-synchronizes vestibular and visual cues, raising the risk of simulator sickness; hence, low end-to-end latency and tightly bounded jitter are first-order requirements [9].

Meeting these constraints also amplifies server-side compute load and bandwidth demand, since the system must encode and deliver stereo frames at high frame rates, while maintaining robustness under network variability [10]. Accessibility remains a concern as well: specialized HMD hardware and high-speed connectivity limit adoption and inclusivity [9]. Quantitatively, non-VR cloud gaming typically targets $< 50ms$ (tolerating up to 100 ms), 10–50 Mbps throughput, and $< 1\%$ loss, whereas VR pushes toward $< 20ms$ (ideally $< 10ms$), 50–100 Mbps, and $< 0.1\%$ loss—benchmarks reflected in 3GPP and ITU-T guidance for interactive media services [11], [12]. These stringent envelopes motivate our empirical study with real devices and instrumentation, and frame the QoS/QoE requirements summarized in Table I.

TABLE I: QoS Requirements for Cloud Gaming

Parameter	Cloud Gaming (Non-VR)	Cloud Gaming with VR	Source
Latency	<100 ms (ideal: <50 ms)	<20 ms (ideal: <10 ms)	3GPP TR 22.841 and ITU-T G.1011
Bandwidth	10-50 Mbps	50-100 Mbps	3GPP TS 22.261 and ITU-T Y.3071
Packet Loss	<1%	<0.1%	3GPP TR 23.791 and ITU-T G.1011
Reliability	99.9%	99.999%	3GPP TR 22.841 and 3GPP TR 23.791

Recent traffic and measurement studies in cloud gaming focus predominantly on non-VR services, dissecting bandwidth consumption, latency/queue control, and adaptation behavior

ISBN: 978-3-903176-79-9

under network stress. Representative work includes packet-level characterization of Google Stadia (RTP/WebRTC over UDP, bitrate dynamics), anatomy/measurements of GeForce NOW at scale, and controlled experiments of Stadia/GeForce NOW/Luna under competing traffic, all centered on traditional setups rather than VR streaming [13]–[15]. By contrast, the VR cloud-gaming literature is smaller and skews toward QoS under impairment or streaming feasibility over Wi-Fi, with limited attention to how gameplay patterns or player behavior shape network load.

We conduct a comprehensive measurement study using a Meta Quest 3 and the AAA title *Half-Life: Alyx*. Our main contributions include: **(i)** a flow-level characterization of the Steam Link protocol, identifying the usage of a unified QUIC tunnel with distinct uplink/downlink asymmetries; and **(ii)** a statistical analysis demonstrating that while visual complexity determines average bandwidth consumption, player behavior significantly impacts flow variance and jitter. The remainder of this paper reviews related work (Section II), details the methodology (Section III), presents the traffic analysis (Sections IV and V), and concludes the study (Section VI).

II. Related Work

In this section, we review the state-of-the-art to contextualize the specific challenges of next-generation Cloud VR. We categorize the related literature into three main areas: **(i)** the extensive characterization of traditional 2D cloud gaming platforms; **(ii)** the emerging studies on Cloud and Edge VR transmission; and **(iii)** approaches for traffic classification and machine learning in network flows. Finally, we summarize the gaps in current literature that necessitate a flow-level analysis of modern hardware like the Meta Quest 3.

A. Characterization of 2D Cloud Gaming

The network behavior of traditional (2D) cloud gaming platforms has been extensively studied. Lyu et al. [14] conducted a large-scale measurement of NVIDIA GeForce NOW, characterizing the network anatomy of sessions across various user setups. They proposed methods to estimate QoE metrics such as frame rate and resolution based on encrypted traffic patterns. Similarly, Wang et al. [16] emphasized that "games are not equal", classifying traffic based on gameplay contexts (e.g., active vs. idle states) to refine user experience measurements. However, these studies focus exclusively on 2D monoscopic video streams. VR gaming introduces significantly more complex constraints, including stereo rendering (two views), 6-DoF input tracking, and a stricter motion-to-photon latency threshold required to prevent cybersickness, which are not addressed in traditional cloud gaming literature.

B. Cloud and Edge Virtual Reality

The literature specifically targeting VR cloud gaming is narrower. One of the foundational studies by Zhao et al. [6] performed a "reality check" on cloud VR traffic using the Meta Quest 2. They modeled frame sizes and inter-arrival times for games like *Beat Saber*. While pioneering, their study relied on previous-generation hardware (1832×1920 pixels per eye) and simpler game workloads. In contrast, our work utilizes the Meta Quest 3 (2064×2208 pixels per eye) running *Half-Life: Alyx*, a high-fidelity AAA title that imposes significantly higher computational and network loads.

TABLE II: Comparison of related works and this study

Reference	Platform/Context	Primary Focus	Limitation addressed by this work
[14]	GeForce NOW	Anatomy of 2D Cloud Gaming	Lacks VR-specific constraints (stereo/6DoF)
[17]	Edge / Unity	WebRTC/Wi-Fi 6 Performance	Focus on Edge/LAN; different protocol stack
[18]	Synthetic	In-network P4 Classification	Uses synthetic traffic; no user behavior analysis
[6]	Meta Quest 2	Cloud VR Traffic Modeling	Older hardware generation; less demanding games
This Work	**Meta Quest 3**	**Cloud VR Analysis w/ Half-Life: Alyx**	**Next-gen resolution; Flow-level separation**

Other works have focused on transmission protocols for VR. Casasnovas et al. [17] evaluated Edge VR streaming over Wi-Fi 6 using Unity Render Streaming. They identified distinct traffic patterns, such as the batching of video frames in WebRTC, which affects jitter and airtime consumption. While they provide valuable insights into protocol behavior (WebRTC/RTP), their focus remains on edge streaming and transport layer dynamics rather than the full characterization of WAN-based Cloud VR traffic during immersive gameplay.

C. Traffic Classification and Machine Learning

Regarding traffic identification, Shirmarz et al. [18] proposed an in-network machine learning approach using P4 programmable switches to classify AR and Cloud Gaming traffic. Their work successfully extracts RTP features to distinguish these flows from background traffic. However, their evaluation relies on synthetic traffic generation and adheres to ITU-T baselines rather than live, user-driven sessions. Our work complements this by providing empirical, real-world traffic traces captured during active VR sessions, which capture the stochastic nature of human interaction and game engine variability that synthetic models often miss.

D. Summary of Contributions

Overall, prior works either focus on 2D gaming, rely on synthetic datasets, or analyze older VR hardware. Our contribution extends the state-of-the-art by performing a fine-grained, flow-level characterization of next-generation Cloud VR. We explicitly separate uplink (tracking data) and downlink (video/audio) dynamics, providing updated insights into the throughput variability and protocol utilization of modern immersive cloud gaming.

III. Methodology

This section details the experimental framework used to assess VR cloud gaming network performance. We describe the testbed architecture, the specific experimental scenarios designed to vary computational and network loads, and the data acquisition workflow.

A. Testbed Architecture

To ensure reproducibility and adherence to strict VR motion-to-photon latency constraints, we established a controlled local testbed, as illustrated in Figure 1. The architecture comprises three main components:

- **Server Side:** The rendering server is a workstation equipped with an Intel Core i7-7700 CPU, 16 GB of DDR4 RAM, and an NVIDIA RTX 3060 GPU, running Windows 10. It hosts the *SteamVR* runtime [19] to manage the computational workload and game rendering.
- **Client Side:** We utilize a Meta Quest 3 headset acting as the streaming client. The device runs the *Steam Link* application to decode the incoming video stream (up to dual 1440p resolution at 72 fps for HL:Alyx) and transmit user input events back to the server.
- **Network Topology:** Connectivity is provided by a Ubiquiti Wi-Fi 5 (802.11ac) access point, bridging the server and the VR headset. Although the Meta Quest 3 supports Wi-Fi 6E, we deliberately utilized a Wi-Fi 5 environment to represent a common "baseline" for residential deployments. This setup isolates the traffic to minimize external interference while allowing for precise packet-level monitoring. Furthermore, assessing the protocol over 802.11ac exposes the application's behavior under stricter bandwidth constraints and legacy contention mechanisms, providing a lower-bound reference for performance characterization.

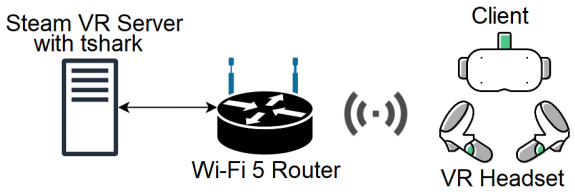


Fig. 1: Experimental testbed configuration for the VR Cloud Gaming service.

B. Experimental Scenarios and Design

We selected *Half-Life: Alyx* as the reference application due to its status as a benchmark for **AAA titles**[1]. In the gaming industry, "AAA" denotes high-budget productions featuring top-tier graphical fidelity and complex mechanics, which consequently impose significant demands on rendering resources and network bandwidth. To capture a comprehensive range of network traffic patterns, we defined two independent variables: *Environment Topology* and *Player Behavior*.

1) Environment Topology: Experiments were conducted across two distinct game chapters to introduce variability in visual complexity and rendering load. Throughout the analysis, we refer to these scenarios as the **Hotel map** and the **Zoo map**:

- **Scenario A (Hotel map):** Based on the *Northern Star* chapter, this environment is predominantly indoors. It is characterized by narrow corridors, staircases, and artificial lighting inside a hotel structure. The topology emphasizes short-range navigation and fine-grained camera adjustments, resulting in frequent occlusion events.
- **Scenario B (Zoo map):** Based on the *Captivity* chapter, this scenario takes place in and around an abandoned zoological complex. In contrast to the Hotel map, the Zoo map features open courtyards and long-distance rendering. It introduces dynamic combat sequences with multiple enemies, increasing both the rendering workload and the frequency of sudden camera movements.

2) Player Behavior Profiles: To assess the human factor in traffic generation, we established two distinct player profiles:

- **Exploratory Profile:** The user interacts deeply with the environment, focusing on narrative details and performing non-linear movements. This results in a more erratic viewport trajectory.
- **Objective-Oriented Profile:** The user focuses on mission completion efficiency, advancing rapidly through stages with linear movement and minimal environmental interaction.

C. Data Acquisition and Dataset

Network traffic was captured on the server-side interface using *Wireshark* and *TShark* tools. We recorded all packet exchanges, including transport and application layer data.

The data collection process involved two participants playing both the Hotel and Zoo maps under both behavioral profiles. Each map-profile combination was repeated three times, with capture sessions of 300 seconds. In total, this resulted in 12 gameplay traces (2 players $\times$ 2 maps $\times$ 3 repetitions), totaling 3,600 seconds of pure gameplay traffic.

IV. Traffic Characterization

The initial step of this study focused on analyzing the communication anatomy between the server and the client. The main goal was to understand the network behavior throughout the entire game session, from initialization to termination. The traffic characterization was based on a temporal analysis of the captured flows, complemented by public data from the Steam platform and insights from related cloud gaming literature.

This section is structured into three parts: (i) communication overview, (ii) analysis of the downlink traffic, (iii) analysis of the uplink traffic.

A. Communication Overview

In our experimentation, the communication flow of a cloud gaming session follows a well-defined lifecycle, governed by the interaction between the local server, the VR headset, and the remote Steam infrastructure. We categorize this lifecycle into three phases: *Initialization*, *Gameplay*, and *Termination*.

1) Session Lifecycle: The session begins with the **Initialization** phase. Initially, the SteamVR software on the local server establishes a connection with Steam's backend (via TCP and TLS) to authenticate the user and synchronize cloud configuration data. Simultaneously, the Steam Link application

[1]Half-Life: Alyx remains the peak concurrent VR-only title on Steam, with over 42,000 simultaneous players and a 96% positive rating [20].

on the VR headset performs a local discovery broadcast. Once the server and headset identify each other, they execute a handshake to negotiate stream parameters and security keys.

Upon successful negotiation, the transition to the **Gameplay** phase occurs. This phase is characterized by a continuous, real-time interactive loop: the headset captures and transmits user inputs and telemetry (uplink) to the server; the server processes this data, renders the game frame, and transmits the encoded video and audio back to the headset (downlink).

Finally, the **Termination** phase is triggered when the user exits the application. The game process is halted, the media streams are torn down, and the control connections are gracefully closed, ending the capture.

2) Throughput and Temporal Analysis: Figure 2 illustrates the temporal evolution of throughput during this lifecycle. A logarithmic scale is employed on the y-axis to simultaneously visualize the signaling traffic (in the order of Kbps) and the high-bandwidth media streams (tens of Mbps).

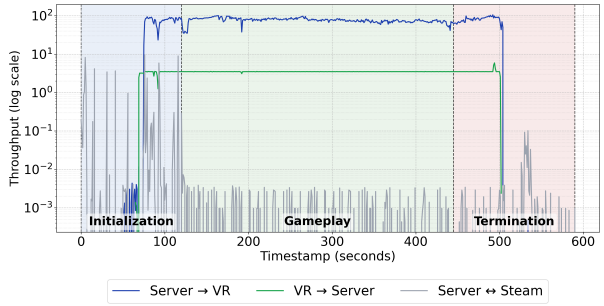


Fig. 2: Throughput evolution across the session (log scale).

Visually, the Initialization phase is marked by sporadic spikes in the *Server* $\leftrightarrow$ *Steam* flow (gray line), corresponding to the authentication interactions described above. A critical visual marker occurs around $t = 60$s: an abrupt "step function" surge in the *Server* $\rightarrow$ *VR* (blue) and *VR* $\rightarrow$ *Server* (green) flows. This jump of several orders of magnitude visually confirms the exact moment the heavy media stream is established after the initial handshake.

During gameplay, these flows stabilize into a plateau. The downlink hovers around 76–80Mbps, a range consistent with the bitrate targets for immersive applications defined by 3GPP TS22.261 and ITU-T Y.3071 [11], [12], while the uplink remains nearly constant, indicating that the network sustains a stable capacity without significant throttling.

Table III summarizes the traffic volume and dominant protocols. It shows that, although the Server-Steam traffic is active during session setup, its volume is negligible compared to the QUIC streams that sustain the gameplay.

3) Traffic Stability and Distribution: To further understand the nature of these flows, Figure 3 presents the Cumulative Distribution Function (CDF). This statistical view reveals a clear distinction in behavior between the uplink and downlink.

The **Uplink (VR $\rightarrow$ Server)** curve exhibits a near-vertical transition around 3–4 Mbps. This shape indicates a highly deterministic flow with low jitter, characteristic of a Constant Bit Rate (CBR) profile. This confirms that telemetry data is transmitted at a fixed frequency regardless of game activity.

TABLE III: Traffic summary: protocols, volume, and average throughput per phase (95% CI)

Flow	Phase	Protocols	Avg. Throughput (95% CI)	Packets (95% CI) (95% CI)
Server to VR	Initialization	QUIC, RTCP STEAMDISCOVER	[45.66 - 55.51] Mbps	[398.17 - 616.53] $\times 10^3$
	Gameplay	QUIC	[76.31 - 80.95] Mbps	[1767.37 - 2281.01] $\times 10^3$
	Termination	QUIC	[29.49 - 41.61] Mbps	[83.47 - 290.08] $\times 10^3$
VR to Server	Initialization	QUIC, STEAMDISCOVER	[2.65 - 3.52] Mbps	[31.70 - 43.24] $\times 10^3$
	Gameplay	QUIC	[3.49 - 3.51] Mbps	[119.79 - 158.52] $\times 10^3$
	Termination	QUIC	[3.17 - 3.51] Mbps	[4.25 - 22.49] $\times 10^3$
Server $\leftrightarrow$ Steam	Initialization	TLS 1.3, TCP, HTTP	[355.28 - 587.11] Kbps	[2.93 - 8.98] $\times 10^3$
	Gameplay	TLS 1.3, TCP	[0.87 - 1.71] Kbps	[0.1684 - 0.2720] $\times 10^3$
	Termination	TLS 1.3, TCP	[0 - 492.43] Kbps	[0 - 1.70] $\times 10^3$

In contrast, the **Downlink (Server $\rightarrow$ VR)** displays a more gradual S-curve, with a wider distribution range (median $\approx$ 80 Mbps). This reflects a Variable Bit Rate (VBR) behavior, inherent to real-time video encoding where the bit rate fluctuates based on the visual complexity of the rendered scene.

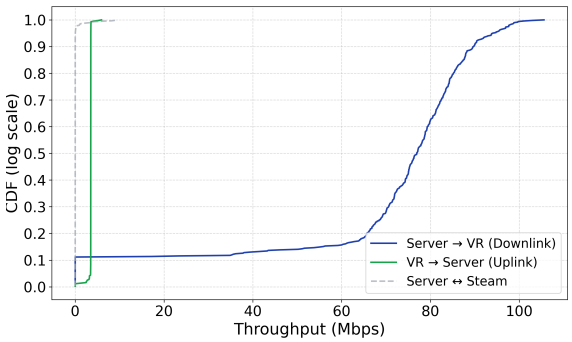


Fig. 3: CDF of throughput for the Uplink (CBR/Telemetry) and Downlink (VBR/Video).

While this overview characterizes the session as a whole, the distinct behaviors of the video stream and control traffic warrant a deeper inspection. The following subsections will provide a detailed analysis of the **Downlink** and **Uplink** directions, respectively.

B. Downlink: Server to VR Client

Downlink traffic constitutes the most critical component of the session due to the high bandwidth requirements of real-time video streaming. This section characterizes the downlink traffic focusing on three key aspects: transport architecture, transmission dynamics, and network reliability.

Figure 4 depicts the temporal distribution of network traffic throughout the gaming session. The x-axis represents time, while the y-axis indicates the active protocols.

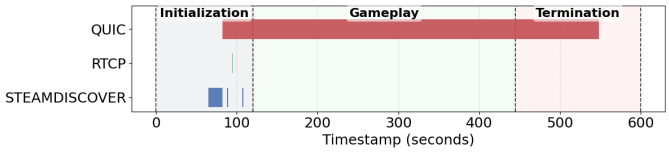


Fig. 4: Downlink: Server to VR Client

1) Transport Architecture: The header analysis of the package reveals that Steam Link employs a different transport strategy from conventional cloud gaming platforms reported in the literature. While recent studies show that NVIDIA GeForce NOW uses separate RTP streams over UDP for video and audio [14], and Google Stadia relies heavily on the standard WebRTC stack [13], our measurements indicate a unified approach.

Steam Link encapsulates video, audio, and telemetry data within a single encrypted tunnel via the QUIC protocol (identified on port 10400 in this experiment). Table IV summarizes the traffic composition. The predominance of QUIC ($\approx 99\%$ of the total volume) may be due to the modern features of this protocol, specifically user-space congestion control and Head-of-Line blocking prevention [21], which are essential to mitigate latency in unstable wireless environments.

Valve's GameNetworkingSockets GitHub repository states its reliability layer is based on QUIC, indicating QUIC-derived mechanisms in Valve's networking stack [22]. The repository does not date when this was adopted, to our knowledge, we are the first to explicitly note this QUIC-based lineage in Valve's streaming context.

TABLE IV: Downlink Traffic Summary by Protocol

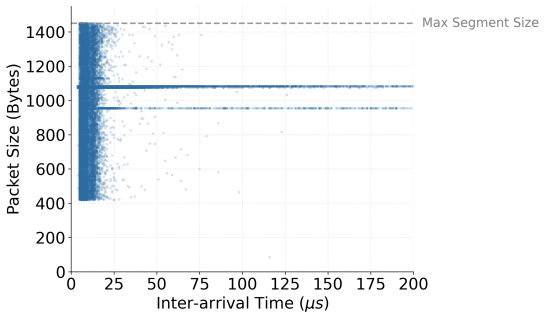
Protocol	Port	Packets (qty)	Volume (Gb)
QUIC	10400	3,861,595	≈ 25.35
RTCP	10400	298	0.002
STEAMDISCOVER	27036	32	< 0.001

2) Transmission Dynamics: To understand the server's frame fragmentation strategy, we analyzed the temporal relationship between packet size and Inter-Arrival Time (IAT), as shown in Figure 5.

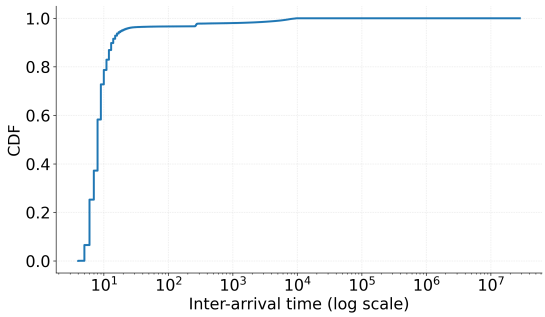
Figure 5a reveals a characteristic Packet Train behavior. We observe a dense concentration of packets saturating the Maximum Transmission Unit (MTU ≈ 1450 bytes) with an IAT near zero ($< 50\mu s$). This pattern is consistent with the fragmentation of large encoded video frames into bursts of MTU-sized packets sent back-to-back, a behavior described as packet trains in the traffic characterization literature.

This behavior differs from the uniform spacing observed in traditional low-latency encoders (e.g., VoIP traffic), resembling instead an aggressive pacing strategy. The underlying goal is to maximize instantaneous throughput to reduce the total transmission time of the frame, thereby minimizing rendering latency at the client side.

3) Reliability Analysis: Connection stability, a mandatory requirement for immersive applications, is evaluated through



(a) Packet Size vs. IAT.



(b) IAT CDF (log scale).

Fig. 5: Downlink traffic dynamics analysis via QUIC. (a) The scatter plot highlights the burst structure (Packet Trains) with MTU saturation. (b) The long tail in the latency distribution indicates the presence of outliers critical to VR stability.

the Cumulative Distribution Function (CDF) of inter-arrival times (Figure 5b). The analysis highlights two points:

- **Steady-State Efficiency:** The curve exhibits a sharp knee, with over 99% of packets arriving with an interval of less than $100\mu s$. This demonstrates the effectiveness of the QUIC protocol in maintaining a continuous, high-throughput flow under normal network conditions.
- **Long Tail Risks:** The logarithmic scale analysis exposes a long tail in the distribution, with outliers reaching the order of milliseconds ($10ms$ to $1s$). In the context of Virtual Reality, where motion-to-photon latency must remain strictly below 20ms to avoid vestibular discomfort and motion sickness [17], these outliers represent critical risks. Although statistically rare, these events correspond to stalls that disproportionately impact the Quality of Experience (QoE), momentarily breaking user immersion.

Collectively, these results seem to suggest that the Steam Link downlink strategy operates as a high-throughput, aggressively paced mechanism, possibly designed to minimize transmission delay. While the adoption of QUIC indicates that it may support this volume under stable conditions, the susceptibility to tail-latency events suggests the presence of an important trade-off: the system tends to maximize instantaneous delivery speed but may remain vulnerable to

network jitter. Thus, maintaining immersion may depend less on average bandwidth and more on the network's ability to mitigate these sporadic high-latency spikes.

C. Uplink: VR Client to Server

While the downlink traffic is driven by high-bandwidth video streaming, the uplink communication is defined by its stringent latency requirements for interactivity. This section characterizes the client-to-server traffic, focusing on the game controls, telemetry, and tracking data required to synchronize the user's physical actions with the virtual environment.

Figure 6 illustrates the temporal distribution of uplink traffic throughout the session. Similar to the downlink, the connection persists continuously, but with a significantly lower throughputs, primarily composed of high-frequency, small-payload packets.

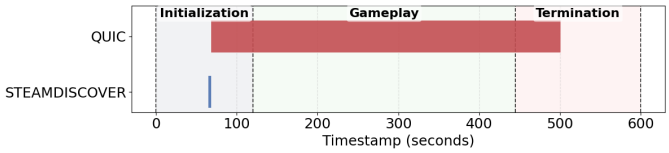


Fig. 6: Uplink: VR Client to Server Traffic over Time

1) Transport Architecture: The uplink transport strategy mirrors the unified approach observed in the downlink. Instead of segregating control messages from audio or telemetry using different protocols, Steam Link multiplexes all upstream data into the same encrypted QUIC tunnel.

Table V details the protocol composition. The traffic is almost entirely carried by QUIC on UDP port 10400. The *STEAMDISCOVER* protocol appears only transiently during the initial handshake (Discovery Phase). The total volume ($\approx$ 1.51 Gb) suggests that the uplink payload includes not only coordinate tracking but possibly dense sensor telemetry.

TABLE V: Uplink Traffic Summary by Protocol

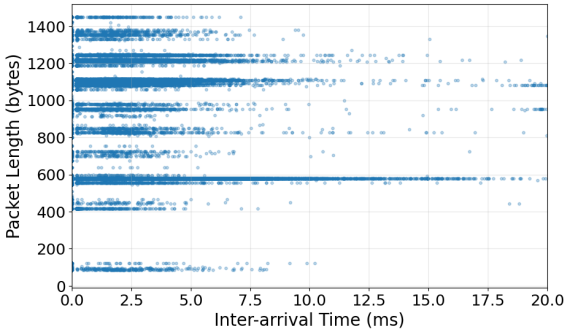
Protocol	Port	Packets (qty)	Volume (Gb)
QUIC	10400	271,397	1.51
STEAMDISCOVER	27036	3	< 0.001

2) Transmission Dynamics: The packet transmission dynamics in the uplink direction differ fundamentally from the video streaming behavior. To understand the client's transmission strategy, we analyzed the temporal relationship between packet size and Inter-Arrival Time (IAT). Figure 7 (a) presents this relationship, with the time axis strictly limited to the $0-20$ ms interval. This window was chosen to align with the critical "motion-to-photon" latency threshold in VR systems; packets arriving with an IAT exceeding this window typically result in input lag or discarded frames, rendering their micro-dynamics less relevant for steady-state analysis.

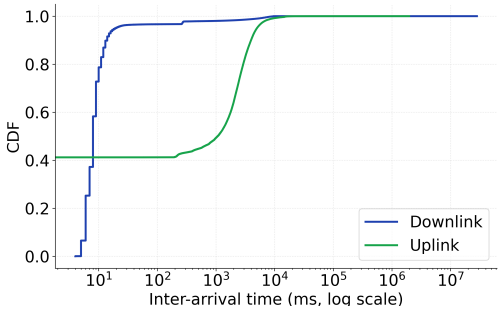
Unlike the downlink, which saturates the MTU with packet trains, the uplink scatter plot reveals a multi-modal distribution of packet sizes, characterized by distinct horizontal quantization bands. We observe three primary "peaks" in payload size:

- **Small Control Signals (≈ 100 B):** Likely associated with protocol keep-alives or acknowledgment packets.
- **Telemetry Payloads (≈ 580 B):** The dense concentration in this band suggests it is the standard container for inertial measurement unit (IMU) data and positional tracking coordinates, sent at the sensor's polling rate.
- **Rich Input Data ($\approx 1100-1350$ B):** The larger packets likely encapsulate aggregated sensor data or simultaneous audio streams from the VR headset's microphone.

Furthermore, the IAT distribution within this 20 ms window is more dispersed compared to the downlink's near-zero bursts. This confirms an *isochronous* transmission pattern, where packets are generated at regular intervals due to the hardware sensors rather than by video frames.



(a) Packet Size vs. IAT (Uplink). The x-axis is truncated at 20 ms to highlight jitter within the operational latency window.



(b) IAT CDF.

Fig. 7: Uplink traffic dynamics analysis. (a) The scatter plot shows size quantization corresponding to specific input data structures. (b) The CDF (green curve) shows a smoother, sigmoidal distribution compared to the sharp burstiness of the downlink (blue curve).

3) Reliability Analysis: The stability of the uplink flow is examined through the Cumulative Distribution Function (CDF) of inter-arrival times, represented by the green curve in Figure 7 (b). The comparison with the downlink (blue curve) highlights two distinct behaviors:

- **Cadenced Flow vs. Burstiness:** While the downlink CDF rises vertically at $< 100\mu s$ (indicating packets

arriving back-to-back), the uplink CDF (green) follows a smoother, sigmoidal curve shifted to the right. This confirms that uplink packets are less prone to buffering or bursting; instead, they flow in a steady, paced rhythm essential for maintaining real-time synchronization.

- **Consistency within Tolerance:** The absence of an extremely sharp "knee" in the uplink curve indicates a natural variance in arrival times, likely due to the internal processing time of sensor fusion on the headset before transmission. However, the vast majority of packets still arrive well within the sub-millisecond range, ensuring that the server receives inputs fast enough to render the next frame without perceptible delay.

In summary, the Steam Link uplink strategy prioritizes the consistent, quantized delivery of sensor data over raw throughput. The use of QUIC facilitates this by preventing Head-of-Line blocking, ensuring that a lost input packet does not delay subsequent, more current tracking updates.

V. Traffic Differentiation Based on Player Profile and Scenario

Recent studies indicate that traffic in cloud gaming and Virtual Reality (VR) applications is not uniform, but rather shaped by the dynamic interaction between environmental complexity and user behavior. Lyu et al. [14] demonstrate that visual density and scene transitions significantly alter throughput patterns, suggesting that maps with distinct visual characteristics produce unique traffic signatures. Complementarily, Wang et al. [16] and Zhang et al. [23] provide evidence that the player's activity level (e.g., idle, passive, or active) and head movement directly influence temporal variability and the size of generated frames. Based on these premises, this section empirically investigates whether the structural differences between the *Zoo* and *Hotel* maps, combined with player profiles (*Objective-oriented* vs. *Exploratory*), result in distinguishable network patterns.

A. Descriptive and Visual Analysis

For this analysis, the collected data was aggregated into 1-second windows. Given the consistency observed across experiment repetitions, data from the three sessions of each scenario were consolidated to provide a more robust view. Table VI presents the mean and standard deviation for throughput and packets per second (PPS) for each combination.

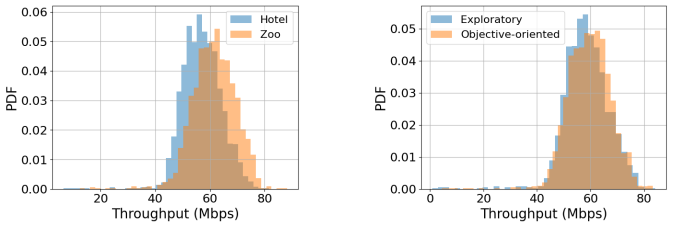
TABLE VI: Consolidated statistics (Mean ± Standard Deviation) for throughput and PPS by scenario and profile.

Map	Profile	Avg. T-put (Mbps)	Max. T-put (Mbps)	Avg. PPS (pps)	Max. PPS (pps)
Hotel	Objective	56.68 ± 7.2	77.57	6573 ± 810	8992
Hotel	Exploratory	56.59 ± 9.4	78.35	6563 ± 1100	9078
Zoo	Objective	62.94 ± 7.7	87.84	7295 ± 880	10178
Zoo	Exploratory	59.54 ± 9.8	83.90	6904 ± 1150	9726

Results indicate that the **scenario** may be the primary factor influencing traffic intensity. The *Zoo* map presented consistently higher throughput averages (≈60–63 Mbps) compared to the *Hotel* map (≈56 Mbps), a pattern replicated in the packet rate. We believe this difference can be partially attributed to the characteristics of modern video codecs (e.g., H.264/H.265), which rely on spatial redundancy for compression. The *Hotel* map features a homogeneous color palette and repetitive textures (gray walls, similar corridors), which optimizes intra-frame compression. In contrast, *Zoo* presents high visual entropy, with dynamic transitions between narrow corridors, illuminated laboratories, and open areas with vegetation. This visual diversity requires a lower compression ratio to maintain image quality, consequently raising the bitrate. Figure 8a illustrates this behavior through the Probability Density Functions (PDFs), where the *Zoo* map curve is shifted to the right compared to *Hotel* map, indicating higher bandwidth.

Regarding the **player profile**, a distinct effect related to flow **stability** is observed. *Objective-oriented* players tend to generate traffic with lower variance (smaller standard deviations in Table VI). We believe that, by performing linear movements focused on the objective, these players allow the codec to utilize inter-frame prediction efficiently, as motion vectors remain consistent.

Conversely, the *Exploratory* profile should generate complex and less predictable motion vectors. This can reduce the efficiency of temporal prediction and increases the size of residual frames (P-frames), resulting in greater instantaneous traffic fluctuation. Figure 8b corroborates this analysis: although the throughput is close between profiles, the distribution for the *Exploratory* profile is flatter, indicating higher variability and the occurrence of peaks.



(a) Comparison by Map (Impact on Volume)

(b) Comparison by Player (Impact on Variance)

Fig. 8: Probability Density Function (PDF)

B. Exploratory Statistical Analysis and Discussion

The application of a Two-Way ANOVA statistically indicated that the **Map** is the primary determinant of average throughput ($F(1, 8) = 14.97$, $p = 0.0047$). This result empirically validates the observations by Lyu et al. [14], suggesting that visual density and scenario complexity dictate traffic volume. This occurs because heterogeneous scenarios, such as the *Zoo* map, limit the spatial compression capabilities of codecs, requiring higher bandwidth regardless of user action.

In contrast, the **Player** factor ($p = 0.182$) and the interaction between factors ($p = 0.201$) did not show statistical significance, which disagrees with the findings of Wang et

al. [16] and Zhang et al. [23]. Although Wang et al. identified differences between activity states (idle vs. active), our study indicates that, within an active state, different *playstyles* (*Objective-oriented* vs. *Exploratory*) do not significantly alter average bandwidth consumption.

However, aligning with what Zhang et al. [23] observed regarding head movement, our data shows that exploratory behavior impacts flow **stability**. The higher variance recorded in this profile suggests that unpredictable movements degrade the codec's temporal prediction, introducing fluctuations that increase the jitter.

Therefore, capacity planning for cloud-based VR services should prioritize the analysis of game visual complexity to estimate average demand, while Quality of Service (QoS) and buffering strategies should consider the behavioral profile of players to mitigate instantaneous latency variations.

VI. Conclusion

This paper presented a flow-level characterization of Cloud VR traffic using the Meta Quest 3. Our analysis reveals that the service relies on a unified QUIC tunnel exhibiting distinct flow asymmetry: a bursty, bandwidth-intensive downlink driven by video fragmentation, and a highly isochronous uplink critical for tracking. Statistically, we demonstrated that visual scenario complexity is the primary determinant of average throughput, whereas player behavior predominantly influences flow stability and jitter. These findings indicate that effective Cloud VR resource management must distinguish between content complexity and user interaction styles.

We acknowledge limitations in our experimental design. First, Wi-Fi 5 does not employ advanced Wi-Fi 6/7 features. Furthermore, this is an isolated testbed, lacking the variable jitter typical in WAN conditions. Second, the statistical analysis involves a limited number of participants; therefore, the observed behavioral effects on flow stability should be interpreted as preliminary trends rather than generalizable user models. Future work should include larger-scale studies under Wi-Fi 6/7 and WAN scenarios to assess the robustness and transferability of our findings.

Acknowledgements

This work was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, CNPq (funding agency from the Brazilian federal government), FAPEMIG (Minas Gerais State Funding Agency), and São Paulo Research Foundation (FAPESP) with Brazilian Internet Steering Committee (CGI.br), grants 2018/23097-3 and 2020/05182-3.

References

[1] U. A. Chattha, U. Janjua, F. Anwar, M. Madni, M. Cheema, and S. Janjua, "Motion sickness in virtual reality: An empirical evaluation," *IEEE Access*, vol. PP, pp. 1–1, 07 2020.

[2] H. T. Crogman, V. D. Cano, E. Pacheco, R. B. Sonawane, and R. Boroon, "Virtual reality, augmented reality, and mixed reality in experiential learning: Transforming educational paradigms," *Education Sciences*, vol. 15, no. 3, 2025.

[3] Y. Yazarkan, G. Sonmez, M. E. Gurses, M. Ucdal, and C. Simsek, "Virtual reality and augmented reality use cases in gastroenterology," *Current Gastroenterology Reports*, vol. 27, 02 2025.

[4] X. Fan, X. Jiang, and N. Deng, "Immersive technology: A meta-analysis of augmented/virtual reality applications and their impact on tourism experience," *Tourism Management*, vol. 91, p. 104534, 2022.

[5] IDC, "EMEA's AR/VR spending to reach US$ 8.4 billion by 2029," https://my.idc.com/getdoc.jsp?containerId=prEUR153526825, 2025.

[6] S. Zhao, H. Abou-Zeid, R. Atawia, Y. S. K. Manjunath, A. B. Sediq, and X. P. Zhang, "Virtual reality gaming on the cloud: A reality check," in *Proceedings - IEEE Global Communications Conference, GLOBECOM*, 2021.

[7] D. Soares, M. Carvalho, and D. F. Macedo, "Enhancing cloud gaming qoe estimation by stacking learning," *J. Netw. Syst. Manage.*, vol. 32, no. 3, 2024.

[8] E. Chang, H. T. Kim, and B. Yoo, "Virtual reality sickness: A review of causes and measurements," *International Journal of Human-Computer Interaction*, vol. 36, no. 17, pp. 1658–1682, 2020.

[9] K.-L. Chan, K. Ichikawa, Y. Watashiba, and H. Iida, "Cloud-based vr gaming: Our vision on improving the accessibility of vr gaming," in *International Symposium on Ubiquitous Virtual Reality (ISUVR)*, 2017, pp. 24–25.

[10] Y.-C. Li, C.-H. Hsu, Y.-C. Lin, and C.-H. Hsu, "Performance measurements on a cloud vr gaming platform," in *Proceedings of the 1st Workshop on Quality of Experience (QoE) in Visual Multimedia Applications*. New York, NY, USA: Association for Computing Machinery, 2020.

[11] 3rd Generation Partnership Project (3GPP), "3GPP TS 22.261: Service requirements for the 5G system," 3rd Generation Partnership Project (3GPP), Tech. Rep. TS 22.261, January 2024. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3107

[12] International Telecommunication Union (ITU-T), "ITU-T Y.3071: Data Aware Networking (DAN) Architecture," International Telecommunication Union (ITU-T), Tech. Rep. Y.3071, November 2022. [Online]. Available: https://www.itu.int/rec/T-REC-Y.3071

[13] M. Carrascosa and B. Bellalta, "Cloud-gaming: Analysis of google stadia traffic," *Computer Communications*, vol. 188, pp. 99–116, 2022.

[14] M. Lyu, S. C. Madanapalli, A. Vishwanath, and V. Sivaraman, "Network anatomy and real-time measurement of nvidia geforce now cloud gaming," in *Passive and Active Measurement (PAM)*, 2024, pp. 61–91.

[15] X. Xu and M. Claypool, "Measurement of cloud-based game streaming system response to competing TCP Cubic or TCP BBR flows," in *ACM Internet Measurement Conference (IMC)*, Nice, France, 2022.

[16] Y. Wang, M. Lyu, and V. Sivaraman, "Games are not equal: Classifying cloud gaming contexts for effective user experience measurement," in *Proceedings of the 2025 ACM Internet Measurement Conference*. ACM, Oct. 2025, p. 761–776. [Online]. Available: http://dx.doi.org/10.1145/3730567.3764455

[17] M. Casasnovas, C. Michaelides, M. Carrascosa-Zamacois, and B. Bellalta, "Experimental evaluation of interactive edge/cloud virtual reality gaming over Wi-Fi using unity render streaming," *Computer Communications*, vol. 226–227, p. 107919, 2024.

[18] A. Shirmarz, M. N. Bragatto, F. L. Verdi, S. K. Singh, C. E. Rothenberg, G. Patra, and G. Pongrácz, "In-network ar/cg traffic classification entirely deployed in the programmable data plane: Unlocking rtp features and l4s integration," in *IEEE International Conference on Network Softwarization (NetSoft)*, 2025, pp. 477–485.

[19] Steam, "Steam hardware survey," https://store.steampowered.com/hwsurvey/Steam-Hardware-Software-Survey-Welcome-to-Steam, 2020.

[20] SteamDB, "Steam charts," https://steamdb.info/charts/?category=54&sort=peak, 2024.

[21] A. Langley and et al., "The QUIC transport protocol: Design and internet-scale deployment," in *ACM Special Interest Group on Data Communication(SIGCOMM)*, 2017, pp. 183–196.

[22] Steam, "Steam gamenetworkingsockets," https://github.com/ValveSoftware/GameNetworkingSockets, 2026.

[23] Y. Zhang, Z. Jia, L. Jiang, Q. Li, X. Zhang, and Z. Guo, "Modeling virtual reality traffic with head movement in remote rendering," in *IEEE International Conference on Communications (ICC)*, 2025, pp. 1–6.