Evaluating the Effects of Network Delay and Jitter on Gameplay Performance and Player Experience in Cloud Game Streaming

Brendan Cilley, Biwen Dong, Huy Le, Shamit Soni

Professor Mark Claypool

March 7, 2026

Abstract

Cloud gaming systems need low latency and jitter to provide a favorable gaming experience, but how much each matter is not well known. We investigated how network latency and jitter affect player performance and experience. We modified a Unity FPS game to simulate different amounts of latency and jitter. Twenty-five participants completed randomized experimental rounds across five latency tiers (0–100 ms) and five interruption tiers (0–120 ms) in various combinations. Objective performance metrics, including kill count and accuracy, were automatically recorded during gameplay. Following each round, participants provided subjective ratings for overall quality, smoothness, and input responsiveness. Results indicate that delay within the tested range produced minimal changes in gameplay performance and perceived experience. In contrast, increasing jitter led to a noticeable decline in quality of experience, suggesting that players are more sensitive to jitter variability than latency.

Table of Contents

1. Introduction

Advancements in computer graphics and rendering technology have transformed how video games are played. Earlier games like Tetris were designed with minimal computational requirements and simple visuals. In contrast, modern titles like Cyberpunk 2077 include complex rendering, and graphical effects that demand substantially more powerful hardware. As a result, system performance has become a key factor affecting overall player experience[7], [12].

As graphical demands continue to increase, high-end hardware has become a necessity for many modern games. Graphics processing units (GPUs), in particular, make up a significant portion of the cost of a gaming setup. The increase in cost of these specialized processors heightens the barrier to entry for gamers. As a result of these limitations, alternative approaches have been introduced to reduce the reliance of gaming performance on local hardware capabilities.

Cloud-based gaming is a possible solution to this challenge, eliminating the need for high-performance hardware to be owned by the player. In cloud gaming systems, heavy computation like game logic and rendering is performed on remote servers managed by a cloud provider[11], [12]. On the other side, the user's device is used to send device inputs and display the streamed content from the server. With a stable internet connection, cloud-based gaming aims to make playing modern video game titles run on hardware comparable to that of streaming video content.

While cloud-based gaming is promising, it also introduces a set of technical challenges different from traditional game execution or streaming. With video streaming, videos are already recorded and may be pre-buffered[8]. However, gaming is interactive and relies on real-time user input, which requires the system to process and render content in short timeframes. As a result, any delay in this process leads to increased latency, impacting gameplay responsiveness, and may degrade player experience.

In addition to latency, an unstable internet connection can cause variability in frame timing, also known as display jitter[13]. While buffering mechanisms can help smooth interruptions, it adds delay to the rendering process. This introduces additional delay, further reducing responsiveness. This trade-off is detrimental in fast-paced games like first-person shooters (FPS), where input feedback is essential[1]. In these games, even small inconsistencies in frame delivery can disrupt player performance and quality of experience.

In order to mitigate jitter in cloud platforms, playout buffers are often used which add delay between frame delivery and display to store incoming frames before displaying them to the user[3], [10]. Increasing the size of these buffers can smooth network variation and help reduce inconsistencies, but it adds latency to the system, which can impact responses to user input[5]. On the other hand, reducing buffer size minimizes latency, but increases the chances of a user experiencing interruptions or stutters[4]. As a result, cloud gaming systems must configure buffers to find the balance between latency and jitter to maintain high quality of experience across games that rely on input responsiveness and visual smoothness.

Research often treats latency and jitter as separate issues, focusing on their causes and mitigation strategies at the network or system level. However, from a player's perspective, the two are often indistinguishable. Players generally do not differentiate between latency and jitter, instead categorizing them collectively as "lag"[9]. While buffering mechanisms are used to achieve this trade-off, the combined impact of latency and jitter on performance and quality of experience remains an area with insufficient research. Understanding how users behave and perform across different combinations of latency and jitter can be used by playout buffer algorithms to find a balance[2], [6].

To address this gap, this project investigates how latency and jitter influence cloud gaming experiences through a controlled user study. Participants played a modified first-person shooter (FPS) game under a variation of simulated network conditions, allowing for a direct comparison across combinations of delay and interrupts. Subjective quality of experience (QoE) ratings and performance metrics were collected to capture perceptual and behavioral effects.

The study included 25 participants who completed gameplay rounds under combinations of added delay (0-100 ms) and jitter (0-120 ms). Results showed that objective gameplay performance metrics, including kills per round and shot accuracy remained stable across increasing conditions. However, subjective QoE ratings declined as conditions increased, with jitter having a stronger perceptual impact than delay. An equivalence analysis further measured that 100 ms of added delay corresponded to approximately 30-35 ms of jitter based on perceived overall QoE. These findings suggest that while moderate conditions may not heavily impact player performance, jitter has a significant effect on perceived experience compared to delay alone.

The outcome of this study is an experimental analysis of how latency and jitter can impact user experience in cloud-based gaming environments. By analyzing player performance and perceived quality across experimental rounds, this study provides insight into the trade-offs in cloud-based gaming and highlights factors that have the strongest impact on a player's overall quality of experience.

The remainder of this report is set up as follows. Chapter 2 presents background information and related work on cloud-based gaming, latency, and buffers. Chapter 3 describes the experimental methodology, including game selection, implementation, and procedure. Chapter 4 presents results of the user study and subsequent data analysis. Chapter 5 summarizes key findings and implications and Chapter 6 concludes the report with potential directions for future research work.

2. Related Work

This chapter reviews prior research relevant to cloud-based gaming performance and quality of experience (QoE) under network conditions. Existing research has focused on the impact of latency, jitter, and buffering strategies on both objective gameplay performance and subjective experience. By analyzing these contributions, this section establishes the technical and conceptual foundation for the study and identifies the research gap in the trade-off between latency and interrupts.

2.1 - Quality of Experience in Cloud-based Gaming

In the field of cloud-based gaming, Quality of Experience (QoE) serves as the primary metric for evaluating player satisfaction, moving beyond purely technical network statistics to capture the human perception of gameplay. Research consistently demonstrates that while network impairments generally degrade this experience, specific thresholds trigger the most significant drops in player satisfaction, particularly in high-intensity genres.

A foundational study explored these thresholds by simulating a cloud gaming model using the FPS game Lead-Rush[1]. In this study, 31 participants played 29 rounds under varying levels of bitrate, latency, and jitter. The results revealed that all three factors significantly harmed QoE, but the sharpest declines occurred during the initial introduction of impairments, specifically the jump from 0 to 50 ms of latency and 0 to 30 ms of jitter. Notably, these subjective ratings were independent of object performance metrics like kill/death ratios, confirming that players prioritize the “feel” and responsiveness of the network over their actual in-game success.

Building on this, Rossi et al[2]. utilized the Moonlight/Sunshine platform to demonstrate that competitive titles like Counter-Strike 2 require a minimum bitrate of 5 Mbps and latency below 102 ms to remain viable. Their work reinforces that high-intensity mechanics necessitate stricter stability, as FPS games are far more sensitive to bursty jitter than casual titles.

In conclusion, these studies highlight that the initial transition from a stable to an unstable network has the most profound psychological impact on the gamer. By following a similar experimental methodology, utilizing a controlled FPS environment and simulated network conditions, our research aims to further define the critical thresholds that dictate the player experience in cloud gaming systems

While we will not be considering the same factors, this study serves as a highly relevant framework for our own study, as we plan to follow a similar methodology, where users will play a simple FPS game under simulated network conditions.

2.2 - Latency-Jitter Trade-off

The management of network instability in cloud gaming relies heavily on playout buffers to balance visual smoothness against input responsiveness[3].

A key study utilized the Moonlight/Sunshine architecture and a Raspberry Pi running NetEm to test E-Policy, I-Policy,and Queue Monitoring (QM) algorithms during War Thunder benchmarks[5]. While results showed that buffering significantly reduced visible stutters, it introduced up to 166 ms of additional latency, highlighting the critical trade-off between playback stability and delay. This MQP inspired our IQP user study, as both projects share the goal of identifying the optimal trade-off point between delay and interrupts in cloud-based gaming. Their experimental approach to analyzing how buffering affects jitter and latency closely aligns with what our team aims to achieve. Building on their methodology, our study seeks to produce similar insights.

Further research into these mechanisms shows that while aggressive strategies like E-Policy effectively eliminate playout gaps, the resulting lag can be more detrimental to the player's Quality of Experience (QoE) than the jitter itself[10]. Our project builds upon this technical foundation by shifting focus to the "human side" of this trade-off. While previous work defines the objective performance and algorithm logic of buffering, our study utilizes subjective feedback to identify the exact point where the cost of increased latency outweighs the benefit of visual smoothness. By analyzing how players tolerate these specific buffer experiments, we aim to define the fundamental boundaries of the latency-jitter trade-off in real-world cloud gaming environments.

Our research employs the E-policy because interactive scenarios like cloud gaming require consideration of both player subjective experience and the balance between EvLag and jitter. Although their experimental environment—based on 1990s LANs and codecs—differs significantly from modern cloud gaming, their conclusions regarding the latency-gap tradeoff and dynamic buffer control provide valuable reference for our investigation[4] into the relationship between jitter and EvLag in cloud gaming.

2.3 - Network-aware Game Design

The evolution of network aware game design represents a critical shift in the cloud gaming landscape, moving the focus from local hardware power to the management of the "latency loop" - the round-trip time between a player's input and server's visual response. Latency poses a critical limitation in cloud-based games, greatly affecting the players experience at different levels. The results from this experiment show the importance of minimizing the round-trip delay that cloud-games use to stream their games to the clients. Future experiments could explore different genres for the games as well as how jitter might affect the players experience.

This research trajectory is anchored by the foundational work of Claypool and Finkel[7], which quantified the technical toll of network delay by establishing a concrete benchmark: for every 100 ms of added latency, player performance and Quality of Experience (QoE) drop by approximately 25%. While their study defined the inherent "disease" of network lag as a linear degradation of playability across various genres, the subsequent research by Xu et al.[6]introduced a "cure" through the implementation of attribute scaling.

By dynamically adjusting in-game variables, such as expanding hitboxes or slowing target velocities in response to real-time ping detection, Xu et al. demonstrated that game engines can mathematically offset these performance losses, potentially restoring gameplay to baseline levels even under suboptimal network conditions[6]. Furthermore, tools like EvLag[9] have been developed to monitor and simulate these lag conditions specifically for Linux input devices, allowing for more granular study of how input delay affects the user. Together, these papers illustrate a vital paradigm shift from the passive observation of network limitations to the development of active, intelligent mitigation strategies.

Together, these papers illustrate a vital paradigm shift from the passive observation of network limitations to the development of active, intelligent mitigation strategies. This progression is essential to our current study,as it provides both the scientific metrics needed to measure latency’s impact and a theoretical roadmap for how modern, “latency-aware” game design can ensure competitive fairness and accessibility across varying internet speeds.

3. Methodology

To investigate the trade-off between delay and interrupts in cloud-based gaming systems, a controlled user study was conducted using a modified first-person shooter (FPS) game. Participants completed multiple rounds of gameplay under varying combinations of simulated delay and interrupt conditions. Both gameplay performance metrics and subjective quality-of-experience (QoE) ratings were collected to evaluate how simulated network conditions influence user performance and perceived responsiveness.

The two primary factors controlled in this study included simulated network delay and interrupt duration. The study was designed such that each participant experienced all combinations of delay and interrupt conditions through experimental rounds, with a different order for each participant. This approach was chosen to reduce the variability between gaming sessions, allowing for a direct comparison of user performance and experience across different conditions.

This chapter first describes the rationale behind game selection and the platform used for testing. It then defines the independent and dependent variables used in the study, followed by a description of participant recruitment and demographic data collection. Next, implementation tools and technical methods are discussed to demonstrate how conditions were simulated and data was collected. Finally, the experimental procedure is outlined to provide the structure for participant sessions.

3.1 - Game Selection and Modifications

For this study, a first-person shooter (FPS) game was selected because of its dependence on low-latency feedback to be responsive to player input. Previous studies on cloud-based gaming performance utilize FPS games because they involve frequent keyboard and mouse input making them a good choice for evaluating the effects of network delay and interrupts on gameplay performance and user experience.

Although alternative genres such as racing games were considered, the study focused specifically on the FPS genre due to time constraints. Limiting the study to one genre allowed for consistency across trials while allowing enough time for game modifications and participant testing.

After selecting the FPS genre, an open-source, Linux-compatible game was required to allow for game customizations and the integration of external tools. While many open-source FPS games were reviewed, many did not satisfy the experimental requirements of a simple game with low graphics complexity. As a result, the Unity FPS Microgame template was selected as the base game for the study. A preview of the modified game is shown in Figure 1. The game includes HUD elements like an FPS counter, game timer, navigation bar for enemies, health bar, and kill count. The objective of the game is for the player to kill as many enemies as they can in the allocated time, and survive, maximizing their damage dealt and minimizing their damage taken.

Figure 1: Screenshot of the modified Unity FPS Microgame used during participant gameplay sessions

The Unity FPS Microgame provides a lightweight gameplay environment with essential mechanics such as enemies, player movement, weapons, and UI. This allowed the experimental conditions to be easily integrated, while maintaining consistent gameplay across users.

In order to reduce variability between sessions, several modifications were made to the game. Non-essential features such as jumping, sprinting, and crouching were removed. These changes were made to minimize differences in player actions across users, which could introduce variation within user performance.

The game map was also modified to create a smaller, circular arena. This design prevented players from avoiding enemies by hiding in corners and made sure players consistently encountered enemies across sessions. Obstacles were placed within the arena to encourage player movement and engagement while limiting the ability to exploit simple movement patterns.

Gameplay mechanics were adjusted to standardize player interaction. Players were provided with infinite ammunition while maintaining a fixed fire rate, with 3 shots to kill an enemy, ensuring consistent shooting behavior. Enemy spawning was limited to a fixed region of the map, and enemy behavior was set to continuously track and approach the player. The game was modified to only allow one alive enemy at a time, with another enemy spawning after being defeated. These changes ensured a consistent level of engagement throughout each round of gameplay.

A visual countdown timer was added to the user interface to indicate the remaining round duration and time constraint of 30 seconds for each session. The end-of-round screen was modified to remove the ability to restart the game, and instead display a custom message instructing participants to complete the post-round questionnaire.

Finally, custom scripts were created to log gameplay performance metrics and positional coordinates of players and enemies. All recorded data was automatically saved at the end of each round, allowing the experiment to continue without manual intervention and reliable data collection.

3.2 - Experimental Variables

This study used a controlled experimental design, where specific variables were manipulated and measured to evaluate the effects of simulated network conditions on gameplay performance and user experience. The experimental variables were divided into two categories:

independent variables, which represent the conditions applied during gameplay, and dependent variables, which include both objective performance outcomes and subjective quality of experience (QoE) responses. Together, these variables allowed for evaluation of how delay and interrupts influence player behavior and perception.

3.2.1 - Independent Variables

Two independent variables were manipulated in this study: network delay and interrupt duration. Delay and interrupts are common conditions experienced in cloud-hosted gaming environments. These variables were applied in order to evaluate their impact on user gameplay performance and experience.

Delay represents the time between a user's physical keyboard or mouse input and the response displayed on the screen. This study defined and evaluated five levels of delay, as shown in Figure 2. Level 1 served as the baseline, representing an ideal gameplay condition with no added latency. On the other hand, level 5 served as the upper bound, where 100 ms of delay was added, as determined through gameplay testing. The upper bound represents a level of delay where the condition of the game is significantly impacted, without being unplayable. Levels 2, 3, and 4 were selected in increments of 25 ms, in order to maintain a consistent range.

Level	1	2	3	4	5
Delay (ms)	0	25	50	75	100

Figure 2: Delay levels applied during the experiment, ranging from 0 ms to 100 ms in 25 ms increments.

Interrupts were used to simulate network jitter and inconsistent frame delivery, which appear as stutters or pauses in videos. We defined five levels of interrupt duration, as shown in Figure 3. Similar to the delay variable, level 1 represented uninterrupted gameplay, while level 5 was the maximum interrupt duration determined when considering the impact of conditions and playability. Intermediate levels were chosen as increments of 30 ms to measure the impact of a gradual increase in interrupts on the overall gaming experience.

Level	1	2	3	4	5
Interrupts (ms)	0	30	60	90	120

Figure 3: Interrupt duration levels used to simulate jitter, ranging from 0 ms to 120 ms in 30 ms increments.

3.2.2 - Dependent Variables

Dependent variables in this study consisted of both objective and subjective factors. Key metrics included gameplay performance and quality of experience (QoE) ratings. Collecting multiple types of data allowed for a complete evaluation of how simulated network conditions influenced player performance and gameplay quality.

Gameplay performance data was recorded during each round and automatically saved following the end of a round. Primary performance metrics included:

- Kill count: The number of enemies defeated by the player.
- Shots fired: The number of bullets fired by the player.
- Shots hit: The number of bullets fired by the player, which successfully hit an enemy.
- Damage taken: The amount of damage taken by the player from enemy attacks.

- Positional data: Coordinates (x, y, z) of the player and enemies on important events, including enemy spawn and shot firing.

Following the end of each round, users completed a short questionnaire, which was used to assess their gameplay experience. Users provided a rating for three questions using a five-point scale, where a value of 1 indicated the poorest experience and a value of 5 indicated the best experience. The questionnaire focused on overall quality, smoothness of the game, and responsiveness of input devices, under simulated network conditions. The exact questions presented to participants are shown in Figure 4.

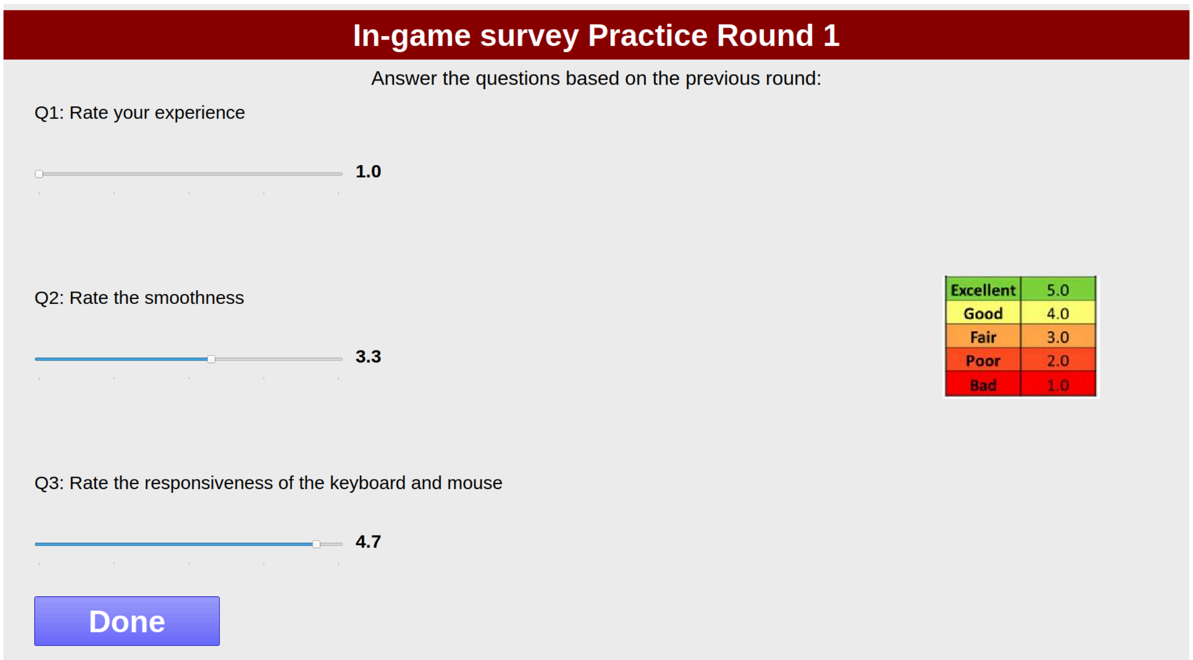

Figure 4: Post-round questionnaire used to collect participant quality of experience (QoE) ratings for overall quality, smoothness, and responsiveness.

Together, these measures provided insight into the measurable performance impact and perceived user experience associated with different combinations of delays and interrupts.

3.3 - Participant Recruitment and Demographics

Participants were recruited from the WPI student population and volunteered to take part in the user study. Each participant was assigned a unique user ID which was used across the data collection process, with only the ID used for data analysis.

Before beginning the study, participants completed a short survey to provide background information relevant to the study. The survey included general questions like age, gender, major, and graduation year, as well as gaming-related questions like level of experience with first-person shooter (FPS) games, and playing games on different platforms. This data was collected to support analysis of performance and experience across different groups of users.

FPS experience level was considered particularly important, as familiarity with similar game mechanics may influence performance outcomes or perception of gameplay quality. Demographic data was isolated from gameplay data and was only accessed by the research team in accordance with IRB-approved data procedures.

3.4 - Experimental Environment

This section describes the controlled setup used during participant sessions, including hardware configuration and the baseline local system delay.

3.4.1 - Laboratory and Hardware Configuration

All sessions were held in a controlled computer laboratory environment to maintain consistency across participants. Each participant was seated at the same workstation within the lab, consisting of a desk with a monitor, keyboard, mouse, and wired headphones. The room layout is shown in Figure 5.

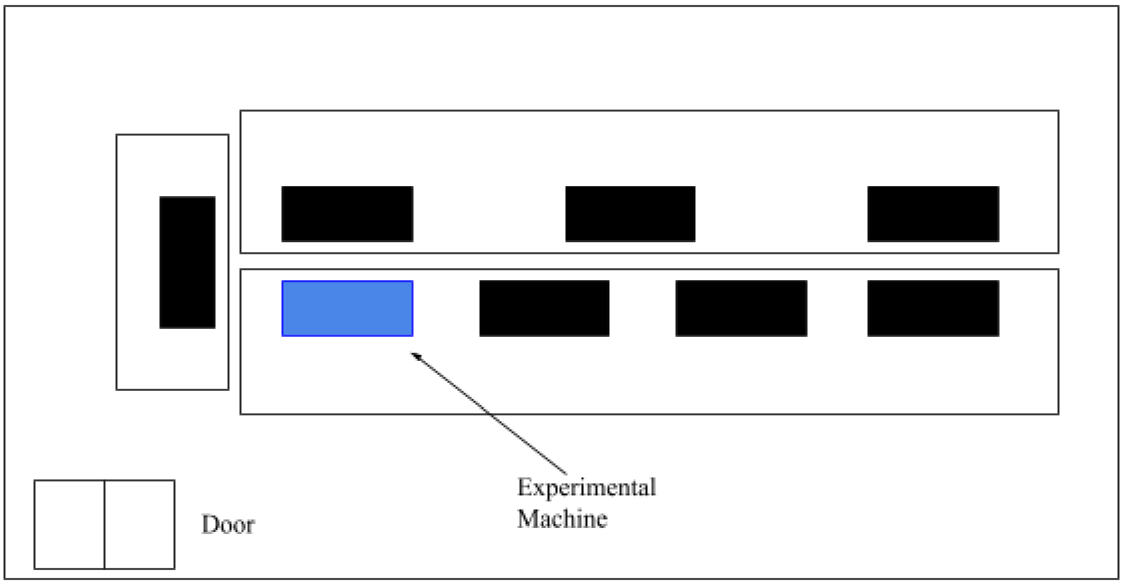

Figure 5: Diagram of the FL316 computer laboratory layout used during participant testing sessions.

The workstation was equipped with a Logitech G502 Hero gaming mouse and Redragon mechanical gaming keyboard to provide a standard level of input responsiveness. A 27-inch BenQ monitor with a resolution of 1920 x 1080 and a refresh rate of 60 Hz was used for all sessions. Audio output was passed through a pair of on-ear headphones to ensure consistent sound feedback.

The experimental computer contained an Intel Core i7-8700K processor, 64 GB of RAM, and an NVIDIA GeForce GTX 1080. The system operated on Ubuntu 22.04.1 (Linux kernel 6.8.0-94-generic, x86_64 architecture), and the game was developed and executed using Unity Editor version 6000.2.4f1. The computer was restarted between each session and all non-essential programs were closed.

Network conditions were controlled through software-based simulation rather than real, external network variability. The system was connected using a wired ethernet connection, and no additional network-intensive applications were active during sessions.

3.4.2 - Baseline System Latency

In addition to controlling delay and interrupt conditions, the baseline latency of the experimental computer was measured to under the delay present under the 0 ms condition.

Baseline latency was measured using high-speed video analysis. An iPhone 13 was used to record the monitor at 240 frames per second in slow-motion mode. Multiple trials were recorded in which a mouse button was pressed while changes in the display scene were observed. Frame-by-frame analysis was performed using VLC media player to count the number of frames between the physical mouse click and the first visible on-screen response.

Across five trials, the measured frame differences were reported and an average was calculated:

$$\frac{Frame\ Differences}{Number\ of\ Trials} = \frac{8 + 10 + 8 + 10 + 7}{5} = 8.6\ frames$$

Given the recording rate of 240 FPS, each frame corresponds to:

$$\frac{1}{240} = 0.004167\ sec/frame = 4.167\ ms/frame$$

Multiplying by the average frame difference yields:

$$4.167\ ms/frame * 8.6\ frames = 35.836\ ms$$

Therefore, the baseline end-to-end system latency under no added conditions was approximately 36 ms. This value represents the system latency that was present across all experimental conditions.

3.5 - Implementation and Tools

This study utilized a set of tools and software to maintain a consistent and controlled experiment. Various tools were used for purposes such as simulating network conditions, collecting data, and automating the process of the study.

3.5.1 - Applying Delay

Simulated network delay was implemented using EvLag version 2.5, a Linux-compatible input latency tool[9]. EvLag was configured to add latency between the player's keyboard/mouse input and the on-screen response, simulating client-side latency, which is commonly experienced in cloud-based gaming. EvLag was configured at the start of each experimental round to apply the predefined delay level to input devices. Additionally, since EvLag is an external tool, this approach allowed delay conditions to be implemented without worrying about altering the game code.

3.5.2 - Applying Interrupts

Interrupts were implemented directly within the Unity game engine using custom C# scripts. These scripts defined how often interrupts would occur and their duration based on the levels defined previously. Since the game was directly modified, five versions of the Unity game were built, each containing a different level of interrupts coded into it. During each round, the participant would play a specific build of the game, corresponding to the level of interrupts applied. By implementing interrupt behavior in the update loop, the game produced stuttering without altering gameplay mechanics.

3.5.3 - Data Collection

Game performance metrics were automatically recorded during each round using custom C# logging scripts that were integrated into the Unity game. Collected data included player and enemy position and performance statistics such as kills, shots fired, shots hit, and damage taken.

At the end of each round, recorded data was renamed and moved to a designated location by a Python script. Similarly, post-round questionnaire data was handled by a Python script which saved data for each user by round number and labeled with experimental conditions. Each dataset was labeled with a participant identifier and was uploaded to Google Drive to prevent data loss and facilitate subsequent analysis.

3.6 - Experimental Procedure

Each participant completed a structured session consisting of an introduction phase, experimental rounds, and post-round questionnaires. In order to maintain consistency across all participant sessions, a procedure was developed and followed by all proctors. The study was approved by the WPI Institute Review Board (IRB).

Upon arrival, participants were provided details about the purpose of the study and instructions on how to participate. Participants were then prompted to sign an informed consent form, followed by a brief demographics survey. Before starting experimental trials, participants played two practice rounds to familiarize themselves with the controls and objective of the game. Although data was collected during these rounds, it was not utilized during analysis.

Following the training phase, participants completed a series of 30 second gameplay rounds, each with a different combination of delay and interrupt conditions. In order to reduce bias, the order of experimental rounds was balanced across participants using a latin square design. Aside from the applied delay and interrupt conditions, each round had the same gameplay objective and difficulty, to ensure consistency across trials.

Following the completion of each round users were prompted to complete a short questionnaire to collect data related to quality of experience (QoE). Both gameplay and QoE metrics were saved for each round, along with the applied delay and interrupt conditions. Participants repeated this process for 25 experimental rounds. Full experimental sessions for each participant lasted approximately 30 minutes, including setup, gameplay, and data collection.

At the end of the session, participants were debriefed and thanked for their participation. Participants were also informed of incentives including IMGD play-testing credit and a raffle for a $25 Amazon gift card.

4. Results and Analysis

This chapter presents the results of the experimental conditions of delay and jitter on objective gameplay performance and subjective quality of experience (QoE). The findings are separated into three main categories. First, participant demographics are analyzed to better understand the background of our sample set. Next, gameplay metrics are observed to determine whether increasing delay and jitter produced a measurable difference in player performance. Lastly, quality of experience is examined to understand how participants perceived increasing conditions of delay and jitter.

Together, these forms of analysis provide an understanding of whether impairments influence player performance, perceived experience, or both.

4.1 - Data Integrity and Preprocessing

Prior to analysis, the dataset was reviewed to verify consistency across experimental conditions. A total of 26 participants completed the study. However, the first participant was removed from analysis due to technical issues during the time of the study. More specifically, there was an issue applying delay during the participants' session, which led to inconsistent results. To maintain validity in the study, this participant's data was removed from all performance and QoE analysis. Therefore, the final dataset included a total of 25 participants.

Additionally, several participants experienced rounds that terminated slightly before the intended 30 seconds, due to technical difficulties. In total, five participants experienced at least one shortened round. To maintain consistency among the rest of the dataset, performance metrics collected from these rounds were adjusted to compensate for the shortened duration. This was completed by analyzing the observed rate of performance during a specific round, and then adjusting metrics to what would be expected in a full 30 second round.

No extreme outliers were removed from the dataset and all data fell within the expected ranges of performance.

4.2 - Demographics

The following demographic analysis reflects the final sample of 25 participants. Their ages were predominantly concentrated between 19 and 21 years old, with an average age of 22.04 years, primarily representing students within the traditional university age range. Figure 6 shows the distribution of gender across participants where approximately two-thirds of the sample were male. Academic backgrounds were mainly concentrated in computer science and IMGD disciplines. Graduation years were primarily distributed between 2026 and 2029, with the largest cohort being the class of 2028, indicating that most participants were in the middle of their undergraduate studies.

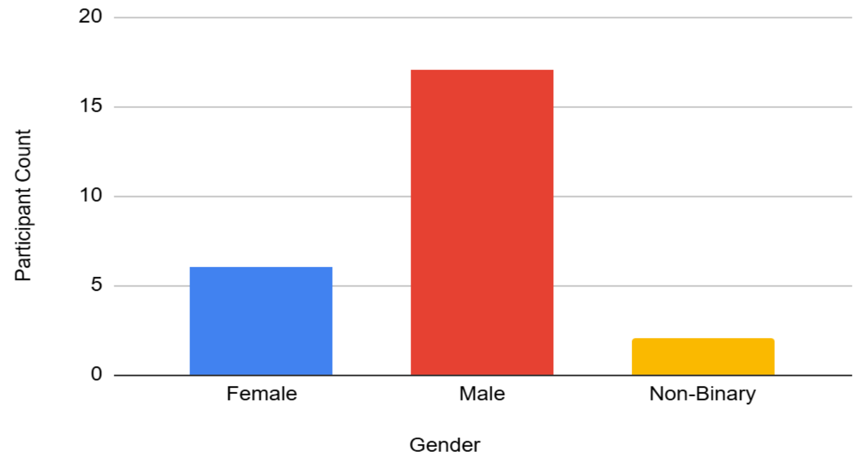

Figure 6: Distribution of participant gender across the study sample. Bars represent the number of participants identifying as male, female, and non-binary (N = 25).

For gaming activity, participants averaged approximately 13.1 hours of gaming time weekly, with a median of 10 hours, indicating a moderate gaming frequency overall. Self-reported gaming proficiency showed a split between experience levels, but novice players constituted the majority, as shown in Figure 7.

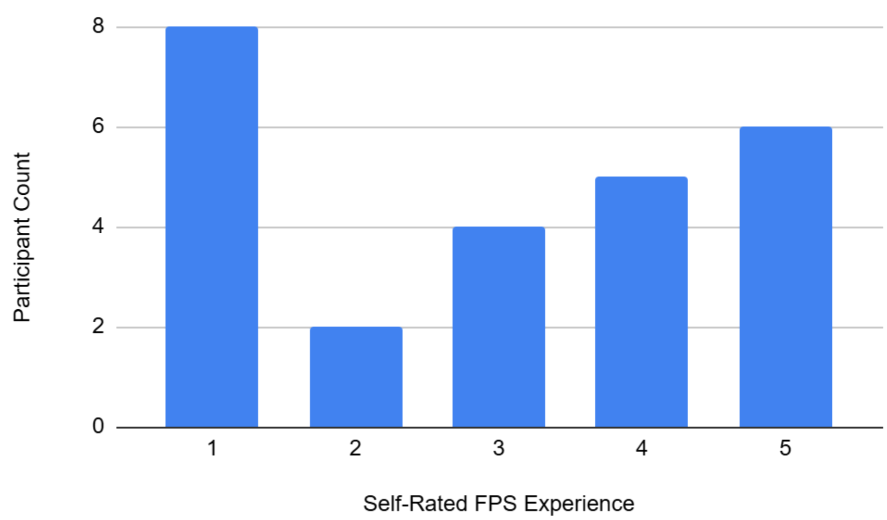

Figure 7: Distribution of self-rated FPS gaming experience across participants. Ratings range from 1 (novice) to 5 (expert). Bars represent the number of participants in each rating (N = 25).

Regarding gaming platforms, participants primarily played on PC, with keyboards and mice for gameplay, aligning with the experimental environment of this study. Overall, the sample consists mainly of PC gamers enrolled in computer-related majors.

4.3 - Gameplay Performance Analysis

This section analyzes the impact of added delay and jitter on gameplay performance. Player performance was primarily assessed using kills, shot accuracy, and shots fired. First a baseline distribution of score was established to understand player performance, followed by analysis of metrics under increasing conditions of delay and jitter.

4.3.1 - Baseline Performance Distribution

In order to establish a baseline for player performance across experimental conditions, two metrics were analyzed: average kills per round and total kills per user. Figure 8 presents boxplots of these both metrics side-by-side to visualize performance across all participants.

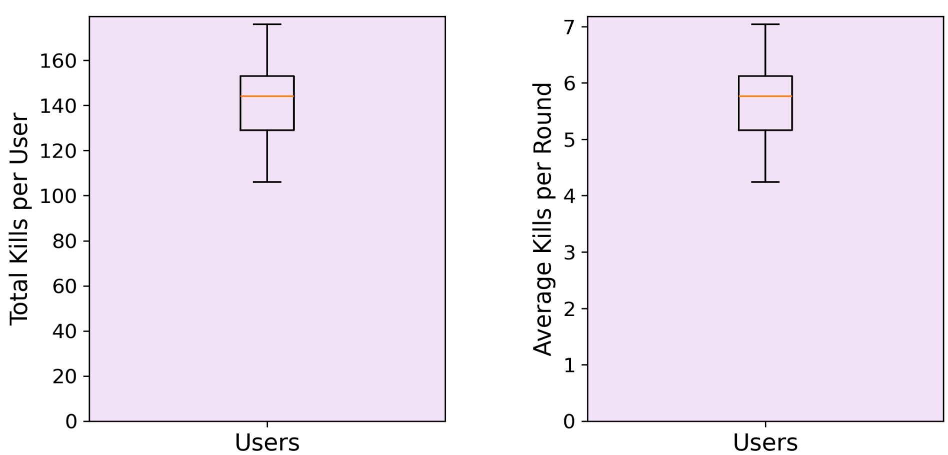

Figure 8: Baseline performance distributions across all participants. Left: average kills per round. Right: total kills per user. Boxplots display median and interquartile range (IQR) across all experimental conditions.

The left panel shows the total kills per user across all completed rounds. As expected, this distribution has a larger spread compared to the average kills per round. The median total kill count is 144.0, with an IQR spanning from 129.0 to 153.0. The full range of data spans from 106.0 to 176.0 kills. Similarly to the average kills per round, while certain participants performed notably above or below the median, all values remained within the fences of the IQR (93.0 to 189.0 kills), indicating that there were no outliers in the cumulative kills performance.

The right panel presents the distribution of average kills per round. The median value is 5.8 kills, with the interquartile range (IQR) extending from 5.2 to 6.1. The majority of players fall in this range, indicating a relatively consistent performance across rounds. The minimum and maximum values were 4.1 and 7.0 kills per round, respectively. While these values fall outside of the main cluster, they represent the extremes of the distribution and do not represent outliers in the dataset.

To further analyze baseline performance of the dataset, shooting accuracy was also reviewed. Figure 9 presents boxplots of average accuracy per round and overall accuracy per user.

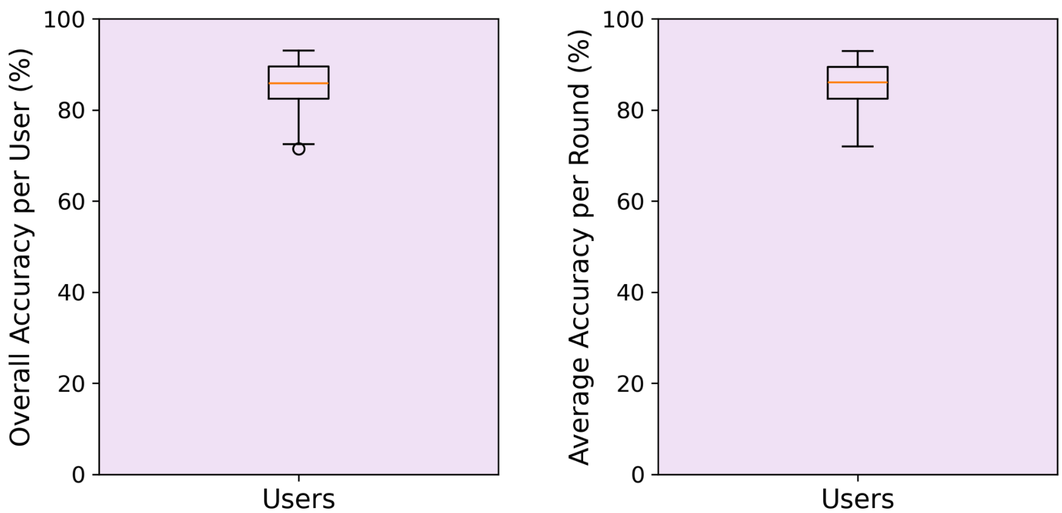

Figure 9: Baseline accuracy distributions across all participants. Left: overall accuracy per user. Right: average accuracy per round. Boxplots display median and interquartile range (IQR) across all experimental conditions.

The left panel presents overall accuracy per user, calculated as total shots hit divided by total shots fired, across all rounds. The distribution appears to be similar to the per-round average distribution. A single user falls outside of the lower fence of the IQR, representing a minor outlier in the dataset. However, this difference is negligible and does not affect the overall distribution of baseline performance.

The right panel shows the distribution of average accuracy per round across participants. The median accuracy is 86%, with an interquartile range spanning from 82.5% to 89.5%. The distribution indicates moderate consistency in shooting precision across rounds, with no outliers.

Together, the baseline kill and accuracy distributions reveal a moderate level of variability among participant performance, while still maintaining consistency across rounds. This serves as a reference point for gameplay performance across users, which can be compared when evaluating the effects of delay and jitter on performance.

4.3.2 - Kills per Round

Figure 10 shows the mean kills per round of gameplay as a function of added conditions. The left panel shows performance under different delay conditions, while the right panel shows performance under different jitter conditions. The x-axes represents the level of impairment in milliseconds, and the y-axis represents the score as kills per round across all participants. Error bars on the graph represent a 95% confidence interval for each point.

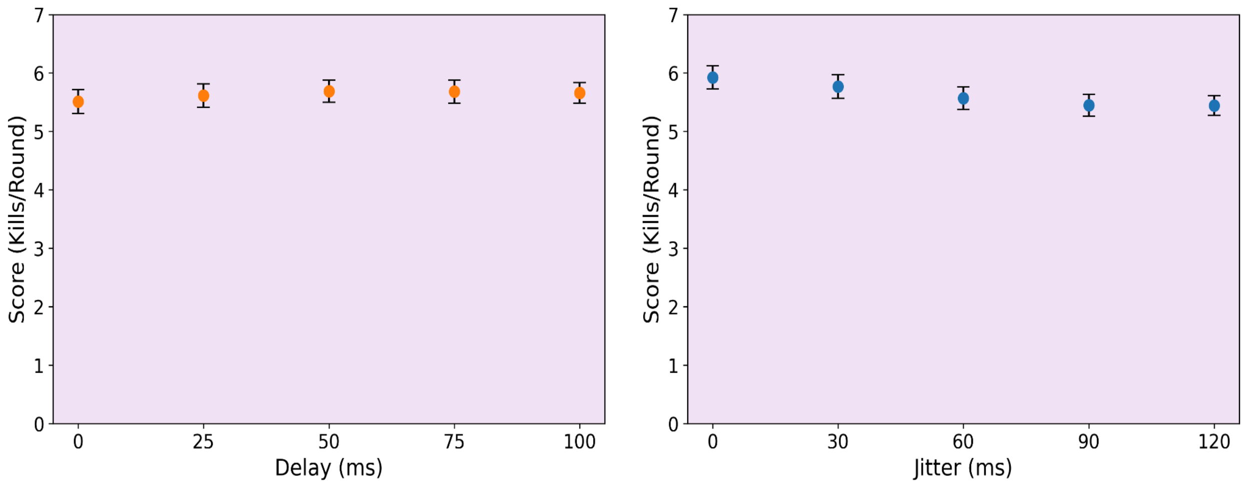

Figure 10: Average kills per round under increasing conditions. Left: added delay (0-100 ms). Right: added jitter (0-120 ms). Error bars represent 95% confidence intervals across participants.

In the left panel, as delay conditions increase, the score remains relatively consistent. While there are slight fluctuations in the score, the confidence intervals continue to overlap, indicating that these differences are not statistically significant. At higher levels of delay, the average kills per round does not consistently decline and in some instances shows a slight increase. However, once again these differences in the score can be attributed to variability across participants and is not enough to establish a strong relationship. This indicates that an increase in added delay does not have a measurable impact on a player's ability to kill enemies during a round of gameplay.

In the right panel, mean kills per round under increasing jitter shows a minor downward trend in kills. However, similar to the analysis on delay, the 95% confidence intervals overlap across the majority of conditions, indicating the decline is due to participant variability. While jitter may have a slightly more noticeable trend with than delay, the overall reduction in score remains minimal.

Overall, these results suggest that within the tested range of values, neither delay nor jitter alone produced a statistically significant change in player score.

To further evaluate the independent effects of delay and jitter on gameplay performance, kills per round were also analyzed under conditions where the opposing impairment was absent. Specifically, mean kills per round were examined as a function of delay when jitter was fixed at 0 ms, and as a function of jitter when delay was fixed at 0 ms. This comparison removes potential interaction effects between the two conditions and allows for a clear observation of each impairment independently. Figure 11 presents the average kills per round under these isolated conditions.

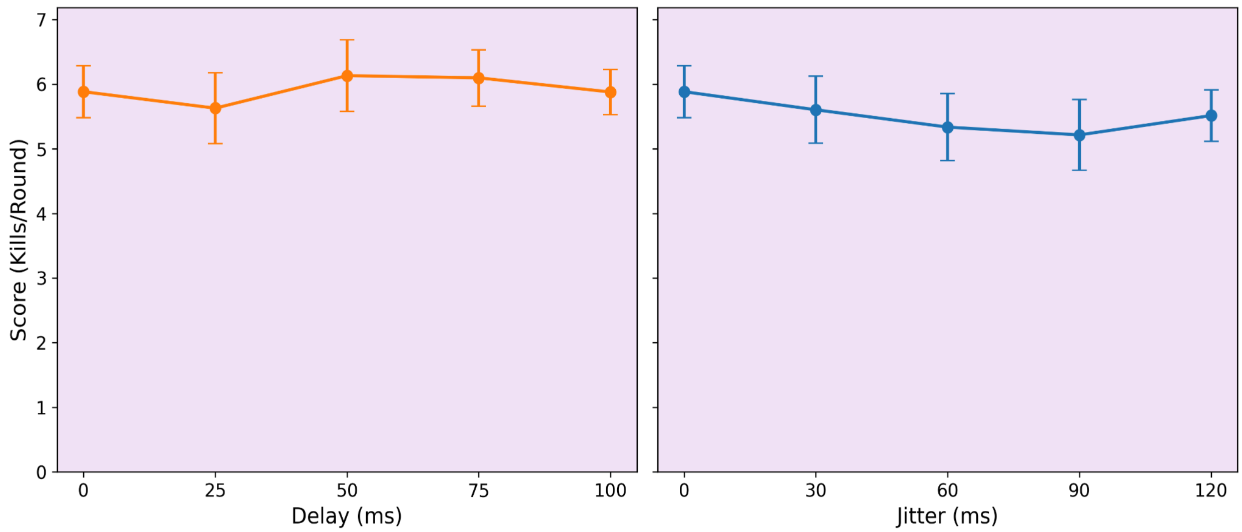

Figure 11: Average kills per round under isolated conditions. Left: Kills vs. delay when jitter is absent. Right: Kills vs. jitter when delay is absent. Error bars represent 95% confidence intervals across participants.

In the left panel, when jitter is held constant at 0 ms, the average kills per round remain relatively stable as delay increases from 0 to 100 ms. Minor fluctuations in score are present between conditions, but the confidence intervals largely overlap, indicating that delay alone does not have a large impact on player performance.

In the right panel, when delay is fixed at 0 ms, the mean kills per round shows only a slight downward trend as jitter increases from 0 to 120 ms. While small variations occur between adjacent jitter levels, the confidence intervals overlap across most conditions. This suggests that jitter alone does not have a major effect on player performance.

Together, this isolated comparison reinforces previous results, showing that neither delay nor jitter independently produce a significant reduction in player kills per round within the tested range of values.

4.3.3 - Accuracy and Shooting Behavior

Accuracy per round was analyzed to determine whether increasing delay and jitter affected player shooting precision. Figure 12 presents the mean accuracy per round across participants under varying conditions of delay and jitter. Error bars represent a 95% confidence interval.

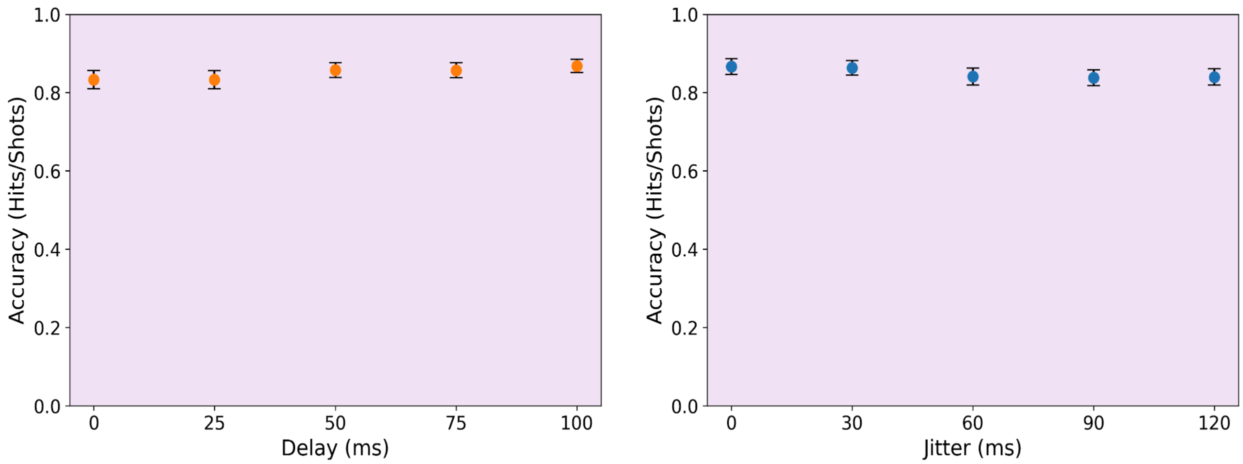

Figure 12: Average shot accuracy per round under increasing conditions. Left: added delay (0-100 ms). Right: added jitter (0-120 ms). Error bars represent 95% confidence intervals across participants.

In the left panel, mean accuracy appears to be stable across delay conditions 0 ms to 100 ms. Although accuracy slightly fluctuates, the 95% confidence intervals overlap, indicating that the observed difference is not statistically significant. No consistent downward trend is observed as delay conditions are increased, suggesting that added delay does not have a meaningful impact on player shot accuracy.

In the right panel, a slight decrease in accuracy is present at higher levels (60-120 ms) compared to lower levels (0-30 ms). However, similar to delay results, the confidence intervals overlap across most conditions. This finding once again suggests that the variation in shot accuracy is not statistically significant and jitter did not have a notable impact on shot accuracy.

To further investigate whether accuracy values were influenced by player behavior, the number of shots fired per round was also analyzed. Figure 13 presents mean shots fired per round under increasing conditions of delay and jitter. Similar to the accuracy results, mean shots fired per round remained stable across increasing delay and jitter conditions. The 95% confidence intervals had substantial overlap, indicating no statistical significant difference.

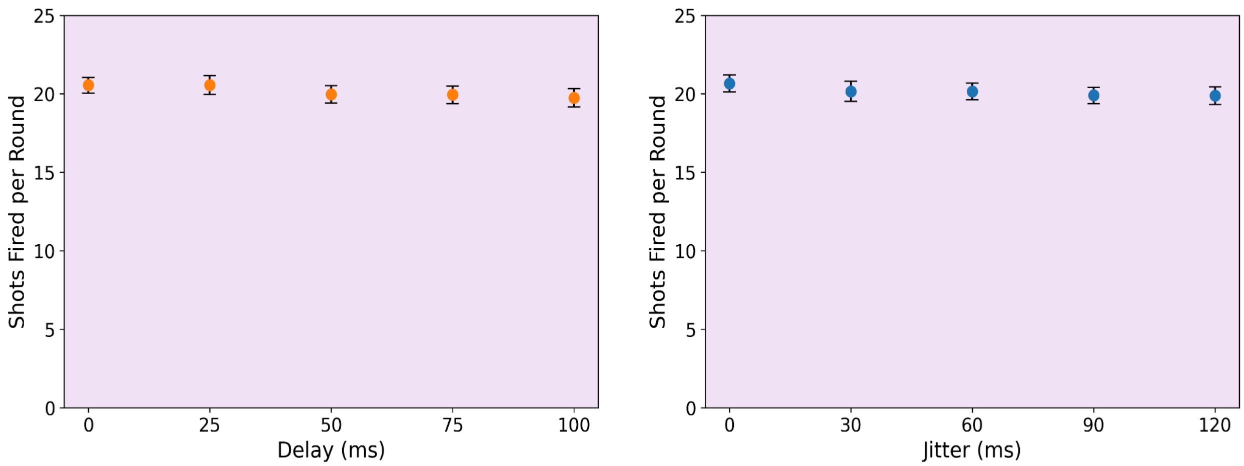

Figure 13: Average shots fired per round under increasing conditions. Left: added delay (0-100 ms). Right: added jitter (0-120 ms). Error bars represent 95% confidence intervals across participants.

Overall, these findings suggest that players did not alter their firing frequency as conditions increased. Therefore, the results observed in player shot accuracy is not due to a reduction in shots fired per round, but rather consistency in shooting performance.

4.4 - Subjective QoE Analysis

This section evaluates how increasing levels of delay and jitter influence participants' perceived gameplay experience. Gameplay experience was split into overall, responsiveness, and smoothness ratings. Each experience metric is presented side-by-side to directly compare the effects of delay and jitter.

4.4.1 - Overall QoE

Overall quality of experience (QoE) ratings were analyzed to determine how increasing levels of delay and jitter influenced participants' perceived gameplay experience. Figure 14 presents overall QoE as a function of added delay and added jitter.

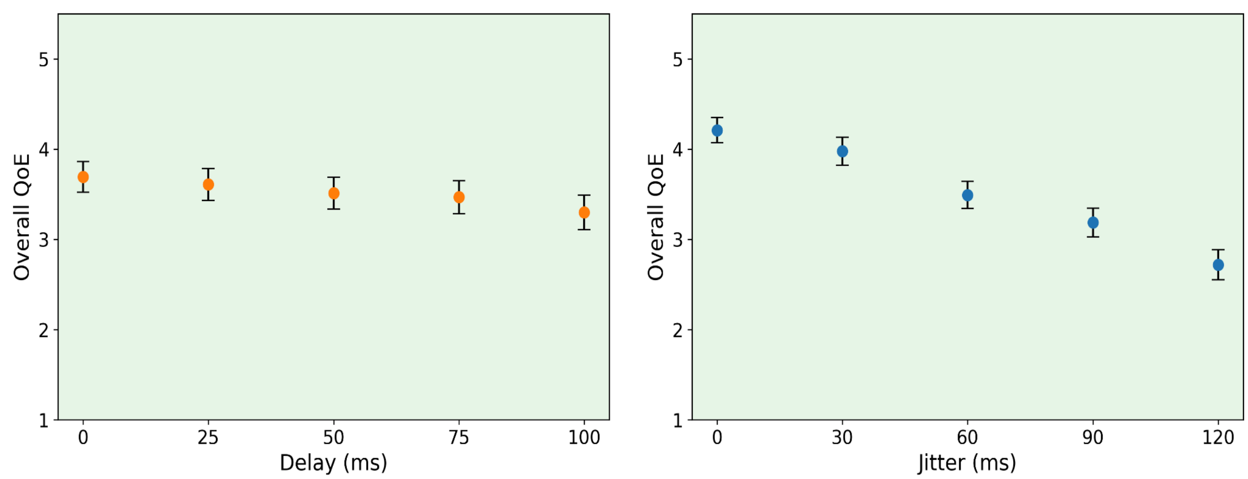

Figure 14: Overall quality of experience (QoE) as a function of added delay (left) and added jitter (right). Error bars represent 95% confidence intervals across participants.

In the left panel, overall QoE shows a minor downward trend as delay increases from 0 ms to 100 ms, However, this decrease is relatively small, and confidence intervals tend to overlap across adjacent delay conditions. This suggests that participants perceived a level of degradation in quality with increased delay, but the overall impact was limited within the tested range of values.

In contrast, the right panel shows overall QoE under increasing conditions of jitter. A stronger negative relationship is observed between overall QoE and jitter. The reduction in ratings across jitter levels is more noticeable than with delay, with less overlap in confidence interval at higher levels. This indicates that jitter had a stronger influence on perceived experience than delay.

Overall, these results show that while increasing delay only led to minor changes in overall QoE, increasing jitter resulted in a clearer decline in perceived experience. This finding suggests that participants may be more sensitive to variability in jitter, than delay.

To further examine the independent effects of delay and jitter, overall QoE was also analyzed under conditions where the opposing impairment type was held constant at zero. Specifically, overall QoE was examined as a function of delay when jitter was fixed at 0 ms, and as a function of jitter when delay was fixed at 0 ms. This comparison removes potential interaction effects and allows for a clear evaluation of each condition independently. Figure 15 presents overall QoE under these isolated conditions.

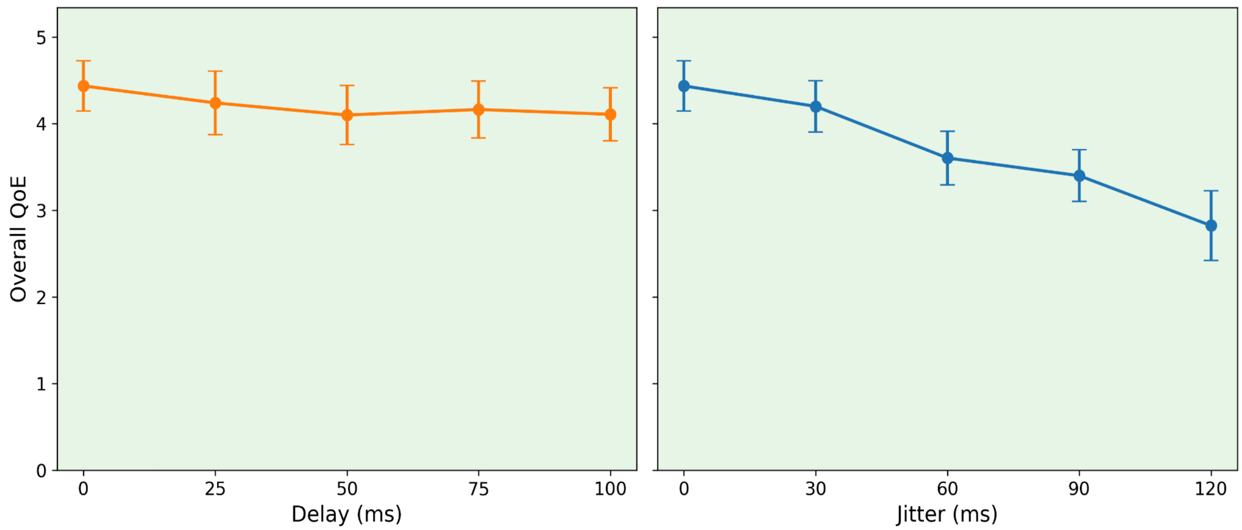

Figure 15: Overall quality of experience (QoE) under isolated conditions. Left: QoE vs. delay when jitter is absent. Right: QoE vs. jitter when delay is absent. Error bars represent 95% confidence intervals across participants.

In the left panel, when jitter is held at 0 ms, overall QoE remains relatively stable as delay increases from 0 to 100 ms. Confidence intervals overlap across cases, indicating that delay alone does not seem to have a significant reduction on overall experience.

In contrast, the right panel shows that when delay is held at 0 ms, overall QoE declines more consistently as jitter increases from 0 to 120 ms. While confidence intervals may overlap across a few conditions, the separation in overall QoE between the lowest and highest jitter condition is clear. This suggests that jitter alone has a strong impact on the degradation of overall experience.

Together, this isolated comparison confirms that the decline observed in overall QoE is driven primarily by changes in jitter, rather than delay.

4.4.2 - Perceived Responsiveness

Perceived responsiveness quality of experience (QoE) ratings were analyzed to evaluate how increasing levels of delay and jitter influenced a participant's perception of system reaction time. Figure 16 presents perceived responsiveness as a function of added delay and jitter.

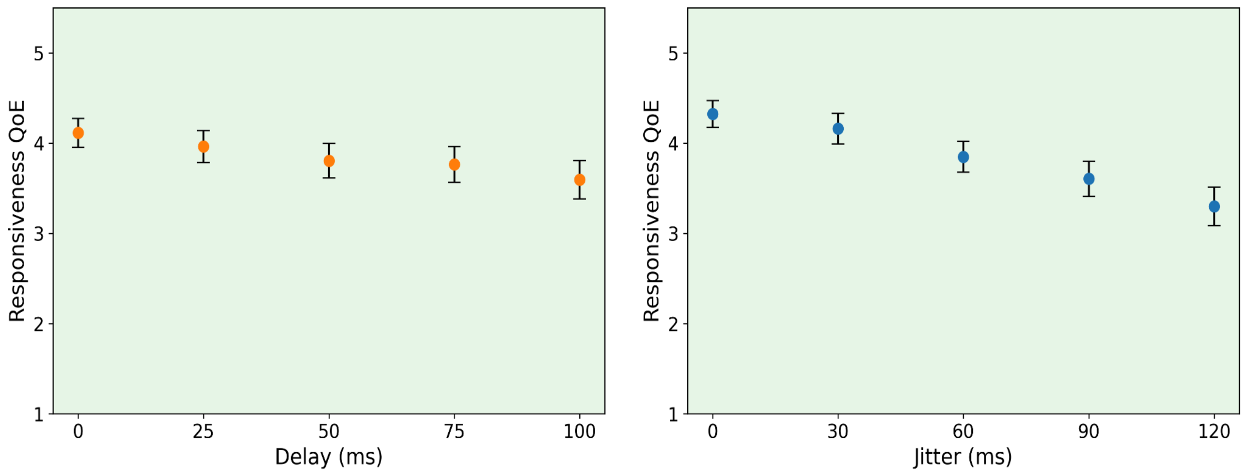

Figure 16: Responsiveness quality of experience (QoE) as a function of added delay (left) and added jitter (right). Error bars represent 95% confidence intervals across participants.

In the left panel, responsiveness ratings show a minimal downward trend as delay increases from 0 ms to 100 ms. Confidence intervals also display overlap, indicating the differences in perceived responsiveness are likely attributed to variability. This suggests that within the tested range of delay values, participants did not perceive a meaningful decline in gameplay responsiveness.

In the right panel, responsiveness ratings under increasing conditions of jitter showed a much clearer downward trend. While confidence intervals may overlap among moderate levels of jitter (30-90 ms), the difference between lowest (0 ms) and highest (120 ms) jitter conditions is more noticeable. This indicates that participants were more sensitive to extreme jitter levels than to incremental changes.

Overall, direct comparison between conditions show that increased delay did not have a significant impact on participants' perceived responsiveness during gameplay. In contrast, participants were able to differentiate between the lowest and highest levels of jitter, while having a similar level of responsiveness perception at moderate levels of jitter. These findings indicate that participants may be more sensitive to variability in jitter when it comes to gameplay responsiveness.

4.4.3 - Perceived Smoothness

Perceived smoothness quality of experience (QoE) ratings were analyzed to determine how increasing levels of delay and jitter influenced a participant's perception of visual consistency. Figure 17 presents perceived smoothness as a function of added delay and jitter.

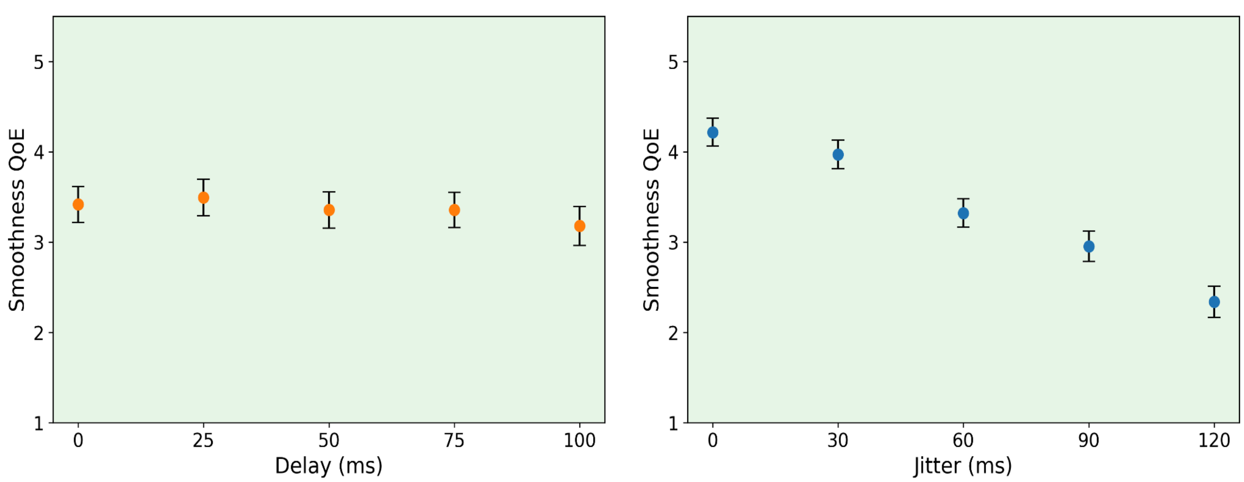

Figure 17: Smoothness quality of experience (QoE) as a function of added delay (left) and added jitter (right). Error bars represent 95% confidence intervals across participants.

In the left panel, smoothness ratings under increasing conditions of delay show minimal fluctuation across conditions. This finding along with substantial overlap among confidence intervals suggest that increased delay did not significantly impact participants' perception of smoothness.

In contrast, the right panel outlines a clear downward trend in smoothness ratings as jitter increases from 0 ms to 120 ms. While slight overlap exists at lower levels (0 and 30 ms), ratings show a clear decline at higher levels (90 and 120 ms). This trend appears to be similar, but stronger than that observed for overall QoE, suggesting that jitter particularly affected participants' perception of smoothness during gameplay.

Together, these findings indicate that perceived smoothness was largely unaffected by delay, but consistently declined with increasing jitter.

4.4.4 - Joint Effects of Delay and Jitter

To examine the combined influence of delay and jitter on perceived gameplay quality, heatmaps were utilized to visualize perceived overall, responsiveness, and smoothness QoE. In each heatmap, the horizontal axis represents delay and the vertical axis represents jitter, allowing their intersection to represent a combination of delay and jitter together. Color intensity ranges from dark green (highest perceived quality) to dark red (lowest perceived quality), with each cell representing the mean participant rating. Figure 18 presents heatmaps for each perceived rating.

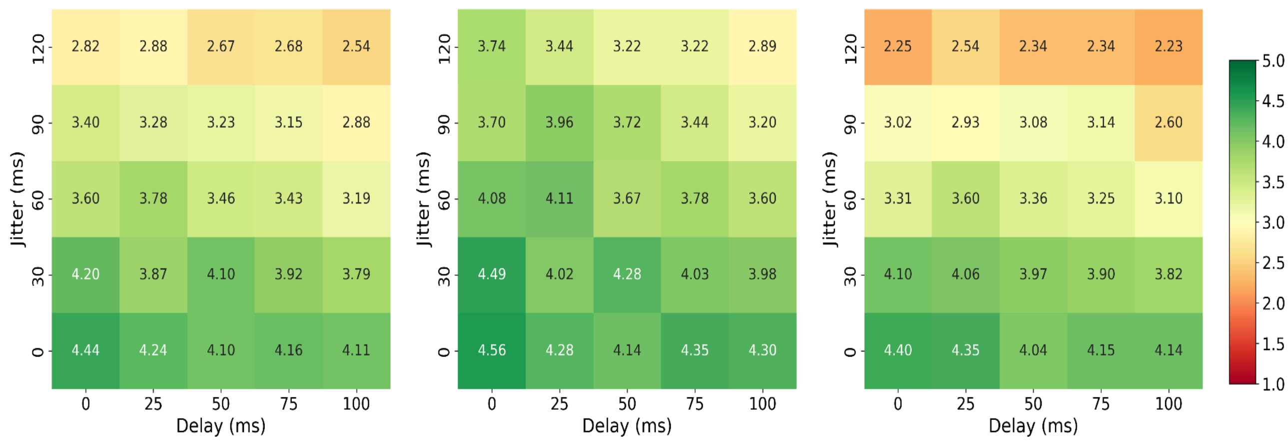

Figure 18: Heatmaps showing mean quality of experience (QoE) ratings as a function of combined delay and jitter for (a) overall, (b) responsiveness, and (c) smoothness QoE. Color intensity represents mean participant ratings.

The overall QoE heatmap shows a vertical gradient, indicating that ratings decline as jitter increases. At low jitter levels, overall QoE remains relatively high across all delay conditions, with only minor variations as delay increases. However, as jitter reaches higher levels, particularly at 90 and 120 ms, ratings shift from dark green to lighter green, and even yellow and orange tones, showing a noticeable degradation in perceived experience. In contrast, increasing delay alone produces small changes in color intensity, noted by the lack of a horizontal gradient. This pattern suggests that jitter is the dominant factor influencing overall QoE, while delay contributes only slightly within the tested range.

The responsiveness heatmap remains green across most delay and jitter combinations, indicating participants generally had high ratings for responsiveness. Only when both delay and jitter reach higher levels does a noticeable shift toward lighter green and yellow occur, with the average rating decreasing to 2.89 in the most extreme conditions. This suggests that responsiveness is only moderately impacted when delay or jitter is increased independently. However, when both impairments are increased together, their combined effect becomes more noticeable. Compared to overall QoE, responsiveness appears less sensitive to jitter alone, and delay does not produce a strong effect.

The smoothness heatmap exhibits the most pronounced vertical gradient. When jitter reaches 120 ms, ratings consistently fall into the orange range, even when delay remains low. In contrast, increasing delay alone results in very minor changes when jitter is minimal, with most cells remaining dark green. This indicates that smoothness perception is highly sensitive to jitter, while delay has a very limited influence on smoothness in isolation. The strong degradation at high jitter levels suggests that variability in packet timing is critical to perceived smoothness.

Together, across all three QoE metrics, jitter demonstrates a stronger and more consistent influence on perceived quality than delay. Overall QoE and smoothness show clear vertical gradients, indicating sensitivity to jitter, whereas responsiveness remains relatively stable across conditions. These heatmaps therefore support the earlier single-variable analyses, confirming that jitter has a greater perceptual impact than delay alone within the tested ranges.

4.4.5 - Equivalence Analysis of Delay and Jitter

While previous sections demonstrated that jitter has a stronger perceptual impact than delay, this section quantifies their relative effects by establishing an approximate equivalence relationship between both conditions.

To isolate the independent effect of each condition, mean overall QoE values were analyzed under conditions where the opposing impairment was absent (delay with jitter fixed at 0 ms, and jitter with delay fixed at 0 ms). Figure 19 presents an equivalence curve derived by matching overall QoE values across delay-only and jitter-only conditions. The x-axis represents delay (ms) and the y-axis represents jitter (ms) that produces a comparable mean QoE rating.

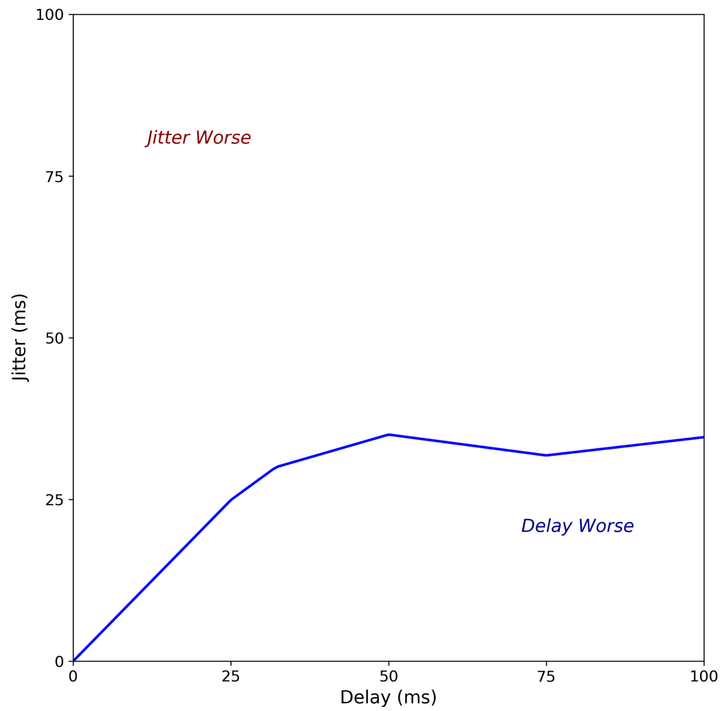

Figure 19: Delay-jitter equivalence curve derived from overall QoE under isolated conditions. Each point represents jitter values interpolated to match mean QoE observed under delay-only conditions.

As shown previously in section 4.4.1 within isolated conditions, overall QoE remained relatively stable as delay increased from 0 ms to 100 ms, decreasing from 4.44 to 4.12. In contrast, jitter-only conditions produced a much steeper decline in QoE, dropping from 4.44 at 0 ms to 2.82 at 120 ms.

By interpolating between jitter-only data points, a delay of 100 ms (mean QoE = 4.12) corresponds to between 30 and 35 ms of jitter. This indicates that a substantially smaller magnitude of jitter produces a comparable reduction in perceived experience relative to delay.

This equivalence relationship is consistent with patterns observed in the heatmaps presented in section 4.4.4. The strong vertical gradients in overall QoE and smoothness indicate that ratings degrade more rapidly as jitter increases, whereas increasing delay conditions result only in minor changes to QoE. These findings suggest that variability in jitter is more disruptive to perceived gameplay experience than latency.

5. Conclusion

This study investigated the effects of added network delay and jitter on both objective gameplay performance and subjective quality of experience (QoE) within a controlled first-person shooter (FPS) environment. By varying delay (0-100 ms) and jitter (0-120 ms) and analyzing both independent and combined conditions, this work aimed to determine the extent to which network impairments influence player performance, perceived experience, or both.

Analysis of objective gameplay metrics such as kills per round, shot accuracy, and shots fired did not reveal a meaningful impact on player performance across increasing delay or jitter conditions. Baseline performance distributions of kills and accuracy showed moderate variation across participants, but found no outliers. Under increasing conditions of delay and jitter, mean kills and accuracy per round remained stable, with confidence intervals consistently overlapping across conditions. Additionally, shooting behavior did not change as conditions increased, suggesting that players did not adjust their playstyle in response. Together these findings indicate that moderate levels of impairment did not lead to a measurable difference in overall gameplay performance.

In contrast, subjective quality of experience demonstrated clear sensitivity to impairments, especially jitter. While increasing delay produced minor decreases in overall, responsiveness, and smoothness quality ratings, increasing jitter levels produced a greater decline in QoE. The effect was especially strong for perceived smoothness, where higher jitter levels produced a consistent decline in ratings even in cases where added delay was absent. Heatmap visualizations further supported this pattern, showing strong vertical gradients, corresponding to increases in jitter, while horizontal gradients corresponding to delay remained minimal. These findings indicate that variability in jitter has a stronger perceptual impact than added latency.

To further analyze this difference, an equivalence curve was constructed using isolated conditions. By interpolating jitter-only and delay-only QoE ratings, a relationship between the conditions was estimated. Results showed that the upper bound of delay (100 ms) corresponded to 30-35 ms of jitter in terms of perceived overall QoE. This finding supports the conclusion that QoE is more sensitive to jitter and suggests that smaller magnitudes of jitter produce similar perceptual effects to much larger amounts of delay.

Overall, the results demonstrate a clear distinction between objective performance and subjective experience under delay and jitter. While gameplay performance metrics remained stable across both impairments, subjective perceptions were significantly more sensitive to jitter than delay.

6. Future Work

Future work includes expanding the participant pool to improve the generalizability of our findings. This study primarily recruited students from the WPI community, with many participants reporting moderate FPS experience and familiarity with PC gaming. Recruiting participants across a broader range of age groups, gaming experience levels, and preferred platforms would allow for comparison of impairment tolerance across different user populations. This effort may involve collaboration with additional institutions or adapting the experimental setup for supervised remote participation while maintaining the conditions that are present in the game.

Further analysis of the positional and event-based data that was logged during gameplay could also be analyzed. Although primary metrics such as kills, shots fired, and accuracy were examined, the recorded player and enemy coordinates provide an opportunity for deeper behavioral analysis. For example, reaction time to enemy spawns, engagement distance, player movement patterns, and time-to-kill could be evaluated to determine whether players adapt their behavior under increased delay or jitter conditions. Analyzing these metrics would provide a more complete understanding of how impairments influence gameplay strategy, even when some performance metrics appear stable.

Another direction would involve extending the experimental design to additional game genres. This study focused exclusively on a first-person shooter (FPS) due to its reliance on low-latency input and rapid responsiveness. However, other genres such as racing, platformer, or strategy games may exhibit different sensitivity profiles to delay and jitter. Evaluating these genres would provide insight into whether the dominance of jitter observed in this study is consistent across interaction styles. This could be implemented by modifying alternative Unity-based game templates while maintaining consistent impairment levels and data collection procedures.

Finally, long term future work includes applying these findings towards an adaptive playout buffer configuration in cloud gaming systems. Results from this study suggest that jitter has a stronger impact on overall QoE and smoothness than delay within the tested range. Future cloud gaming platforms may dynamically adjust buffer sizes to prioritize jitter mitigation while maintaining responsiveness. Designing and evaluating adaptive buffering strategies would require integration with a live cloud gaming platform and testing under real network conditions, extending the experimental contributions of this project toward practical system-level optimization.

Together, these future directions build directly upon the completed work, strengthening the analytical depth of the study while expanding its applicability to broader user populations, additional game genres, and real-world cloud gaming system design.

References

1. Peacock, Sean, Quinn Budnick, and Elijah Gray. **"Evaluating Quality of Experience Factors in Cloud Gaming Through a First Person Shooter Game."** *Interactive Qualifying Project*, Worcester Polytechnic Institute, 2025. Advisor: Mark Claypool. https://digital.wpi.edu/pdfviewer/zs25xd997.
2. Rossi, Henrique Souza, Karan Mitra, Tore Myhr, David Lindero, and Niclas Ögren. **"QoE Assessment of PC Cloud-Based Gaming over Heterogeneous Access Networks."** *Proceedings of the IEEE International Conference on Quality of Multimedia Experience (QoMEX)*, Madrid, Spain, 2025. https://doi.org/10.1109/QoMEX65720.2025.11219990.
3. Biswas, Sudipta. **"Improvement to Quality of Experience in Cloud-Based Game Streaming with Jitter Buffer Management."** *Master's Thesis*, Worcester Polytechnic Institute, 2025. Advisor: Mark Claypool. http://web.cs.wpi.edu/~claypool/ms/rumu-25/.
4. Stone, David L., and Kevin Jeffay. **"An Empirical Study of Delay Jitter Management Policies."** *Multimedia Systems*, vol. 2, no. 6, 1995, pp. 267–279. https://doi.org/10.1007/BF01225244.
5. Heineman, Nicholas. **"A Playout Buffer for Moonlight Cloud-Based Game Streaming to Smooth Out Network Jitter."** *Major Qualifying Project*, Department of Computer Science, Worcester Polytechnic Institute, 2024. Advisor: Mark Claypool. http://web.cs.wpi.edu/~claypool/mqp/cloud-buffer-24/.
6. Xu, Xiaokun, Michael Bosik, Adam Desveaux, Alejandra Garza, Alex Hunt, Cameron Person, James Plante, Joseph Swetz, Nina Taurich, Brian Clark, Doris Hung, Philip

Lamoureux, and Mark Claypool. **"Compensating for Latency in Cloud-Based Game Streaming Using Attribute Scaling."** *Proceedings of the IEEE International Conference on Quality of Multimedia Experience (QoMEX)*, Lippstadt, Germany, 2022. https://web.cs.wpi.edu/~claypool/papers/game-scale-qomex-22/paper-final.pdf.

7. Claypool, Mark, and David Finkel. **"The Effects of Latency on Player Performance in Cloud-Based Games."** *Proceedings of the Annual Workshop on Network and Systems Support for Games (NetGames)*, Zagreb, Croatia, 2015. https://web.cs.wpi.edu/~claypool/papers/cloud-games/paper.pdf.
8. Allard, Josh, Andrew Roskuski, and Mark Claypool. **"Measuring and Modeling the Impact of Buffering and Interrupts on Streaming Video Quality of Experience."** *Proceedings of the 18th International Conference on Advances in Mobile Computing and Multimedia (MoMM)*, Association for Computing Machinery, New York City, United States of America, 2021, pp. 153–160. https://doi.org/10.1145/3428690.3429173.
9. Liu, Shengmei, and Mark Claypool. **"EvLag: A Tool for Monitoring and Lagging Linux Input Devices."** *Proceedings of the 12th ACM Multimedia Systems Conference (MMSys)*, Istanbul, Turkey, 2021, pp. 281–286. https://doi.org/10.1145/3458305.3478449.
10. Han, Botao. **"Implementation of a Playout Buffer for a Cloud-Based Game Client."** *Master's Thesis*, Worcester Polytechnic Institute, 2023. Advisor: Mark Claypool. https://digital.wpi.edu/pdfviewer/0r9677105.
11. Anouna, James. **"Network Latency and Cloud Games: A Study Using GamingAnywhere."** *Interactive Qualifying Project*, Worcester Polytechnic Institute.

Advisors: Mark Claypool and David Finkel.
http://web.cs.wpi.edu/~claypool/iqp/game-anywhere/.

12. Desveaux, Adam C., and Vincent S. Courtemanche. **"The Effects of Latency in Commercial Cloud Video Gaming Services."** *Interactive Qualifying Project*, Worcester Polytechnic Institute, 2020. https://digitalcommons.wpi.edu/iqp-all/5701.
13. Oliveira, Michael, Thomas Flanagan, and Carter Nakagawa. **"Effects of Jitter on Quality of Experience in Cloud Gaming."** *Interactive Qualifying Project*, Worcester Polytechnic Institute, 2023. Advisor: Mark Claypool. https://digital.wpi.edu/pdfviewer/6w924g15d.

Appendix

Pre-Study User Survey

User Information

Enter user ID (Proctor will handle) *

Short answer text

How old are you? *

Short answer text

What is your gender? *

- Male
- Female
- Prefer not to say
- Other:

What is your major and graduation year? *

Short answer text

Gaming Background

How many hours per week do you typically spend on gaming? *

Short answer text

Rate your expertise in playing FPS games with a keyboard and mouse *

Enter a number between 1 (low) and 5 (high)

Short answer text

Which gaming platform(s) do you primarily use? (Select all that apply) *

- [ ] PC/Desktop (Keyboard & Mouse)
- [ ] PC/Desktop (Controller)
- [ ] Console (PlayStation, Xbox, Nintendo)
- [ ] Mobile (Smartphone/Tablet)
- [ ] Other: ____________

Informed Consent Form

Informed Consent Agreement for Participation in a Research Study
Advisor: Mark Claypool, **claypool@wpi.edu**
Investigators: Shamit Soni, Brendan Cilley, Huy Le, Biwen Dong
Contact Information: **gr-cloudiqp2025@wpi.edu**
Title of Research Study: Investigating Jitter and Interrupts in a Simulated Game Setting

Introduction
You are being asked to participate in a research study. Before you agree, you must be fully informed about the purpose of the study, the procedures, and any benefits or risks that you may experience as a result of your participation. This form presents information about the study so that you may make a fully informed decision regarding your participation.

Purpose of the study
Users will play a simple first-person shooter (FPS) training game under variable conditions, controlled by the study. Users will report their quality of experience to collect data and better understand the relationship between simulated network conditions and the overall gaming experience.

Procedures to be followed
The duration of the user study will be approximately 30 minutes. During the study, the user will play multiple rounds of the game and provide a rating of their gaming experience after each round. To conclude the study, users will be asked to answer a set of general demographic questions, to better understand the user sample.

Risks to study participants
No increased risks than what would be expected from regular day-to-day computer usage.

Benefits to research participants and others
Users have the opportunity to receive IMGD playtesting credit for their participation. In addition, all participants will be entered into a small raffle to have a chance to win a $15-30 gift card.

Record keeping and confidentiality
Data collected during the study will only be accessed by our team and all records will be anonymized by id. Any presentation of the data will not identify individuals who participated.

For more information about this study or the rights of research participants, reference the contacts at the top of the page and below:
IRB Manager: Ruth McKeogh, Tel. 508 831-6699, irb@wpi.edu
Human Protection Administrator: Gabriel Johnson, Tel. 508-831-4989, gjohnson@wpi.edu

Your participation in this research is voluntary. Your refusal to participate will not result in any penalty to you or any loss of benefits to which you may otherwise be entitled. You may decide to stop participating in the research at any time without penalty or loss of other benefits. The project investigators retain the right to cancel or postpone the experimental procedures at any time they see fit.

By signing below, you acknowledge that you have been informed and consent to be a participant in the study described above. Make sure that your questions are answered to your satisfaction before signing. You are entitled to retain a copy of this consent agreement.

____________________________ __________________
Participant Name (Print) Date

____
Participant Signature

____________________________ __________________
Investigator Signature Date

Participant Recruitment Email

Hello everyone,

Want to play a cloud-based first-person shooter game and enter a raffle for a **$25 Amazon Gift card**?

We're looking for participants for a **25-minute on-campus** user study as part of our **Cloud-based Gaming IQP**.

In this study, you will:

- Complete a brief demographic survey
- Play a first-person shooter game on a PC
- Fill out a short feedback questionnaire

All sessions will take place at **WPI CakeLab (Fuller labs - Room 316)**. Headphones will be provided, but you may bring your own wired pair if you prefer.

All responses are confidential and not linked to personally identifiable information.

IMGD playtesting credit available.

IRB-approved: The study has been reviewed and approved by the WPI IRB and presents no more than minimal risk.

If you're interested, please sign up here: **link**

Have questions? Feel free to reach us at: **gr-cloudiqp2025@wpi.edu**

Thank you for your time and support!