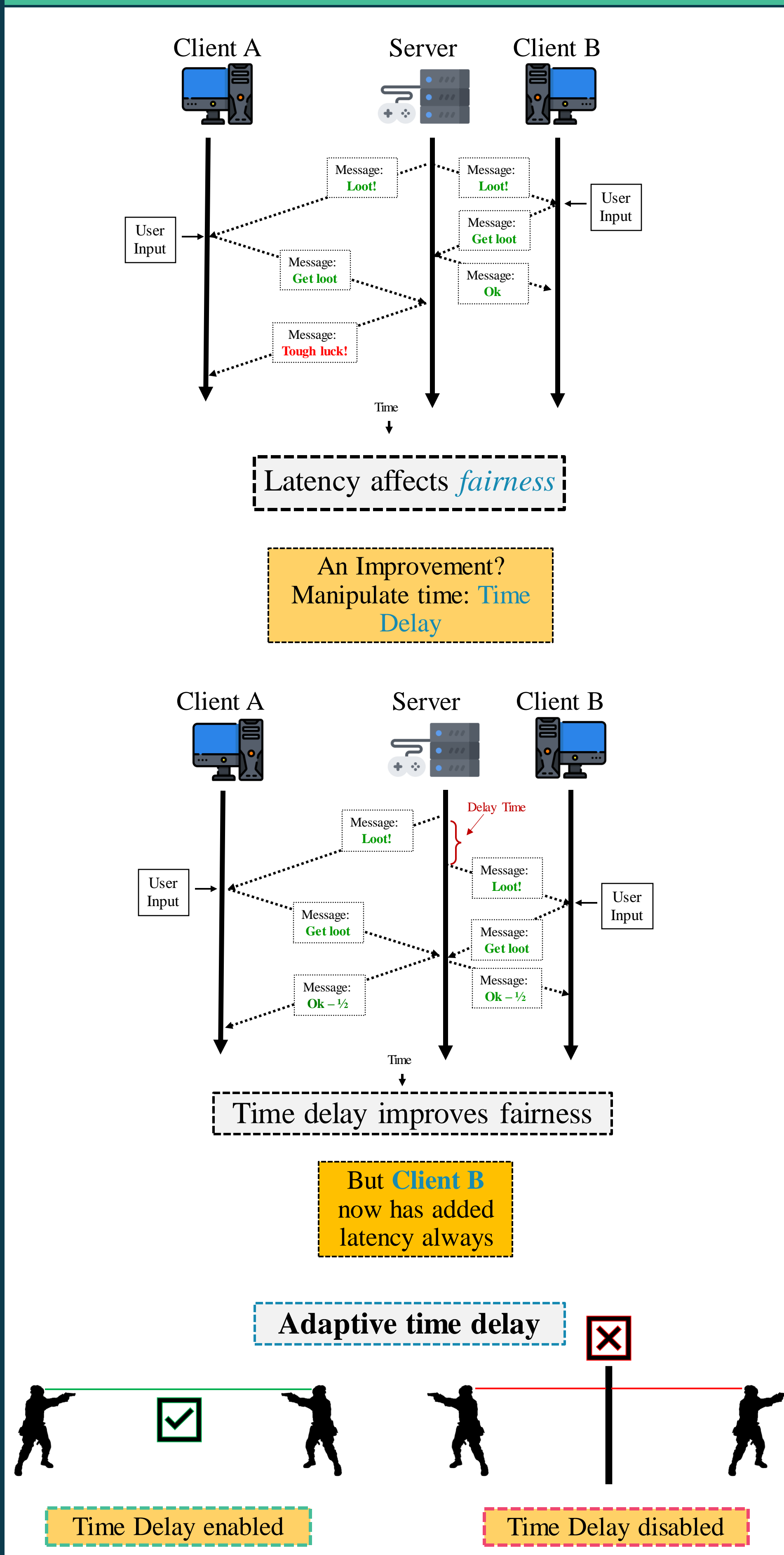


Introduction

- FPS games are affected by network latency
- Higher latency degrades responsiveness
- Latency can affect fairness
- Time delay can improve fairness by making all players experience same latency
- Adaptive time delay** improves responsiveness and player experience by adding latency to the low latency player only **during interaction**

Adaptive Time Delay



Methodology

The Study

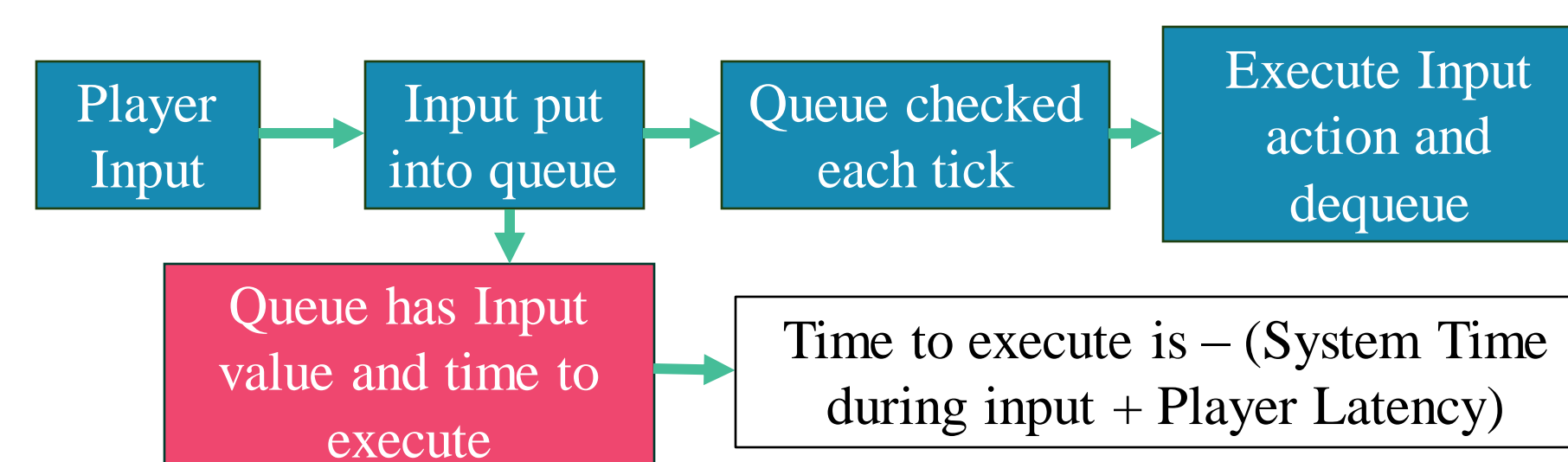
- A single player shooter game was developed
- Latency simulation was added by delaying inputs
- Adaptive time delay was implemented
- Users played short 75s rounds
- Player's performance was measured and logged by the game

The Game - Zombiefield

- Single player zombie shooter made with Unreal Engine 5.1
- 2 Types of zombies were used
 - Standard (Local)
 - Networked
- Input delay was used to simulate latency

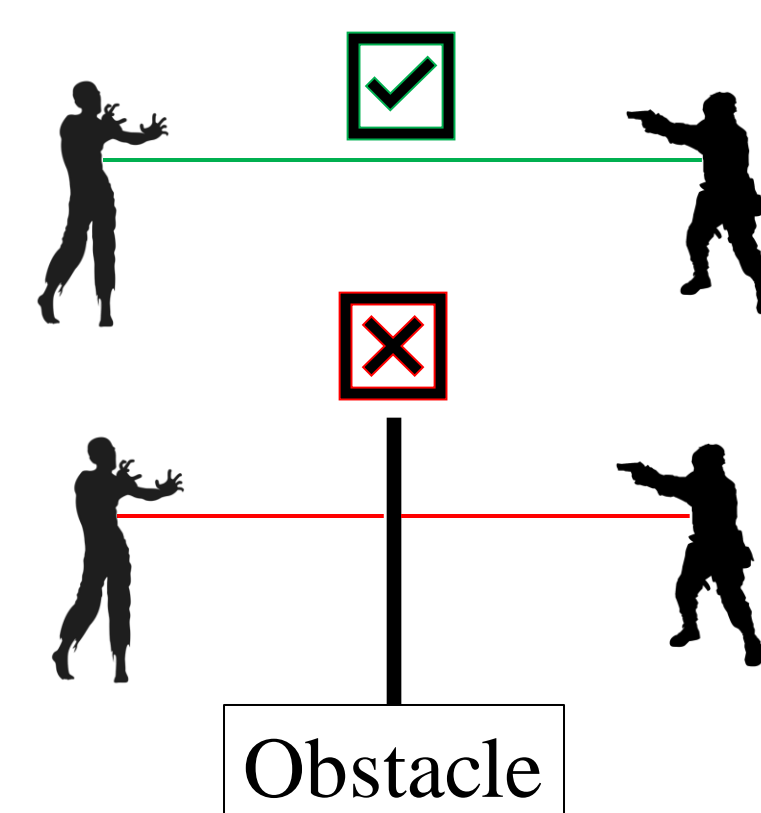
Latency Simulation

- Whenever player had latency, there were input delay on every player action



Adaptive Time Delay

- For adaptive time delay, ray is shot from the players chest to all the networked Zombies chest.



Experimental Design

- Total of 16 main rounds were played with 1 practice round.
- Each round lasts 75 seconds.
- Two questions were asked at the end of each round:
 - Rate your *Quality of Experience* for this round - (1) Low to (5) High – A slider was used.
 - Was this experience acceptable? (Yes/No)
- Configs were shuffled each session
- Latency Conditions: none, fixed and adaptive

Game Screenshot

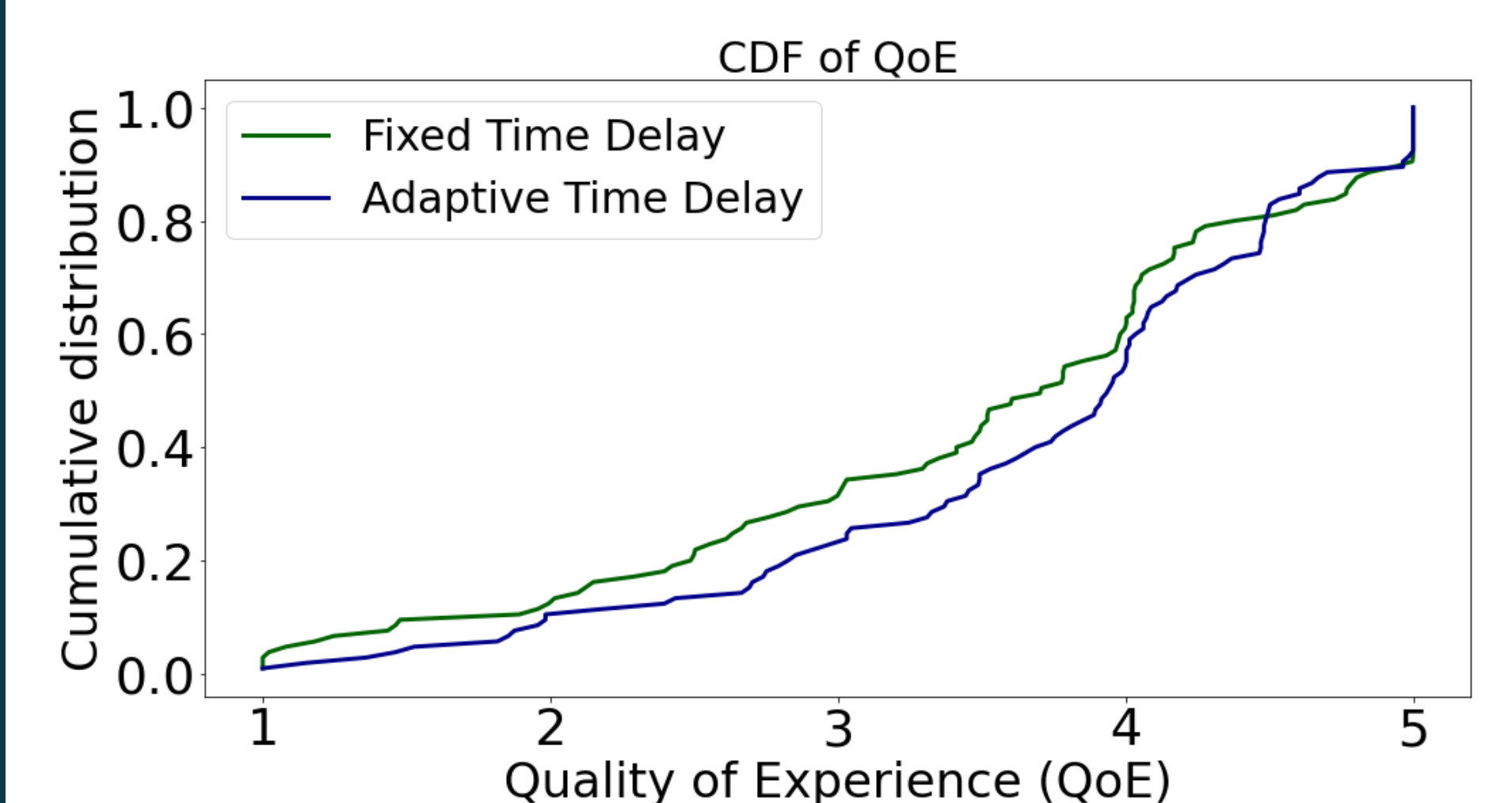
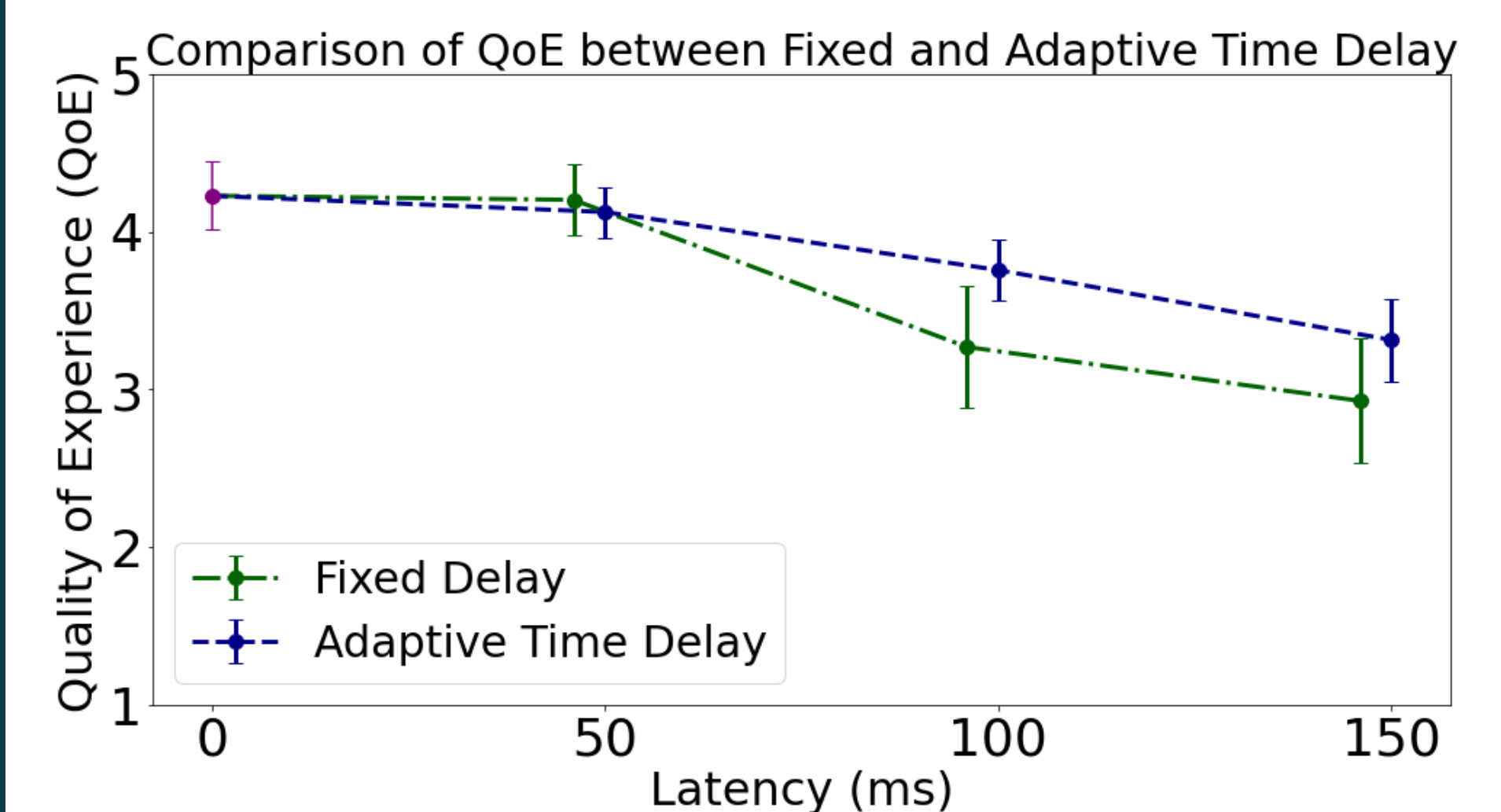


Analysis

Demographics

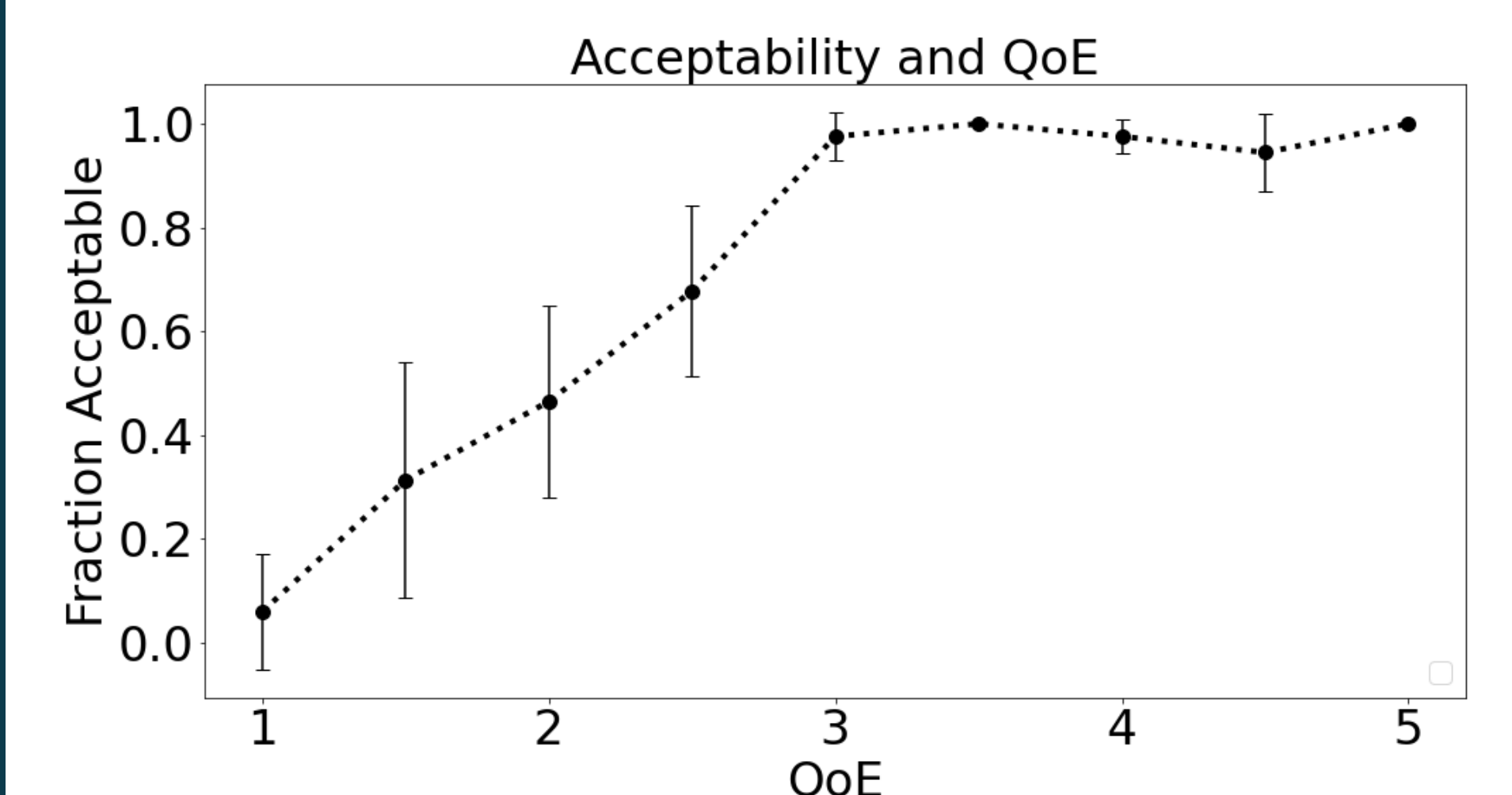
User	Age	Gender	Gaming Skill (1-5)	FPS Skill (1-5)	Reaction Time (ms)
38	20.2 (3.8)	♂36 ♀2	3.7 (0.9)	2.9 (1.1)	195.6 (24.1)

Quality of Experience



Adaptive time delay provides better QoE compared to fixed time delay

Acceptability



Most users found the experience acceptable for QoE above 3

Related Works

Time Delay for Consistency Management

Savery *et al.*: Explored incoming delay for consistency management on servers and clients.

Optimal Server Selection to Improve Fairness

Brun *et al.*: Used heuristics for optimal server selection to enhance fairness with minimal response time increase.

Improving Energy Efficiency and Gameplay Fairness using Time Delay

Kaiser *et al.*: Combined game updates into larger packets in Quake 3 Arena to improve efficiency and fairness in a 7-player study.

Time Delay to Improve Fairness

Zander *et al.*: Used outgoing delay with SAGLU to improve fairness in Quake 2 bot simulations.

Probability Based and Rank Based Time Delay to Improve Fairness

Paik *et al.*: Developed server-based incoming delay adjusted by player count and proximity, using probability-based and rank-based methods to balance responsiveness and fairness in bot simulations.

None of the studies used adaptive time delay on a user study to evaluate its impact on responsiveness

Future Work

Short Term

- Use an advanced detection technique which covers the entire avatar of the opponent
- Thoroughly evaluate various adaptation strategy

Medium Term

- Implement adaptive time delay on large multiplayer game modes
- Improving adaptive time delay activation method to support various types of weapon

Long Term

- Evaluate effectiveness of adaptive time delay for different genres
- Combine adaptive time delay with other latency compensation techniques