# Compensating for Latency in Cloud-based Game Streaming using Attribute Scaling

Xiaokun Xu, Michael Bosik, Adam Desveaux, Alejandra Garza, Alex Hunt, Cameron Person, James Plante, Joseph Swetz, Nina Taurich, Brian Clark, Doris Hung, Philip Lamoureux
Advisor: Professor Mark Claypool

Abstract

Cloud-based game streaming has the disadvantage of added latency from the thin client to the cloud-based server and back, decreasing player performance and degrading their experience. Attribute scaling can make the game easier, potentially exactly counteracting the difficulty added by the latency. We incorporate attribute scaling models into two different games, deploy them on a commercial cloud-based game streaming system and evaluate their efficacy by measuring impact on player performance and Quality of Experience (QoE).
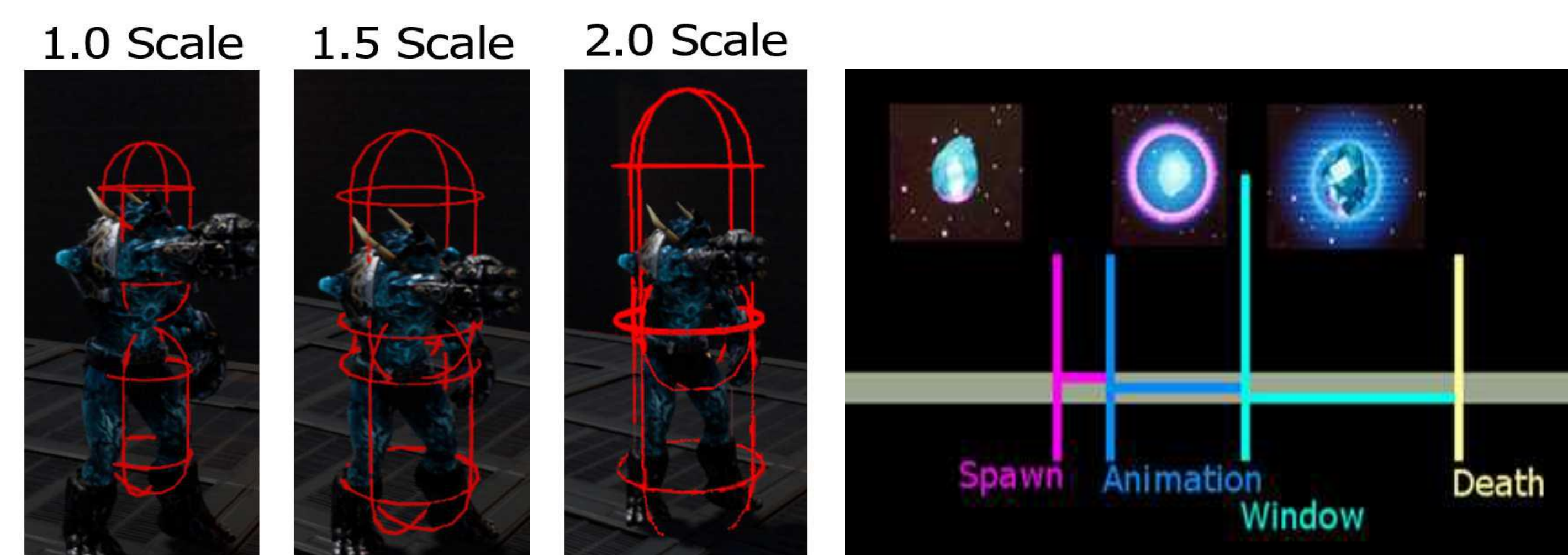
Introduction
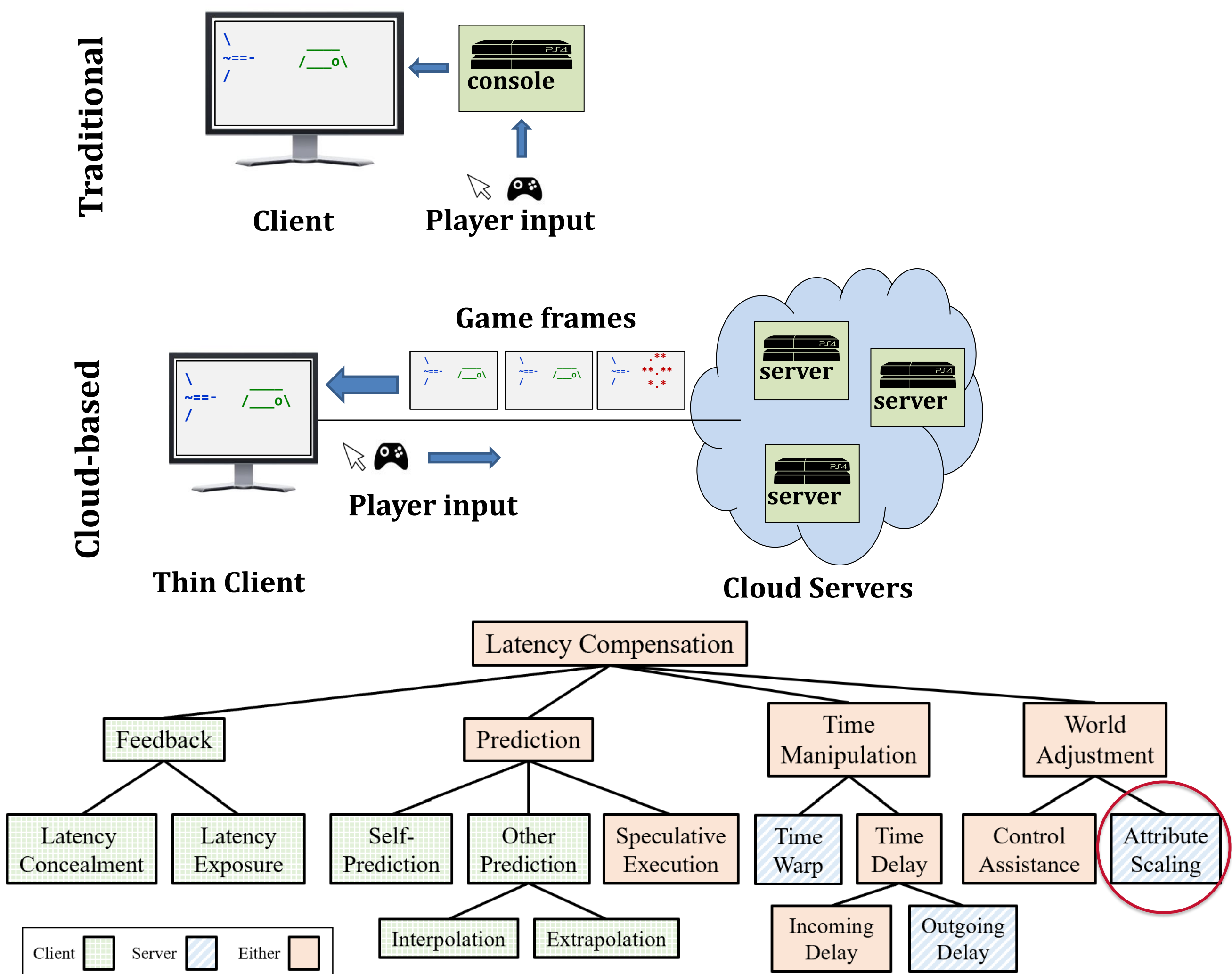
Methodology
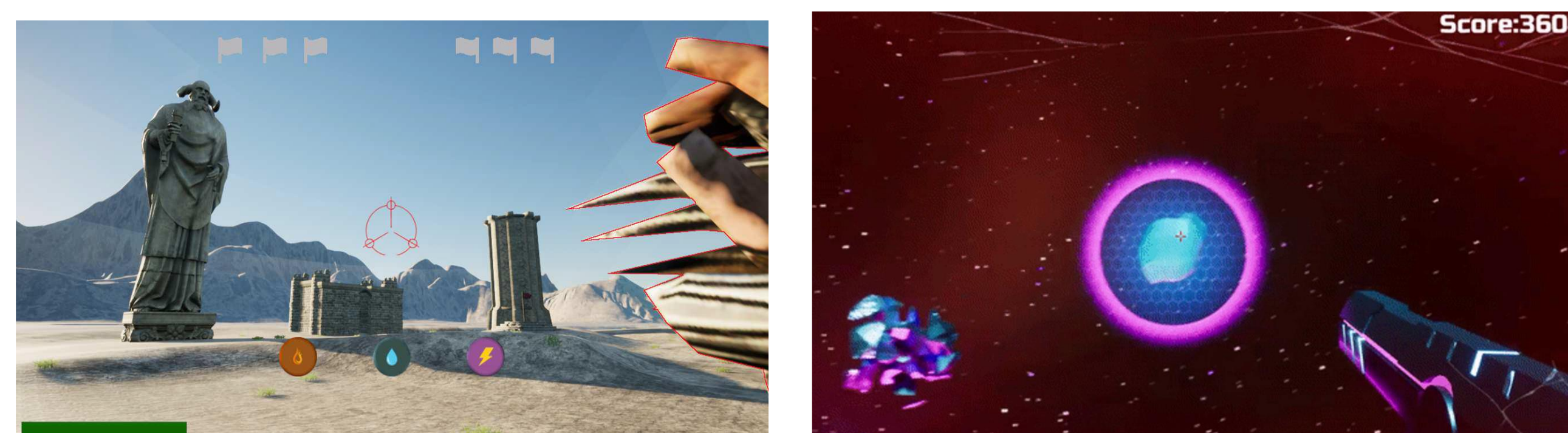
(a) Catalyst – a first person, capture-the-flag shooter game

(b) Nova – a first person, target selection rhythm game

Deploy the games to Google Stadia

Workflow

1. Develop games and attribute scaling

1.0 Scale   1.5 Scale   2.0 Scale

Catalyst – attribute scaling by adjusting hitbox size

Nova – attribute scaling by adjusting window duration

For Catalyst, the hitbox scaling factor (s) is:
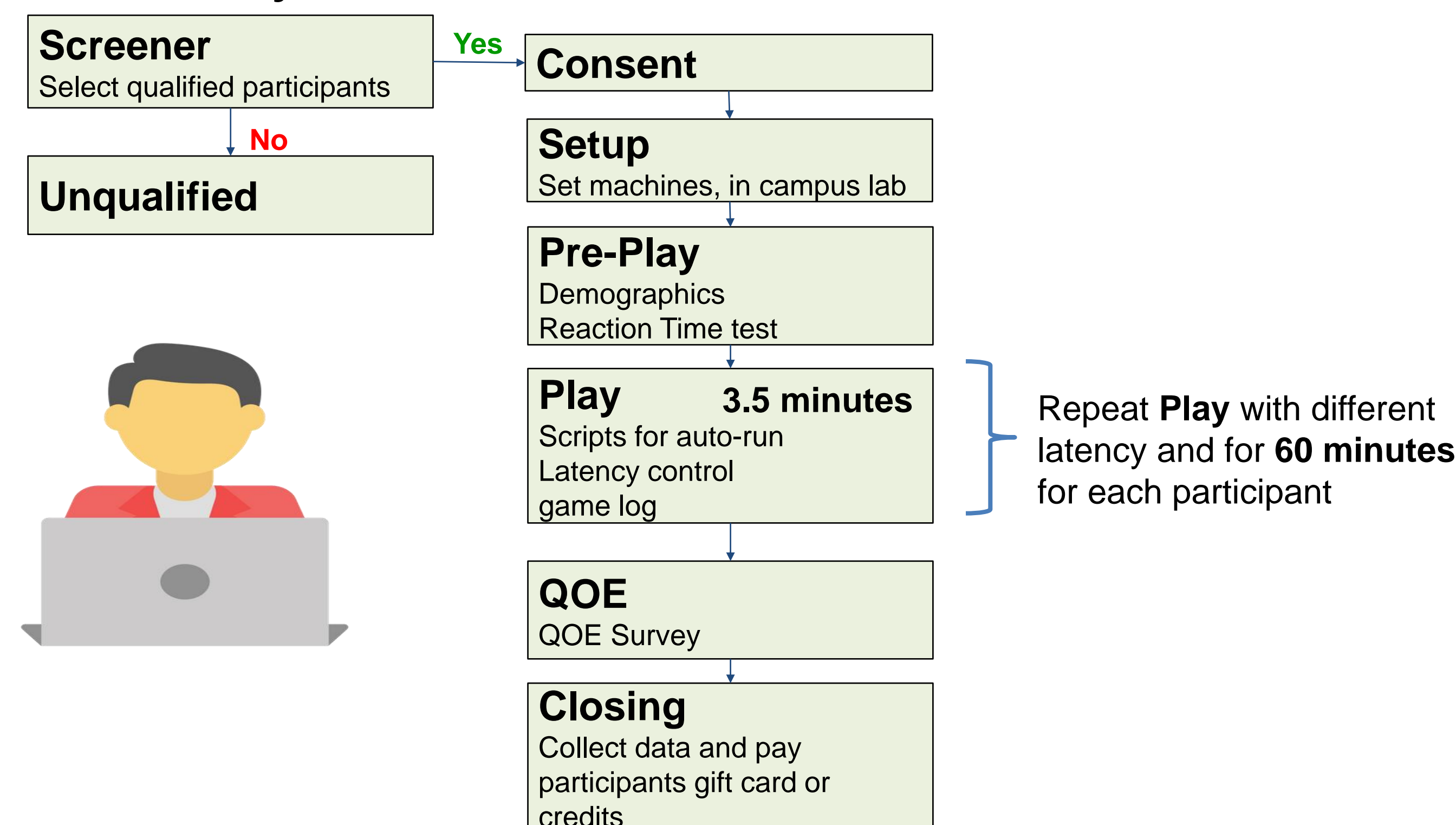$$s = \frac{a + 0.09l + 0.1d - 73}{20}$$
where l is the latency (in milliseconds) and d is the difficulty (the speed of the opponent avatar, in cm/s).

For Nova, the time window scaling factor (s) is:
$$s = \frac{a + 0.1l - 12d + 73}{10}$$
where l is the latency (in milliseconds) and d is the difficulty (the cooldown between notes spawning, in seconds). [1]
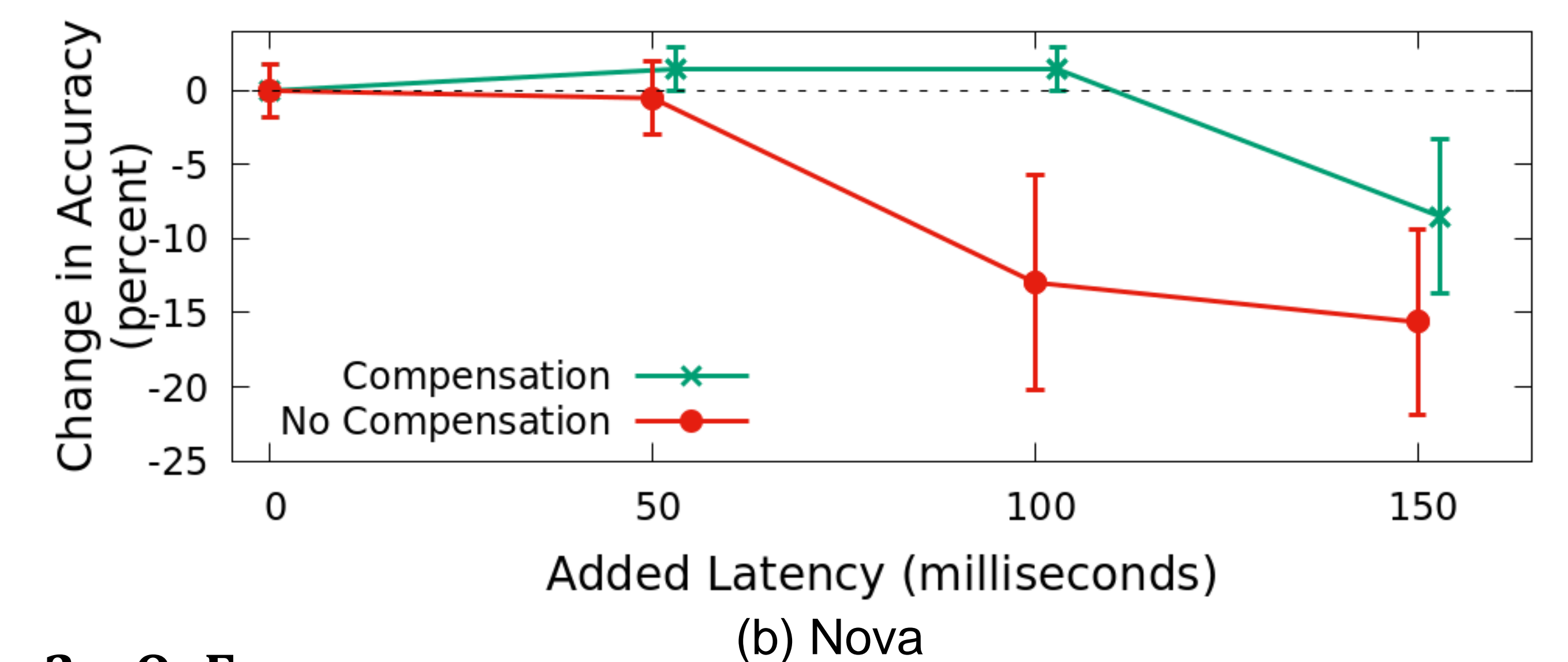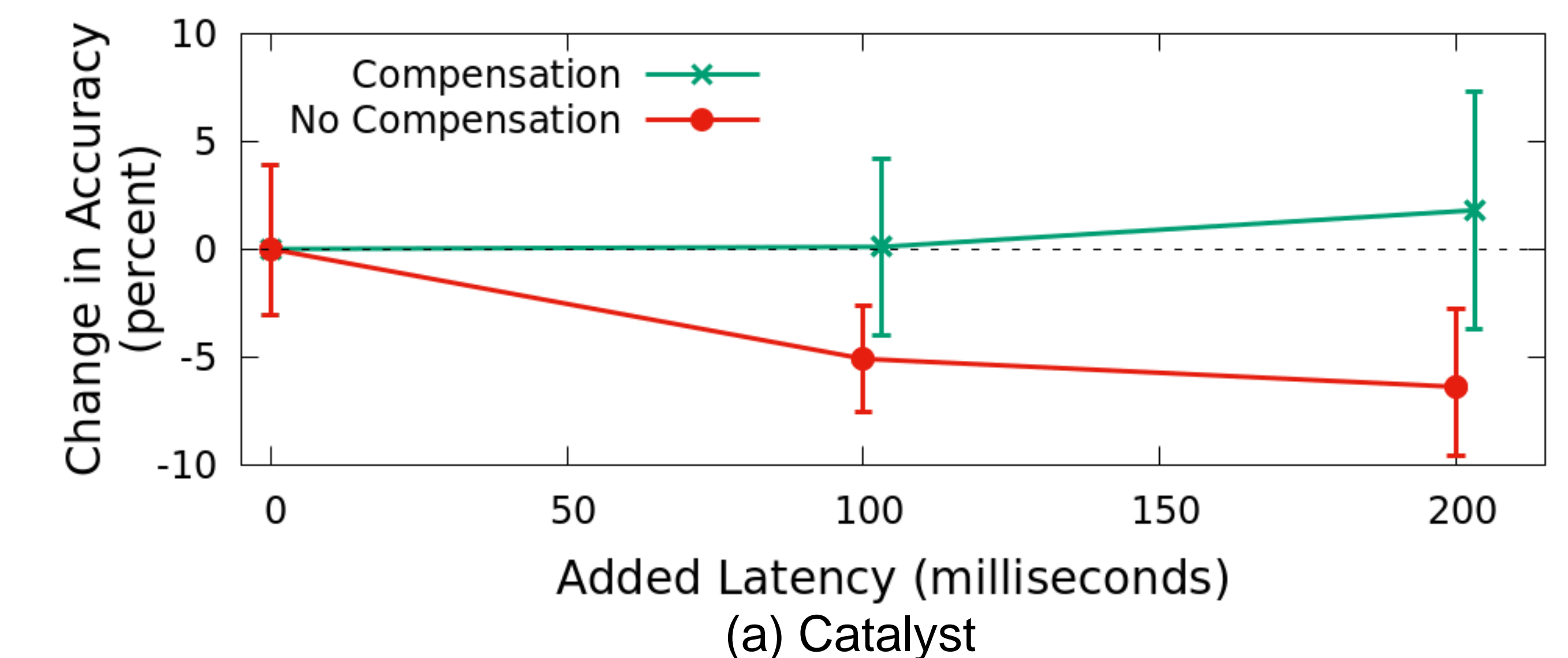
2. User Study

Participant demographics

Study	Users	Age (yrs)	Gender	Gamer
Catalyst	26	21.4 (2.4)	20 ♂ 6 ♀	3.5 (1.3)
Nova	27	20.2 (1.2)	18 ♂ 9 ♀	3.6 (1.3)

Results

1. Performance

(a) Catalyst

(b) Nova

2. QoE

For both Catalyst and Nova, participant QoE decreased with added latency both with and without latency compensation.

Conclusion

- Attribute scaling latency compensation can keep player performance high despite network latency with additional tuning work needed for QoE.

Future Work

- Build attribute scaling into a game engine (e.g., UE4) to scale select game object attributes automatically.
- Explore any other way to improve the QoE, e.g., Buffer policy algorithms
- Understand the connection of QoE and the presence of competing network traffics
- Understand the latency and gaming [2]

[1] Carlson et al. "Towards Usable Attribute Scaling for Latency Compensation in Cloud-based Games", GameSys 2021.
[2] Xu et al. "The Effects of Network Latency on Counter-strike: Global Offensive Players", QoMEX 2022.