Ph.D. Dissertation Defense

The Impact of Latency on Players in First-person Shooter Games

Committee:
Prof. Mark Claypool, Worcester Polytechnic Institute
Prof. Carl Gutwin, University of Saskatchewan
Prof. Lane Harrison, Worcester Polytechnic Institute
Doc. Jamie Sherman, Atlassian

Student:
Shengmei Liu

Worcester Polytechnic Institute
Latency

Local latency

Network latency

Performance
Latency and gamers

System delay reduced from 125 ms to 25 ms:

- Performance improved by 25\%\footnote{1}
- Pro players: Rank 50 $\rightarrow$ RANK 3\footnote{2}
- Competitive rank $\uparrow\uparrow\uparrow\uparrow 8$

[Liu et al., 2021]
Motivation

Games

Weapons

Maps

Skills
Approach

1. Gather data
2. Build models
3. Simulate scenarios using models
4. Validate simulations
5. Explore game scenarios

Repeat for each action
Related publications

- Navigation, CHI 2022
- Selection, MMSys 2023
- CSGO local latency, CHI 2021
- CSGO network latency, QoMEX 2021
Gaming actions - Selection
Gaming actions - First-person navigation
Related work – Gaming actions

Selection:
[JT14, CER17, LG18, LG19]
- Evaluated latency on 2D selection tasks
- Latency has significant impact on player performance

Navigation:
[Dru71, AZ97, AZ99, AZ01, ZAW04, KGS07, KS16]
- Proposed or enriched steering law
- Steering law can accurately predict player performance
Related work – Latency

**Latency:**

[PW02, Arm03, QML+04, DWW05, FRS05, CC06, CC07, AJG+13, HCW+14, ISGS15, HFPG16, CER17, ERC18, LG18, LG19, LKS+21b]

- Studies on specific game or game tasks
- Significant impact on players
Related work – Latency compensation techniques

**Time warp & Self-prediction:**

[Jef85, Mau00b, Mau00a, WO00, MVHE04, JSB05, SK05, BK06, BSB06, CCC+07, TAS07, SGG10, LC15, LC17, LSGH17, AMC18, LC18, SC19]

- Proposed, enriched and/or evaluated time warp
- Both techniques can significantly improve player performance with latency
Approach

1. Explore game scenarios
2. Simulate scenarios using models
3. Validate simulations
4. Explore game scenarios

Repeat for each action

- Gather data
- Build models
- Navigation
- Selection
- Client-server architecture

- Consistent opponent (me)
Navigation user study

• Local latency

Gaming system: 0, 22
Normal PCs: 100, 175
Console, TV, etc.: 200, 300

• Network latency

Good Internet Connection: 0, 25
Moderate Internet connection: 100, 200
Other: 100, 300

3 trials for each latency → 27 rounds
40 sec per round → ~ 30 min total
Navigation demographics

- 30 users

<table>
<thead>
<tr>
<th>Users</th>
<th>Age (yrs)</th>
<th>Gender</th>
<th>Gaming per week (hours)</th>
<th>Game Self-rating</th>
<th>FPS Self-rating</th>
<th>Reaction-time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>23.1 (4.0)</td>
<td>26♂ 4♀</td>
<td>10.4 (8.3)</td>
<td>3.4 (1.1)</td>
<td>3.1 (1.0)</td>
<td>227.2 (40.0)</td>
</tr>
</tbody>
</table>
A decrease in total latency by 100 ms improves player performance by 11 percent

$R^2 = 0.93$

$P < 0.001$
Navigation distributions

- Seeker intervals, hider intervals

- Exponential
- Stretched exponential
- Weibull
- Pareto
- Log-gamma
Navigation models

- Seeker intervals, hider

\[ p = 1 - \exp\left(-\frac{\text{in.sight.window}}{a \times (\text{loc.lat} + \text{net.lat}) + b} + c x (\text{loc.lat} + \text{net.lat}) + d + e\right) \]

\[ R^2 = 0.99 \]

\[ p = 1 - \exp\left(-\frac{\text{out.of.sight.window}}{0.5^{0.72}}\right) \]

\[ R^2 = 0.99 \]
Approach

1. Explore game scenarios
2. Simulate scenarios using models
3. Validate simulations
4. Explore game scenarios

Repeat for each action:
1. Gather data
2. Build models
3. Selection
Approach

1. **Gather data**
2. **Build models**
3. **Simulate scenarios using models**
4. **Validate simulations**
5. **Explore game scenarios**

*Repeat for each action*
Simulation

Game:
• 2 player game
• Goal: Kill the opponent as fast as possible
• Whoever kills the opponent first wins the game
Simulation - overview

Player 2
5 hits first -- > winner!
Simulation pseudocode

fpsSimulation()  
  GameInfo  
  PlayerInfo  
findWinnerForaGame()  
  GameInfo  
  PlayerInfo  
getHitTimestamps()  
getWindow(is_hider)  
...

deriveSampleFromModel(model_type)
  getOutOfSightWindow()
  getInSightWindow()
  getSelectionTime()

getHitTimestampsInAWindow(
  window_start_time,
  window_end_time, hit_count)
Simulation pseudocode

```javascript
function getHitTimestampsInAWindow (window_start_time, window_end_time, hit_count, gameInfo, playerInfo) {
}
```
Simulation pseudocode

```plaintext
function getHitTimestampsInAWindow (window_start_time, window_end_time, hit_count, gameInfo, playerInfo){
    hit_timestamps_window = []
    total_time = window_start_time #Timeline
}

Model:
\[ p = 1 - a \times e^{-(b \cdot L + c) \cdot T} \]
```
Approach

1. Gather data
2. Build models
3. Simulate scenarios using models
4. Validate simulations
5. Explore game scenarios

Repeat for each action

Navigation
Selection
Validation user study

Map

Game screenshot

Evlag
(Local latency)

Netem
(Network latency)

FPS

Server

Client

Client
(control)
Validation user study

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Test player values</th>
<th>Control player values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local latency</td>
<td>25, 100 (ms)</td>
<td>25 (ms)</td>
</tr>
<tr>
<td>Network latency</td>
<td>0, 150 (ms)</td>
<td>0 (ms)</td>
</tr>
</tbody>
</table>

Parameters | Values (same for both players) |
---         |-------------------------------|
Latency compensation | none, both time warp and self-prediction |
Firing rate | 250, 1000 (ms) |
Number of hits required | 1, 4 |
Target size | 50, **200** (cm) |
Movement speed | 5, **10** |
Map size | 18 x 18 m, **36 x 36** m |

Conditions
A. Best condition (No latency, others default)
B. Vary local latency, network latency, latency compensation
C. Vary 5 game parameters under no latency and compensation
D. Vary 5 game parameters under 150 ms latency and no compensation
E. Vary 5 game parameters under 150 ms latency and compensation on

\[ A \times 3 + (B + C + D + E) \times 2 \]
= 39 rounds
+ 2 practice
~ 30 min total
Validation Demographics

T test: p = 0.69

<table>
<thead>
<tr>
<th>Group</th>
<th>Users</th>
<th>Age (yrs)</th>
<th>Gender</th>
<th>Gaming per week (hours)</th>
<th>Gamer Self-rating</th>
<th>FPS Self-rating</th>
<th>Reaction-time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>46</td>
<td>19.3 (3.0)</td>
<td>37♂ 6♀ 3 Other</td>
<td>13.6</td>
<td>3.4 (1.1)</td>
<td>2.8 (1.2)</td>
<td>183.3 (29.5)</td>
</tr>
<tr>
<td>Control</td>
<td>38</td>
<td>20.7 (3.9)</td>
<td>29♂ 8♀ 1 Other</td>
<td>12.8</td>
<td>3.5 (1.2)</td>
<td>3.1 (1.2)</td>
<td>196.1 (32.4)</td>
</tr>
</tbody>
</table>
Validation results

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Test player</th>
<th>Control player</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network latency</td>
<td>150 (ms)</td>
<td>0 (ms)</td>
</tr>
<tr>
<td>Local latency</td>
<td>25 (ms)</td>
<td>25 (ms)</td>
</tr>
<tr>
<td>Player skill</td>
<td>all</td>
<td>all</td>
</tr>
<tr>
<td>Latency compensation</td>
<td>none</td>
<td></td>
</tr>
<tr>
<td>Firing rate</td>
<td>250 (ms) (4 shots per sec)</td>
<td></td>
</tr>
<tr>
<td>Number of hits required</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Target size</td>
<td>200 cm</td>
<td></td>
</tr>
<tr>
<td>Map size</td>
<td>36 x 36 m</td>
<td></td>
</tr>
</tbody>
</table>
Validation results

- Our simulation predicts scenarios in the custom FPS game well.
Validation Results

- Our simulation predicts scenarios in the custom FPS game well.
Approach

1. Gather data
2. Build models
3. Simulate scenarios using models
4. Validate simulations
5. Explore game scenarios

Repeat for each action
# Base Explorations

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local latency</td>
<td>25 (ms)</td>
</tr>
<tr>
<td>Player skill</td>
<td>all</td>
</tr>
<tr>
<td><strong>Latency compensation</strong></td>
<td>none</td>
</tr>
<tr>
<td>Firing rate</td>
<td>250 (ms) (4 shots per sec)</td>
</tr>
<tr>
<td>Number of hits required</td>
<td>1</td>
</tr>
<tr>
<td>Target size</td>
<td>200 cm</td>
</tr>
<tr>
<td>Map size</td>
<td>36 x 36 m</td>
</tr>
<tr>
<td>Network latency</td>
<td>0 - 300 ms</td>
</tr>
</tbody>
</table>
Exploration

- Both techniques together can nearly completely overcome the effects of network latency on player performance.
# Exploration

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default values</th>
<th>Win rate %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network latency</td>
<td>150 ms</td>
<td>24.98</td>
</tr>
<tr>
<td>Local latency</td>
<td>25 (ms)</td>
<td>24.98</td>
</tr>
<tr>
<td>Latency compensation</td>
<td>none</td>
<td>24.98</td>
</tr>
<tr>
<td>Number of hits</td>
<td>1</td>
<td>24.98</td>
</tr>
<tr>
<td>Player skill</td>
<td>all</td>
<td>24.98</td>
</tr>
<tr>
<td>Firing rate (w/ 10 hits)</td>
<td>4 (shots/sec)</td>
<td>4.62</td>
</tr>
<tr>
<td>Map size</td>
<td>36 x 36 m</td>
<td>24.98</td>
</tr>
<tr>
<td>Target size</td>
<td>200 cm</td>
<td>24.98</td>
</tr>
<tr>
<td>Firing rate (w/ 1 hit)</td>
<td>4 (shots/sec)</td>
<td>24.98</td>
</tr>
</tbody>
</table>

- **Large Impact on latency**
- **Medium impact on latency**
- **Small impact on latency**
Future work

Further enrich models and simulations

Validate with more games and scenarios

Apply the same methodology to study different game genres
Conclusions
Contributions

• Navigation
  – Impact of latency
  – Models of in-sight and out-of-sight time windows

• Selection
  – Impact of latency
  – A model of elapsed time

• Simulations
  – Validated
  – Parametrized by many FPS parameters

• Explorations

[Call of Duty, Activision, 2003]
THANKS FOR YOUR ATTENTION!

ANY QUESTIONS?
An increase in total latency by 100 ms decreases player QoE by half a point on a 5-point scale.


Results Prune

• CDF shapes are similar between two datasets
• Intervals in the FPS game study are longer in general
Simulation

- Win rate mean and standard deviation are more stable with number of iterations over 10,000.
- We use 100,000 as the number of iterations in following explorations.
Validation with CS:GO

- Player performance degrades linearly in the range of 25 - 125 ms
- Player performance improves about 34% in our simulation and 25% in the CS:GO study
User study - navigation

Network latency

Local latency

Performance
Latency and gamers

[Call of Duty, Activision, 2003] [Claypool and Claypool, 2010] [Liu et al., 2021]
Related work – Players

**Reaction time:**
[Ric14, Huma, Kos08, TG05, BS99, HPM91, Whe0]
- Gamers tend to have shorter reaction time

**Player skill:**
[Cla18, AJG+13, DWW05]
- Higher skill players are more resilient to latency
Related publications (In my thesis)

Chapter 5

Chapter 6

Chapter 8

Chapter 8
Related publications (Inform my thesis)

**Latency**


**Latency**


**Latency compensation**

Related publications (Inform my thesis)

**Player skill**

Shengmei Liu, Atsuo Kuwahara, James Scovell, Jamie Sherman, and Mark Claypool. "L33t or N00b? How Player Skill Alters the Effects of Network Latency on First Person Shooter Game Players." 2021 Proceedings of the Workshop on Game Systems (GameSys' 21).

**Player skill**


**Methodology**

An increase in total latency by 100 ms degrades seeker time by 1.5 seconds per minute.

The ability of a player to hide from an opponent is not significantly impacted by latency.
Models

- Selection

\[ p = 1 - a \times \exp\left(-b \times (\text{loc.lat} + \text{net.lat}) + c \times \text{firing.rate} + d \times 3D\_distance \right. \\
\left. + e \times \frac{\text{target.size}}{3D\_distance^2} + f \right) \times \text{elapsed.time} \]

\[ R^2 = 0.98 \]
Exploration

- Both techniques together can nearly completely overcome the effects of network latency on player performance.

- Player performance is more impacted by latency the greater the number of hits required for a kill.
Gaming actions - Selection

[Call of Duty, Activision, 2003]

[Duck Hunt, Nintendo, 1984]

[League of Legends, Riot Games, 2009]

...
Gaming actions - First-person navigation

Position A → Position B
Related work – Latency

**Network latency:**

[PW02, Arm03, QML+04, DWW05, FRS05, CC06, AJG+13, HCW+14, HFPG16].
- Studies on specific game or game tasks
- Significant impact on players

**Local latency:**

[CC07, ISGS15, CER17, ERC18, LG18, LG19, LKS+21b].
- Studies on specific game or game tasks
- Significant impact on players
Validation results

- Latency has significant impact on player performance
- Latency compensation can significantly mitigate the effects of latency on player.
- More hits make it harder for players with network latency to win
- Room size and firing rate do not have significant impact on player win rates.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Test player</th>
<th>Control player</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network latency</td>
<td>150 (ms)</td>
<td>0 (ms)</td>
</tr>
<tr>
<td>Local latency</td>
<td>25 (ms)</td>
<td>25 (ms)</td>
</tr>
<tr>
<td>Player skill</td>
<td>all</td>
<td>all</td>
</tr>
<tr>
<td>Latency compensation</td>
<td>none</td>
<td></td>
</tr>
<tr>
<td>Firing rate</td>
<td>250 (ms) (4 shots per sec)</td>
<td></td>
</tr>
<tr>
<td>Number of hits required</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Target size</td>
<td>200 cm</td>
<td></td>
</tr>
<tr>
<td>Map size</td>
<td>36 x 36 m</td>
<td></td>
</tr>
</tbody>
</table>
Latency and gamers

[Call of Duty, Activision, 2003]

[Liu et al., 2021]
## Base Explorations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Test player</th>
<th>Control player</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network latency</td>
<td>0 - 150 (ms)</td>
<td>0 (ms)</td>
</tr>
<tr>
<td>Local latency</td>
<td>25 (ms)</td>
<td>25 (ms)</td>
</tr>
<tr>
<td>Player skill</td>
<td>all</td>
<td>all</td>
</tr>
<tr>
<td><strong>Latency compensation</strong></td>
<td><strong>none</strong></td>
<td></td>
</tr>
<tr>
<td>Firing rate</td>
<td>250 (ms) (4 shots per sec)</td>
<td></td>
</tr>
<tr>
<td>Number of hits required</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Target size</td>
<td>200 cm</td>
<td></td>
</tr>
<tr>
<td>Map size</td>
<td>36 x 36 m</td>
<td></td>
</tr>
</tbody>
</table>