Grid Battle
A turn-based strategy network game
Junjie Gu
Introduction

- Add multiplayer aspect
- Over local area network
- Evaluate different network conditions
Outline

- Introduction
- Background
- Design
- Implementation
- Results
- Conclusions
Turn-Based Strategy Game

- Turn based
- Chess like
- Omnipresent
- example: Sid Meier’s Civilization
iOS LAN game

- wireless local area network
- peer to peer architecture
- high transfer rate
- widely used on iOS
Design

• Actions include move, attack, spell magic, rest and screen moving

• Each player can execute actions on 2 units on each turn

• A timer constrains actions done in 30 seconds
Network Framework

- P2p communication architecture
- Not simultaneously, instead one way communication
- Using GKSession API
GKSession

- discover and connect nearby device
- using both Wi-Fi or bluetooth
- also provides basic VoIP
- don’t know network protocol
Synchronization method

- Touch events not sent
- Only sends actions data
- Timer to synchronize time
Framework

- Screen
- moving

- Move
- Attack
- Magic
- Rest
- ...

- timestamp
- position
- action index

Monday, April 22, 13
Framework

timestamp
position
action index

Screen
moving

Move
Attack
Magic
Rest
...

Monday, April 22, 13
Jitter affect screen moving

<table>
<thead>
<tr>
<th>Server</th>
<th>Send Time</th>
<th>Receive Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Client</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
no buffer
Client side buffer

- not process directly on receiving data, instead push into buffer
- Pop out periodically (0.01s/0.05)
- hypothesis: should improve smoothness while increase delay
using buffer
Other concerns

• Packet loss will be found if game states varies.

• Jitter doesn’t affect menu actions, does affect screen moving.

• Bandwidth should be small
Evaluation

- 2 iPhone 5 with iOS 6.1.2
- 3 users
- 1 minute for each test
- use router Wi-Fi, p2p Wi-Fi and bluetooth
- use 0s, 0.01s, and 0.05s buffer timer
Conclusions

- Users can play against each other in LAN without bug
- Bluetooth and p2p Wi-Fi works well
- no packet loss because states are always the same
- delay is not a problem for this game
- Client side buffer works well
Future work

- Add more players
- Use message aggregation
Thank you for your questions?