Application Assisted Power Management in Multiplayer Mobile Games

Bhojan Anand‡, A. L. Ananda‡, Mun Choon Chan‡, Rajesh Krishna Balan†
‡National University of Singapore and †Singapore Management University

Introduction

Parallel growth Converge to next "Killer" application

- Online games – phenomenal increase in market share
- The number of cell phone users continues to escalate
- Multiplayer Mobile Games – Next "Killer" application?!!

Usability Constraint:

- High Power Consumption
  - Processor load, Continuous NW traffic, Display intensity

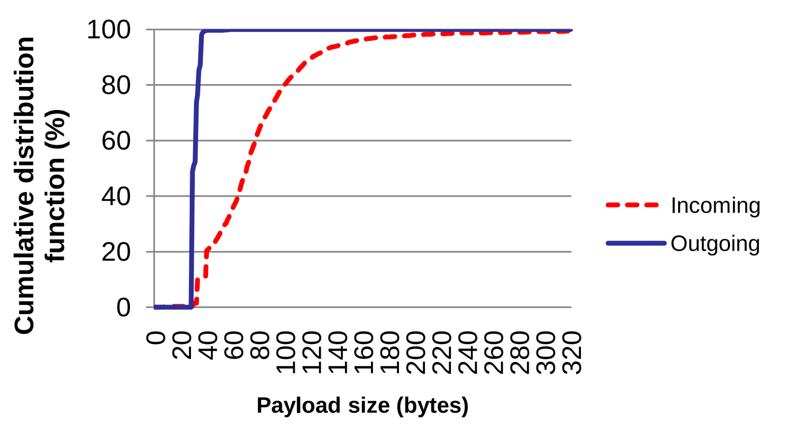
Objective:

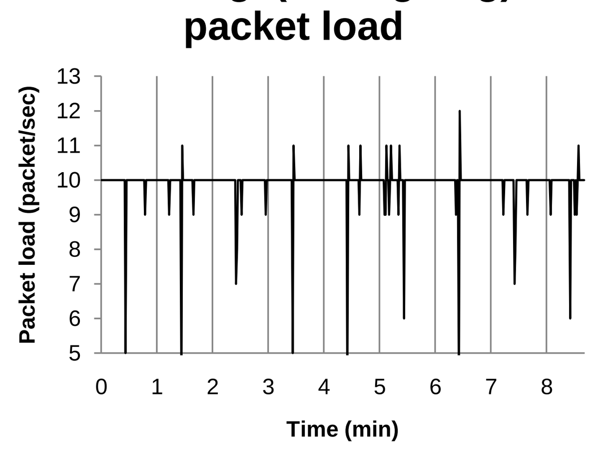
- Develop an energy saving scheme for multiplayer mobile gaming

Characteristics of Networked Games

Game

- Average bandwidth per client 7Kbps
- Average packet rate (incoming and outgoing): 20 pps
- Very small packet size, Average: 24 DATA +8 UDP + 20 IP Bytes





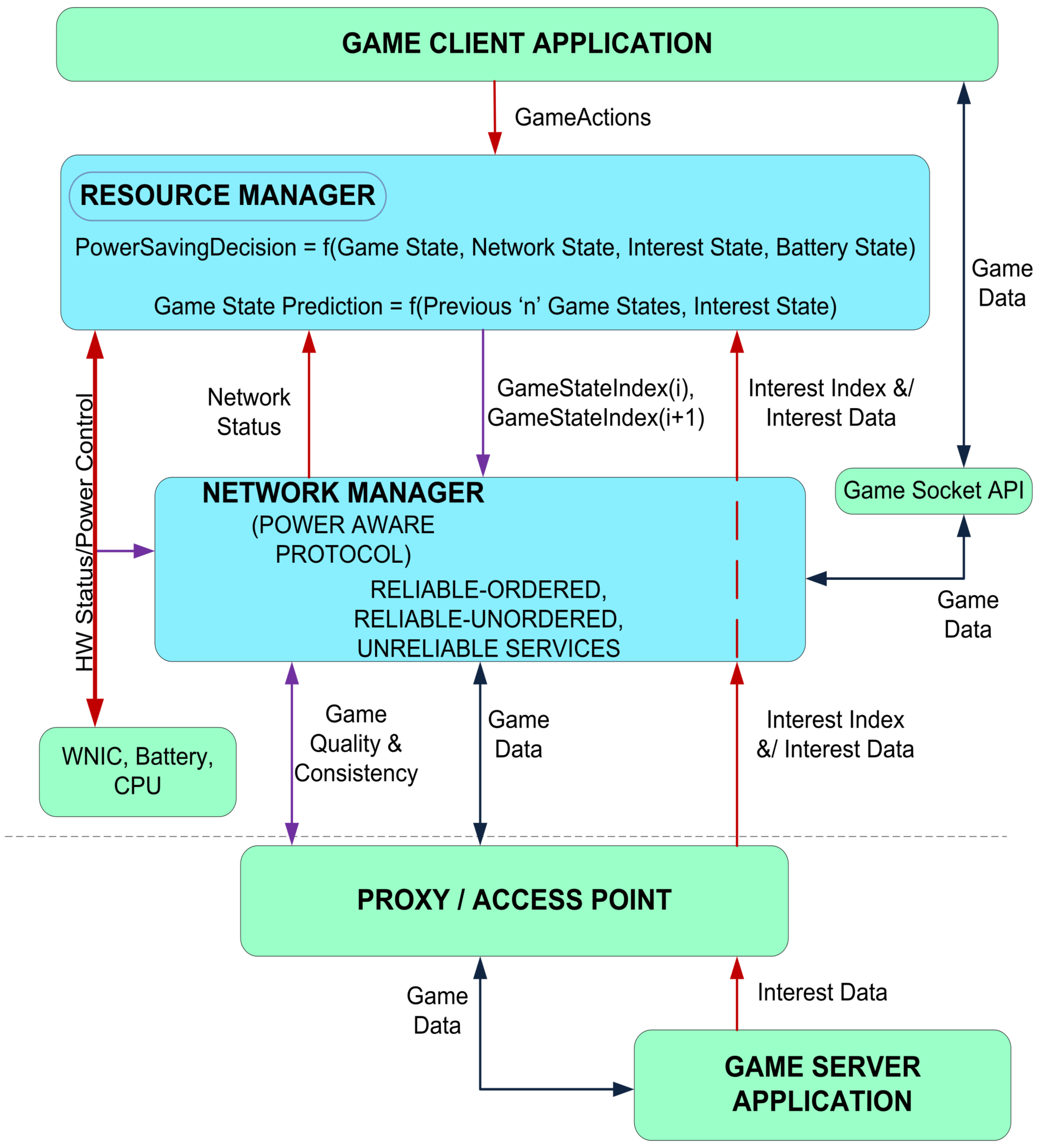
* Based on *Quake Mobile* and *Armageddon Mobile* (5-20 players) in our test bed

Saving WNIC Power consumption

Safe sleep mode

- Acceptable latency - 200 ms for FPS and 1500 ms for RPG (slow speed) with Dead Reckoning.
- Challenge 1: When to sleep without affecting the quality of game play?
  - When the actions are not significant for the quality of the game play.
- Challenge 2: How long is the sleep duration?
  - Long duration effects the quality of game
  - Short duration will result in more overhead power consumption
- Complex algorithm results in adverse results!

Architecture & Information Flow



Algorithms

Game State

$$GameStateIndex_i = f(w_{i1}.gameActionIndex_i, w_{i2}.interestIndex_i)$$

Network State

$$NetworkStateIndex_i = f(latency_i, bandwidth_i)$$

Interest State

$$InterestStateIndex_i = f(AoI_i, GameZone_i, AvatharProximity_{i-k...i}, NPC\ Proximity_{i-k...i})$$

Battery State

$$BatteryStateIndex_i = f(remainingBatteryLife_i)$$

Game State Prediction

$$GameStateIndex_{i+1} = f_1(w_{i1}.gameActionIndex_{i+1}, w_{i2}.interestData_i)$$

$$where,\ gameActionIndex_{i+1} = f_2(interestData_i, w_{i5}.gameActionIndex_i, w_{i6}.gameActionIndex_{i-1}, w_{i7}.gameActionIndex_{i-2}, \ldots, w_{i8}.gameActionIndex_{i-k})$$

- High probability for important actions,
  - When the Avatar is near another Avatar or NPC
  - When Avatars and NPCs are approaching the player
- Avatar will have different actions in different zones

Results & Conclusion

Average Amount Energy Saved in Various Modes:

- With current implementation in Android platform (*Mobile Armageddon* - RPG game)
  Full Mode (**35%**), Secured Mode (No Server Hints -> **20%**), Black-box Mode (No Application Hints -> **13%**)

We have:

- Learnt about Characteristics of Network Gaming
- Developed Linear Prediction Algorithm to predict "Importance of Game State"
- Developed Algorithm for Power Management using Application & Environment hints
- We have developed an API extension for existing game engine to learn about the entire game state for RPG games