QoS Control for Sensor Networks

R. Iyer & L. Kleinrock
Dept. Computer Science
UCLA

ICC'03

Introduction

- Study of QoS in sensor networks
- QoS def. in this paper: *sensor network resolution*
- Ideas
  - Use Gur Game, a mathematical paradigm
  - BS communicates QoS info. to sensors using broadcast channels
- Result: BS dynamically adjust the number of sensors being activated

Problem Statement

- QoS = sensor network resolution = optimal # of active sensors
- Goals:
  - Maximize life time of sensors by turning off sensors
  - Have enough sensors active and report data
- Use Gur game control theory [13]

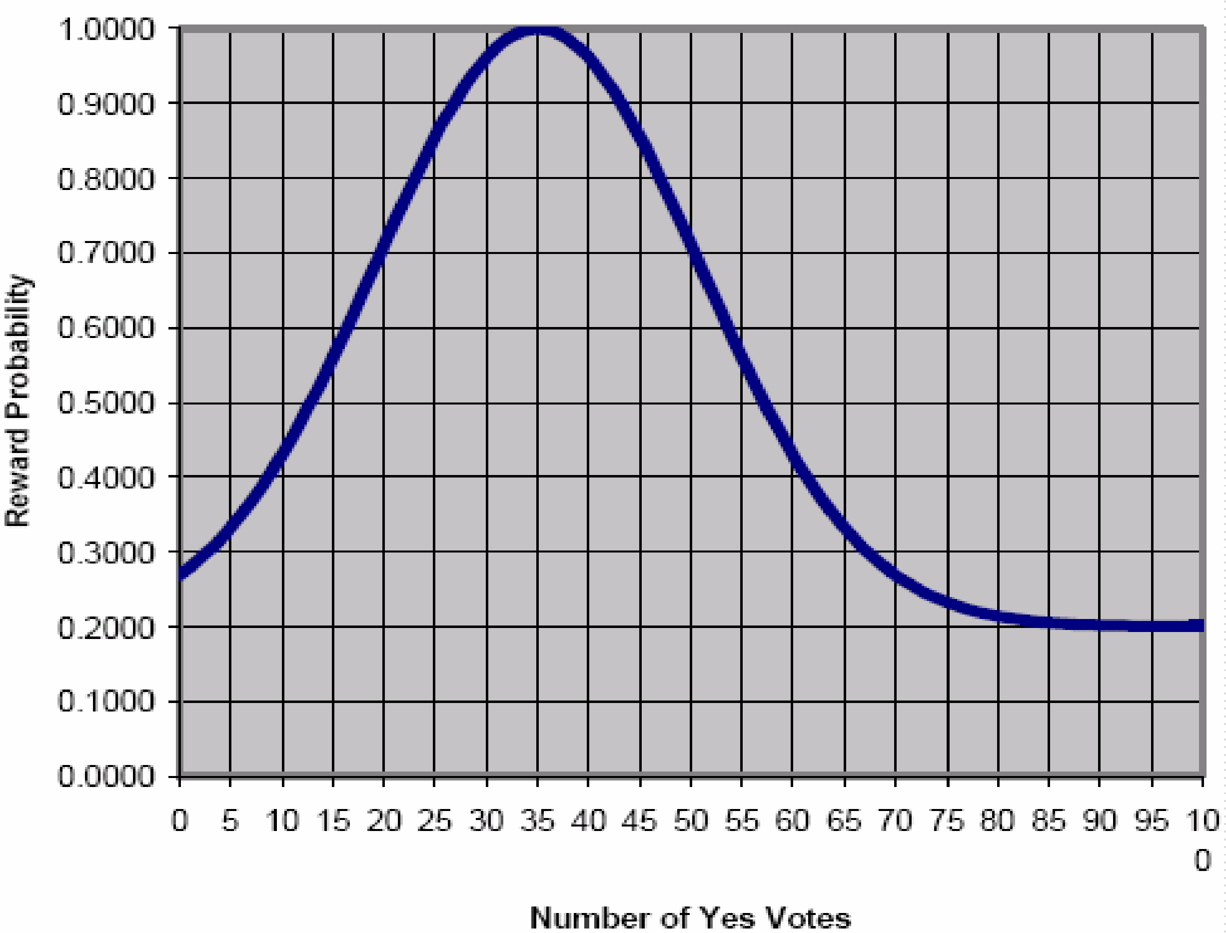
Gur Game

- There are n players (sensors), none of whom are aware of others
- There's a referee (BS) who periodically polls for simplified information from each player
  - The referee asks each player to vote "yes" (power-up) or "no" (power-off) and he counts up each response (# of packets received from active sensors = # of active sensors)

Gur Game (cont'd)

- A reward probability r = r(k) is generated, k is the # of players who voted "yes"
- A player is then rewarded (probability r) or penalized (probability 1-r) independently of their vote
- Gur property: K* of players vote yes, K* is the max value of function of r(k)

Typical Reward Function



Gur game with memory

- How to achieve Gur property: by trial and error
- The player votes *yes* when he is in a positive numbered state, and *no* when he is in a negative numbered state
- “center seeking” behavior is for punishment, and “edge seeking” behavior is for reward.

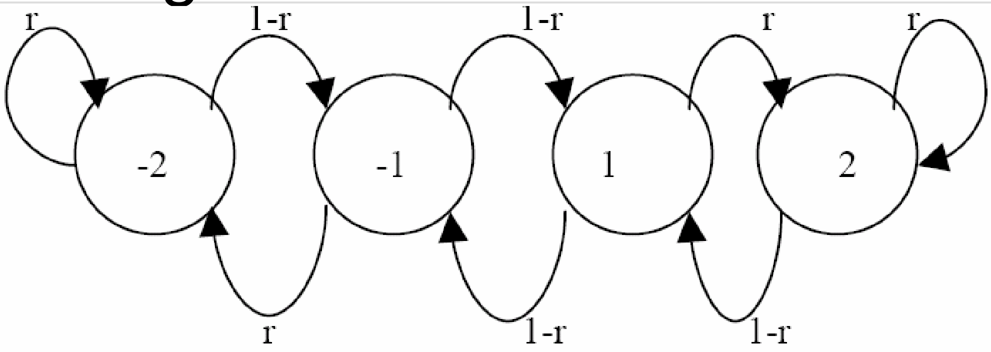


Figure 2. Gur Memory of Size N=2.

Assumptions

- One BS + n sensors
- BS polls sensors once a second
- Each sensor Si is a distance di from BS
  - mean that a packet is sent reliably from Si to BS and takes di seconds to reach BS
- BS have a broadcast channel to all the sensors
- BS receives k packets at time t means that approximately k sensors are powered-on at time t

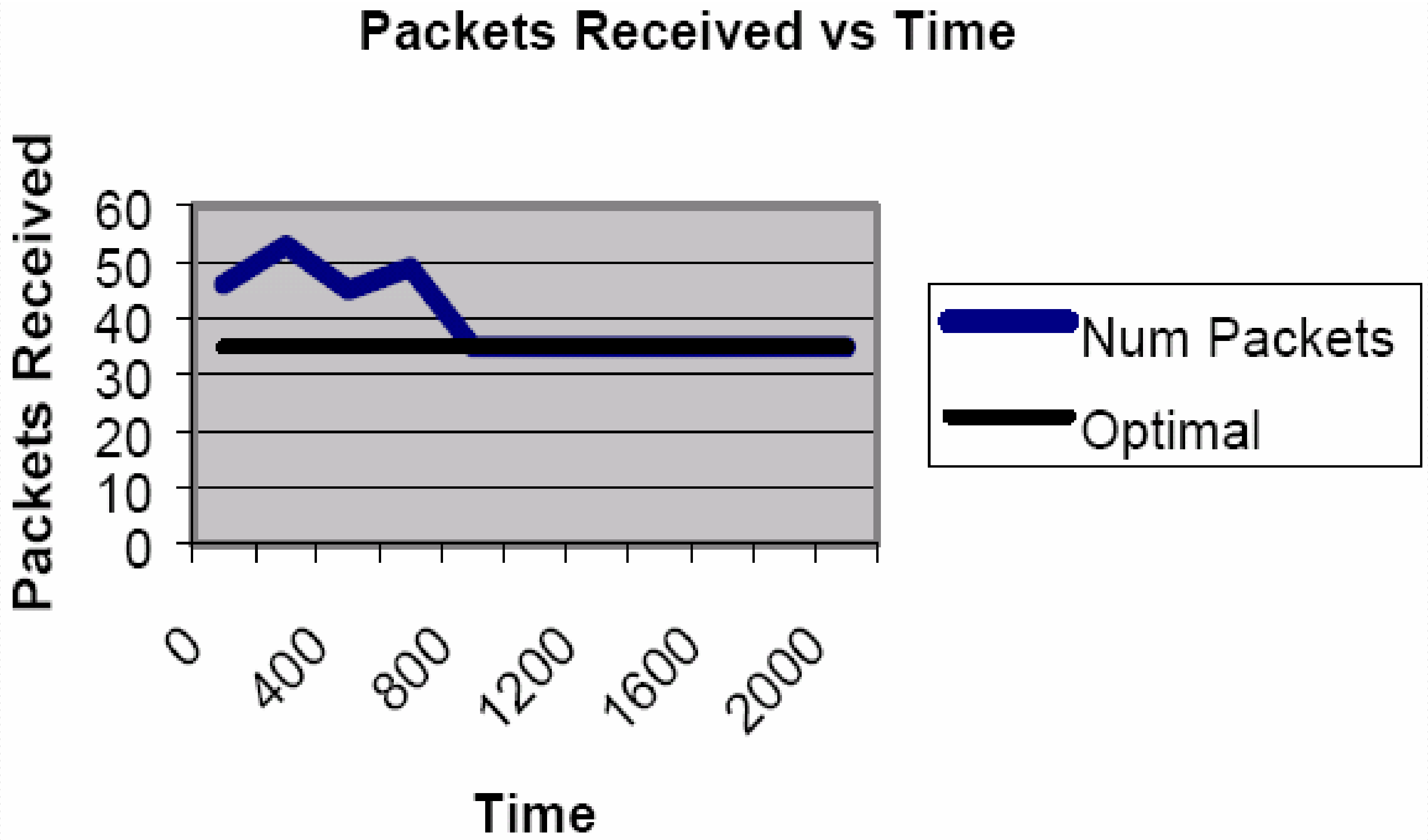
Results

- Will reach an equilibrium, K* sensors keep active until they use up all power

Simulation (simple case)

- Assume memory size N = 1
- 100 sensors as well as a BS
- No sensor failures or renewals
- Each sensor picks a random state as its initial state
- Assume # of optimum sensors = 35
- di = 1 sec (packet delay)

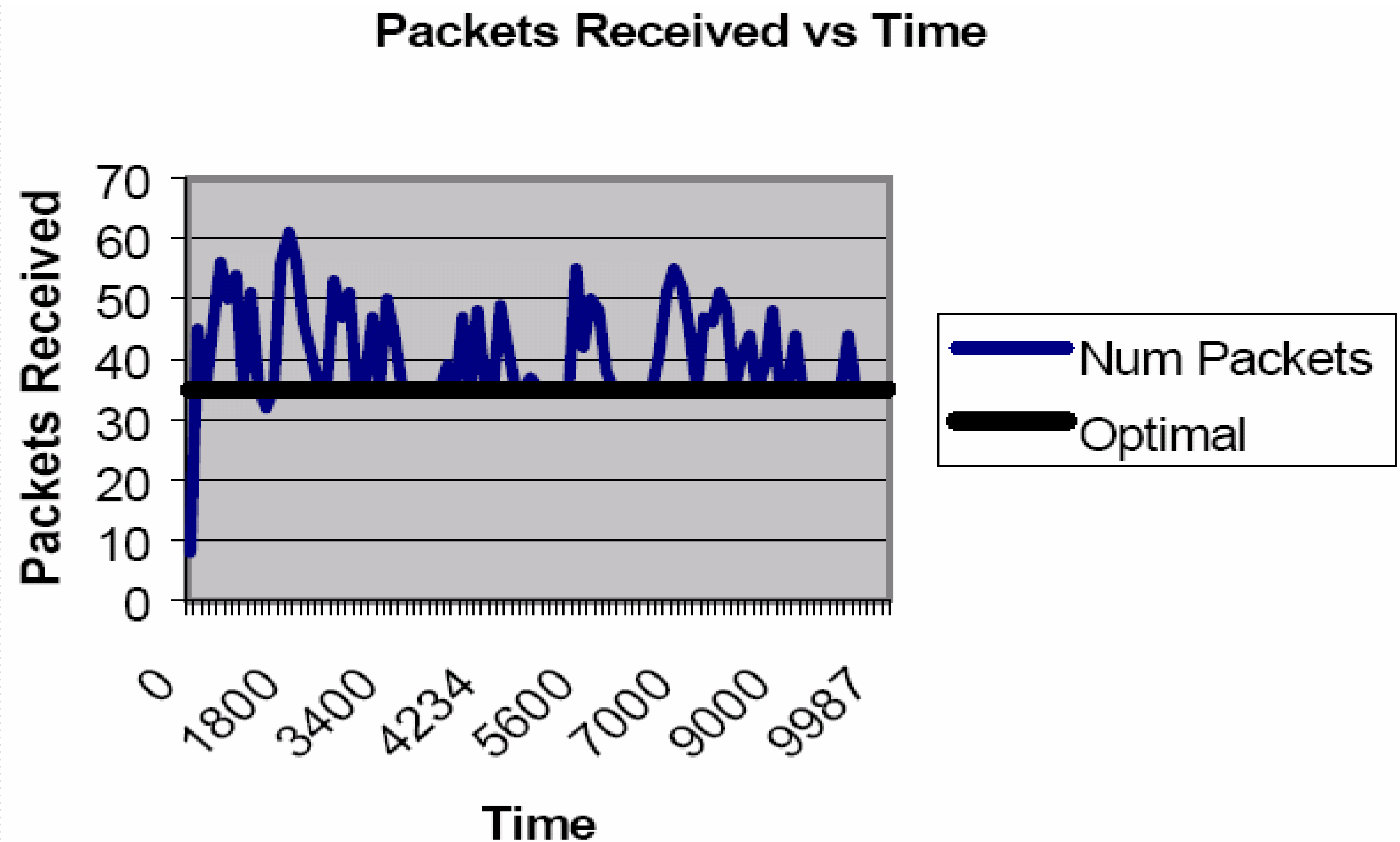
Simulation (simple case)



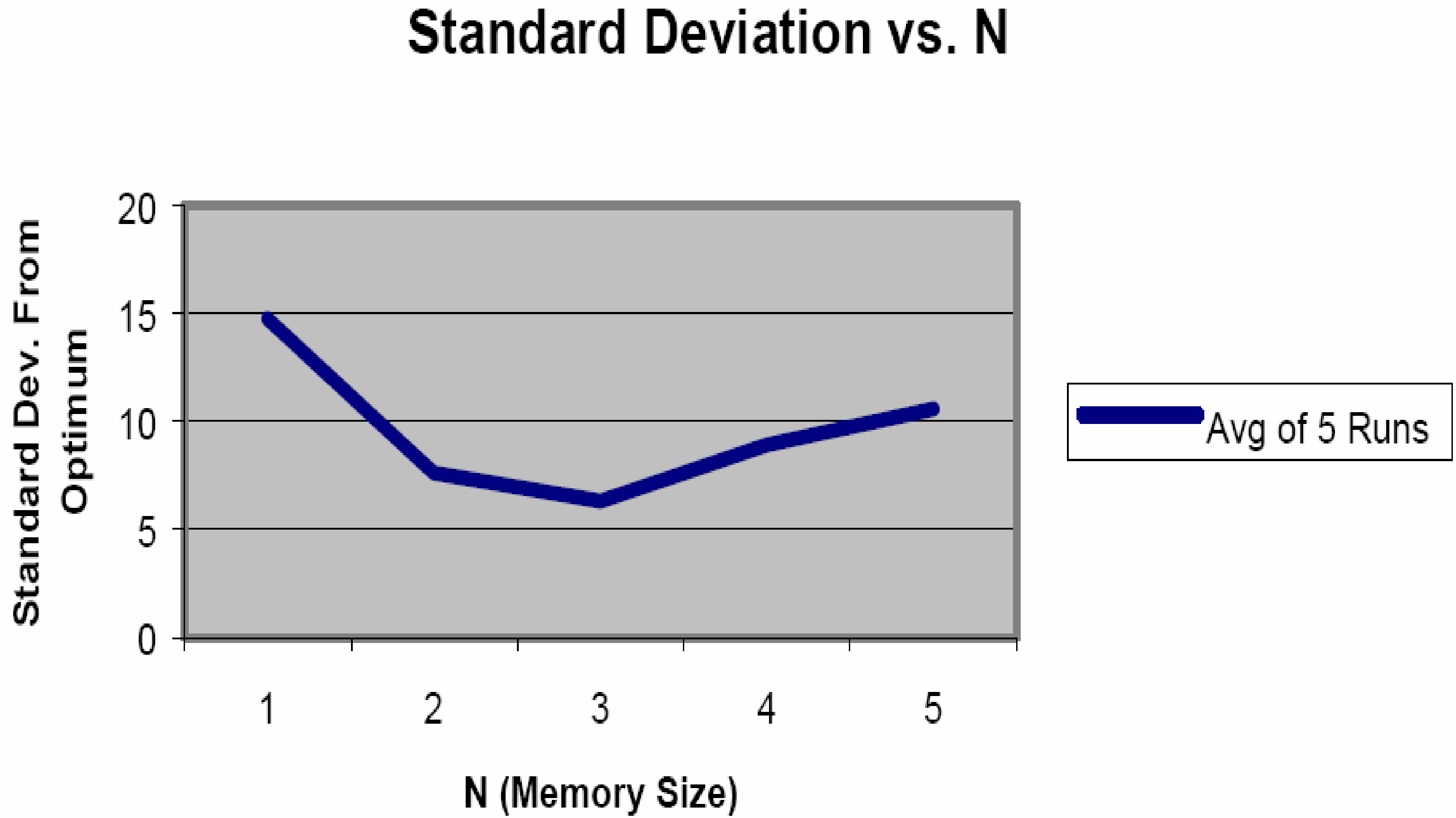
Simulation (realistic case)

- Allow the birth and death of sensors
- di is distributed uniformly from 0-5 secs
- Other parameters are the same as simple case

Simulation (realistic case)



Simulation (memory size)



Discussion

- Apply math on sensor networks? (e.g. OR, Game Theory)