Modeling Energy Constrained Routing in Selfish Ad Hoc Networks

Petteri Nurmi

Helsinki Institute for Information Technology HIIT

petteri.nurmi@cs.helsinki.fi

http://www.cs.helsinki.fi/Petteri.Nurmi

Introduction

- Nodes in ad hoc networks are energy constrained and potentially selfish è defining rational and optimal behavior is complicated
- the effects of selfishness have been widely studied
- but the effects of energy not
  - intuitively, when nodes condition their behavior on their energy, this leads to a more balanced resource usage
  - thus, energy optimization may well be the rational strategy
  - accordingly, in order to act optimally, a node should try to model the energy of other nodes
- Questions:
  - how to capture this formally?
  - how to build mechanisms that allow nodes to act optimally?

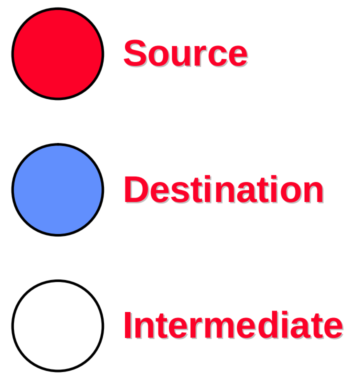
Contribution

- We present a game theoretic model that
  - models the energy of the nodes and links it with the behavior of nodes
  - models the effect of selfishness
- We also present a mechanism that
  - allows nodes to learn optimal strategies over time
  - is guaranteed to converge to a sequential equilibrium point
- And present simulation results for the model

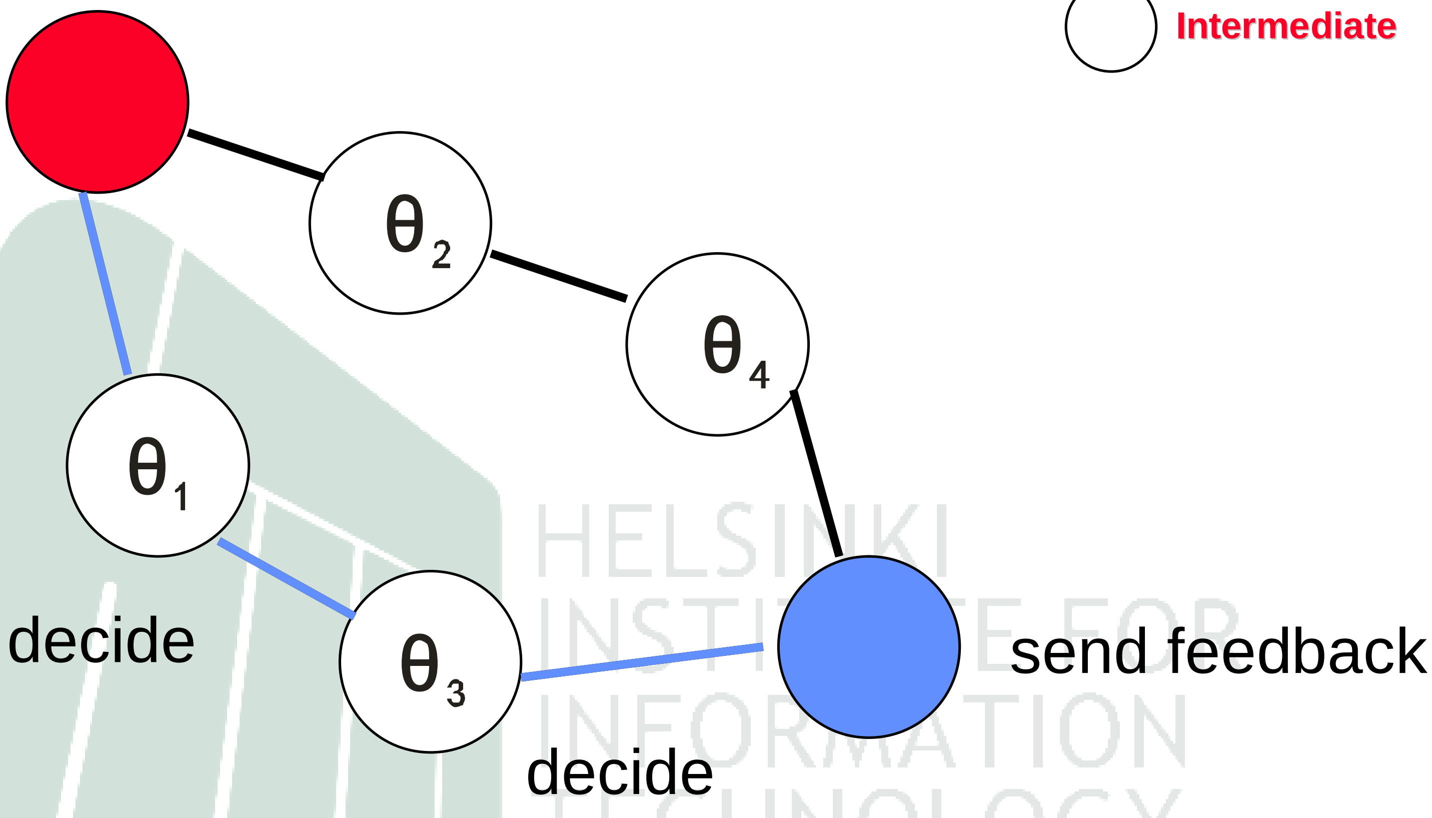
General Framework

- We consider repeated Bayesian games in an adaptive learning framework
  - agents optimize immediate (single shot) gain
    - thus differs from standard repeated games
  - agents hold theories about parameters of interest
    - in our case selfishness and energy
  - the sequence of theories forms a *learning process* that is *adaptive* if
    - the action selection probabilities of the agents eventually come close to empirical frequencies
    - and the beliefs of the agents converge.
  - closely related to stochastic approximation theory

Model Intuitively



optimize route, decide how many of the packets to send and send them



The Game Structure

- Source node
  1. Discover routes to the destination
  2. Estimate the “goodness” of the routes
  3. Select the route to use
  4. Gather feedback and update theories

- Intermediate node
  – Optimizes an utility function and selects the optimal action as given by the utility function
  – The utility function can utilize estimates about the behavior of the source

Estimating the Goodness of Routes

- Once the source has information about available routes, it calculates a score for each route
- The score combines
  - the energy and selfishness estimates
  - the importance of energy
  - the cost and the value of the packets
- We use the expected utility for a path
  - first we estimate the probability that a given number of packets arrives at the correct destination $\mu(P)=\prod[\alpha\varphi+\beta\psi]$
  - next, we combine this with the utility of the packets

$$u(s,P)=\mu(P)\,[h(s)-g(s)]$$

Intermediate Nodes

- The behavior strategies of the intermediate nodes are assumed to be
  - weighted linear combinations of the selfishness of the source and the remaining energy of the intermediate node
    - the weights represent importance in the individual components
  - thus we have a punishment mechanism
  - and we also have a part that depends on the current energy of the nodes

$$\sigma = \alpha\varphi + \beta\psi$$

Obtaining Feedback

- In order to update the theories of the nodes, they require feedback
  - i.e., how many packets finally arrived at the destination
  - in order to accomplish this, we need
    - packet timers
    - acknowledgements
  - if the packets are not acknowledged before the timer goes off, they are considered to be dropped

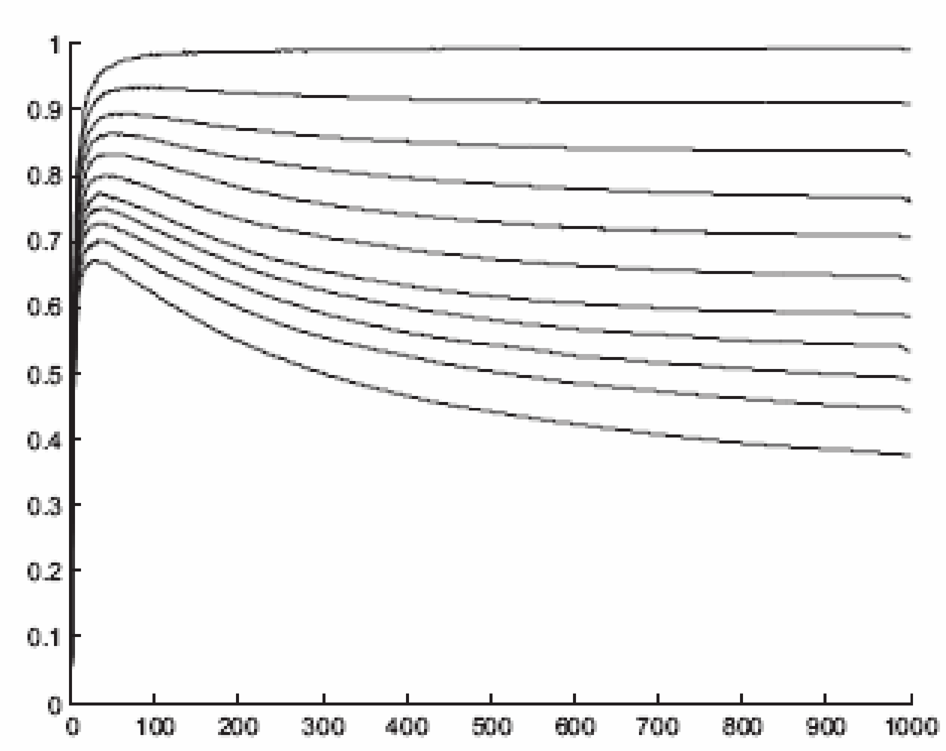
Learning Optimal Strategies

- It turns out that a learning process can be shown to converge to an equilibrium when
  - the beliefs are updated using the Bayes' rule
  - strategies are selected using reinforcement learning rules and random experimentation occurs often enough
- In our case, we use the rule 1/k, where k is the current iterate, to select suboptimal actions
  - Also other rules possible, large literature on this in the field of reinforcement learning
- For the convergence proof, see the paper.

Experimental Setup

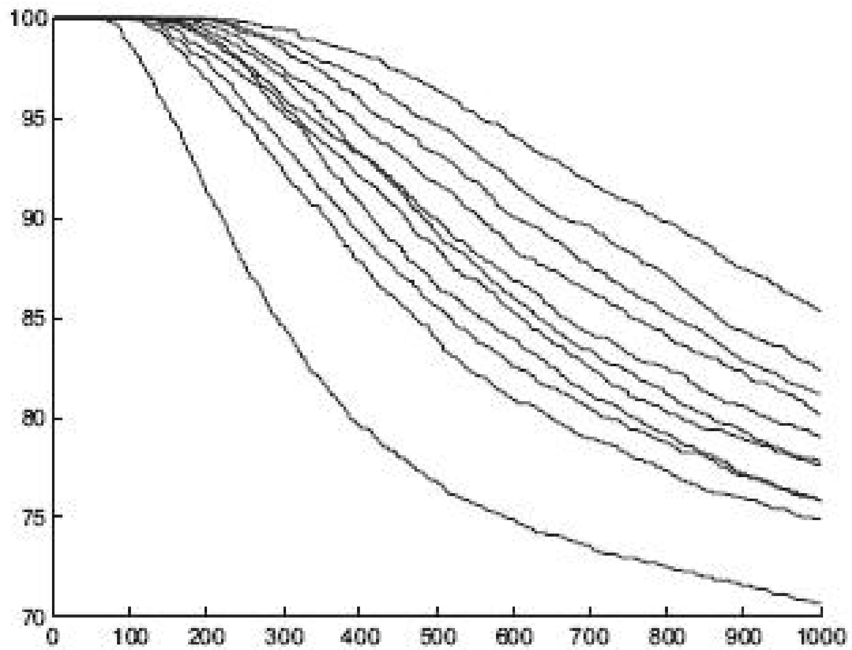
- We tested the model using randomly generated network topologies
  - constrained uniform deployment: nodes were not allowed to be "too close" to each other
  - random waypoint model with constant speed
  - Poisson distributed traffic
- In each setup, the values of the importance parameters were the same for all nodes
  - tested with different initial values of the importance parameters
  - each initial configuration repeated 30 times
- In the experiments we ignore selfishness for clarity

Results 1: Success rate of nodes



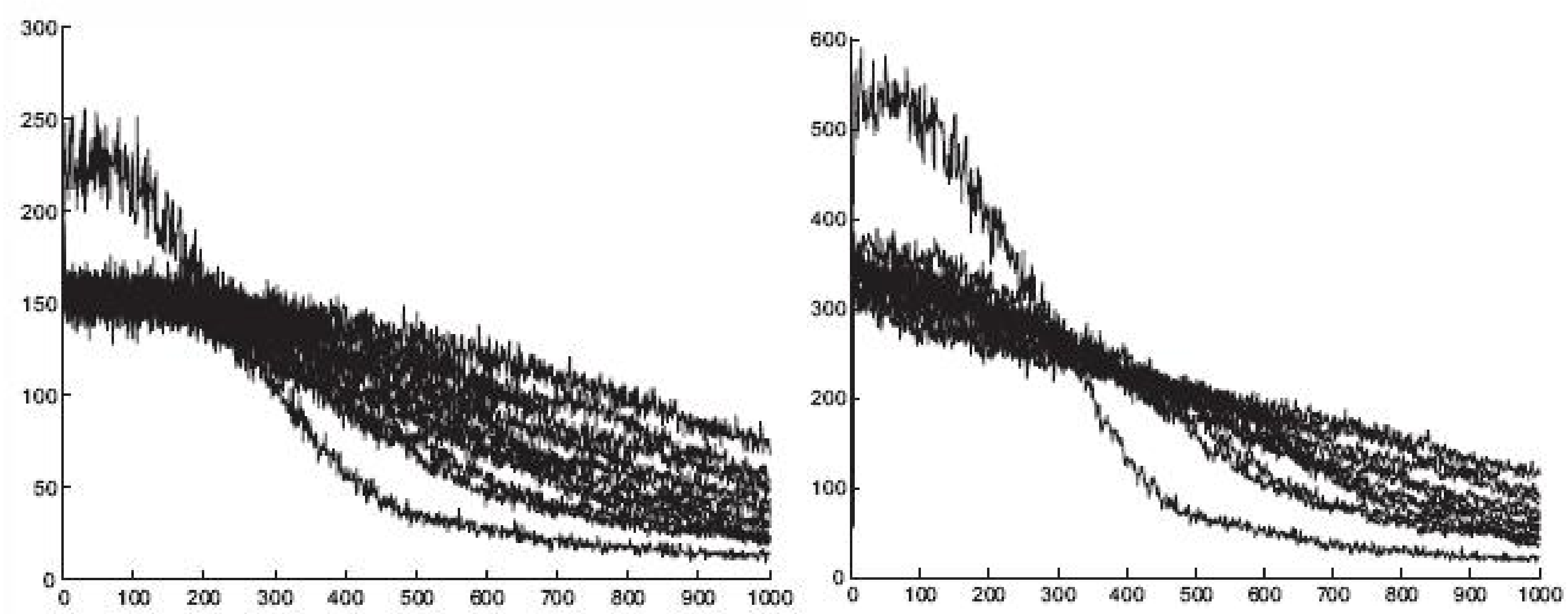
- **Success rate of nodes**
  - Importance of energy class varied (in steps of 0.1) between 0.0 and 1.0
  - Topmost, importance of energy = 0.0
  - Subsequent, increase by 0.1

Results 2: Amount of nodes alive



- **Amount of nodes that are alive**
  - Importance of energy class varied (in steps of 0.1) between 0.0 and 1.0
  - Topmost, importance of energy = 1.0
  - Subsequent, decrease by 0.1

Results 3: Packets sent and forwarded



- **Amount of packets sent (left) and forwarded (right)**
  - Importance of energy class varied (in steps of 0.1) between 0.0 and 1.0
  - Steepest descent, importance of energy = 0.0
  - Subsequent, increase by 0.1

Conclusions

- Our simulations indicate that energy optimization in decision making affects network lifetime è this should be the rational choice of the nodes
- The results also indicate that nodes can learn a reasonable strategy when energy influences decisions
  - Though the results show that this easily results in a decrease of overall performance

è building mechanisms that are able to model the energy of others can prove useful in highly resource constrained environments

Summary

- We have presented a game theoretic model that captures selfishness and resource constraints of devices
- We presented a mechanism for learning optimal strategies over time
- And we presented simulation results for the model
- Our model shows that:
  - Energy optimization is likely to be the rational choice of nodes
  - But other devices can learn to optimize their behavior also in this kind of settings