An Experimental Study of Basic Communication Protocols in Ad-hoc Mobile Networks

Ioannis Chatzigiannakis, Sotiris Nikoletseas,

Nearchos Paspallis, Paul Spirakis, Christos Zaroliagis

Computer Technology Institute
and
Dept of Computer Eng & Informatics, University of Patras

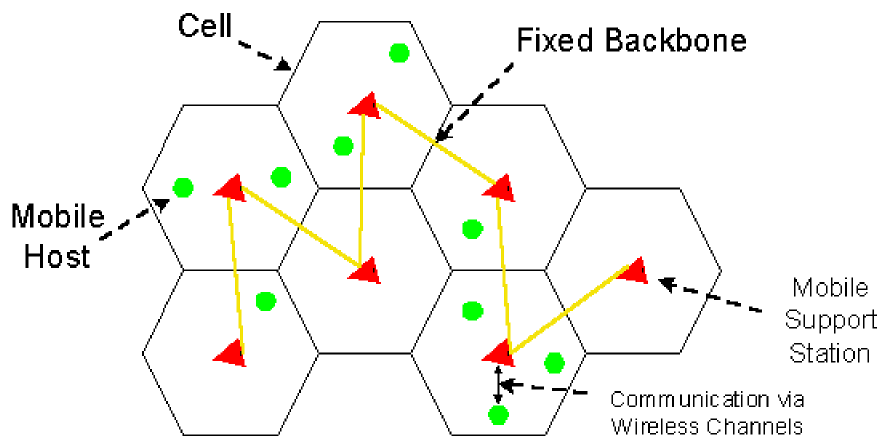
Mobile Computing

- A new computing environment
  - New wireless products and research
  - Wireless media for transmitting data
  - Slower but more convenient than wired media
- Constraints in Mobile Computing
  - Poor resources
    - Limited communication bandwidth
    - Highly dynamic connectivity
    - Volatile energy sources

Models

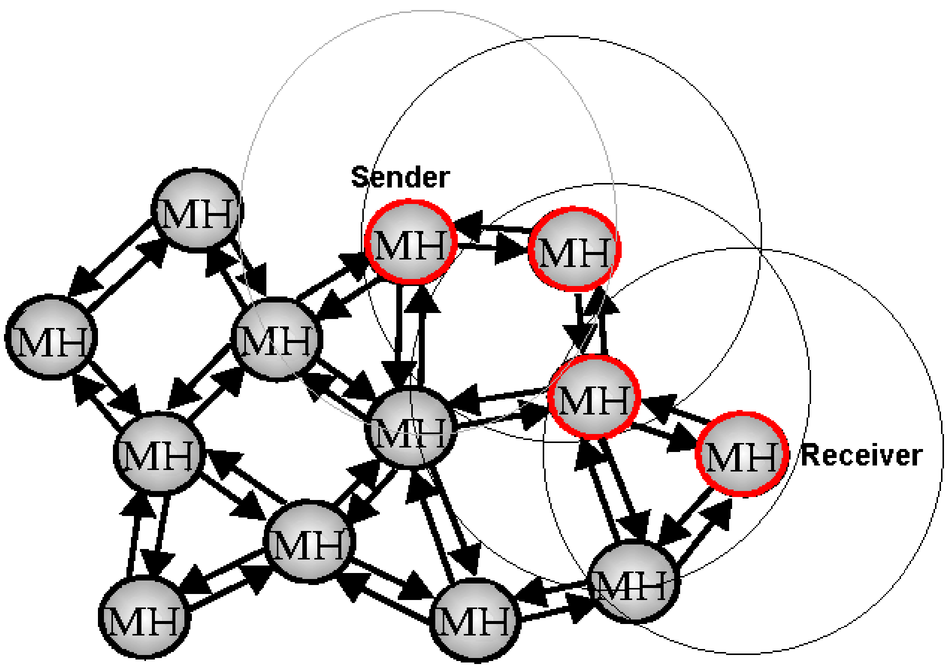
- Fixed-backbone model
- Ad-hoc model

Fixed-backbone Networks



- *Mobile support stations* are hard-wired connected
- Each mobile support station is assigned a cell
- Communication is always routed through the fixed backbone part of the network
- Modern cellular networks $\subset$ Fixed-backbone Networks

Ad-hoc Networks



- No established infrastructure
- No centralized administration
- Every user is mobile with wireless communication capabilities
- Not-in-range users use multi-hop communication
- Users must be willing to forward packets

Motivation for Ad-hoc Mobile Networks

- Easy and rapid deployment in unknown terrain
- No existing infrastructure is available
- Robustness
- Instant networking
- Ad-hoc networks are ideal for rescue missions (or military operations)

Problem

- Basic communication problem in ad-hoc networks
  - Send information from some *sender S* to some *receiver R*
- Difficulties
  - Highly dynamic system
  - Connections are constantly forming and breaking
  - No centralized control

Compulsory vs Non-compulsory protocols

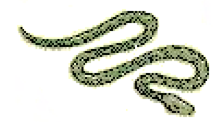
- *Non-compulsory*: movement of hosts is independent of the protocol
- *Compulsory:* all hosts move as per the needs of the protocol

A compromise

- *Semi-compulsory:* a very small part of hosts (the *support* Σ) move as per the needs of the protocol

Our contribution

- A new simulation environment
- Experimental study of two semi-compulsory protocols
  - *Snake* [Chatzigiannakis, Nikoletseas, Spirakis, 2000]
    - Re-implementation
  - *Runners* (a new semi-compulsory protocol!)
- New protocol improves performance
- Both protocols require a very small $|\Sigma|$

The *Snake* Protocol 

- Set-up phase: form snake structure
- *Snake*
  - *Head:* perform random walks
  - *Others:* move where their “predecessor” was before
- Messages forwarded from Σ to *Receivers*
- Synchronization: messages are copied to each member of Σ

The *Runners* protocol 

- No set-up phase: independent runners

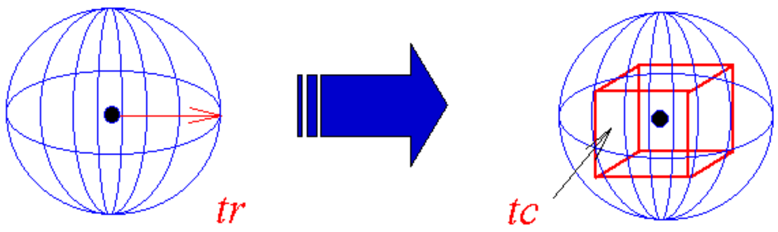
- *Runners:* each runner performs an independent random walk

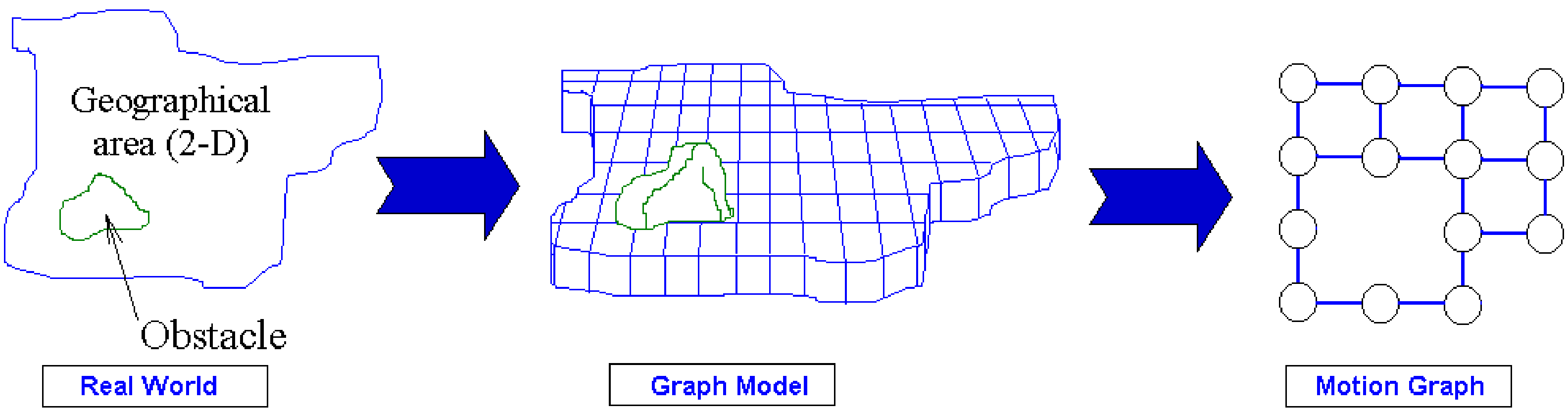
- Messages forwarded from Σ to *Receivers*

- Synchronization: requires 2 rounds

The Model of the Space of Motion [Hatzis et al, 1999]

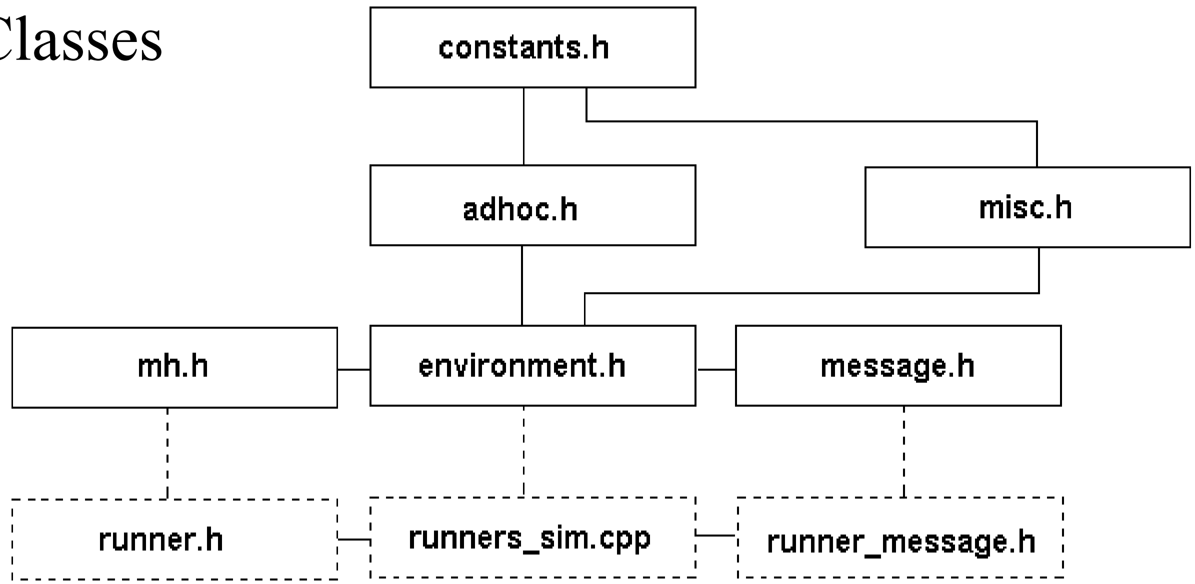
- Graph-theoretic model
- The environment is abstracted by a graph





The simulation environment

- C++ Classes



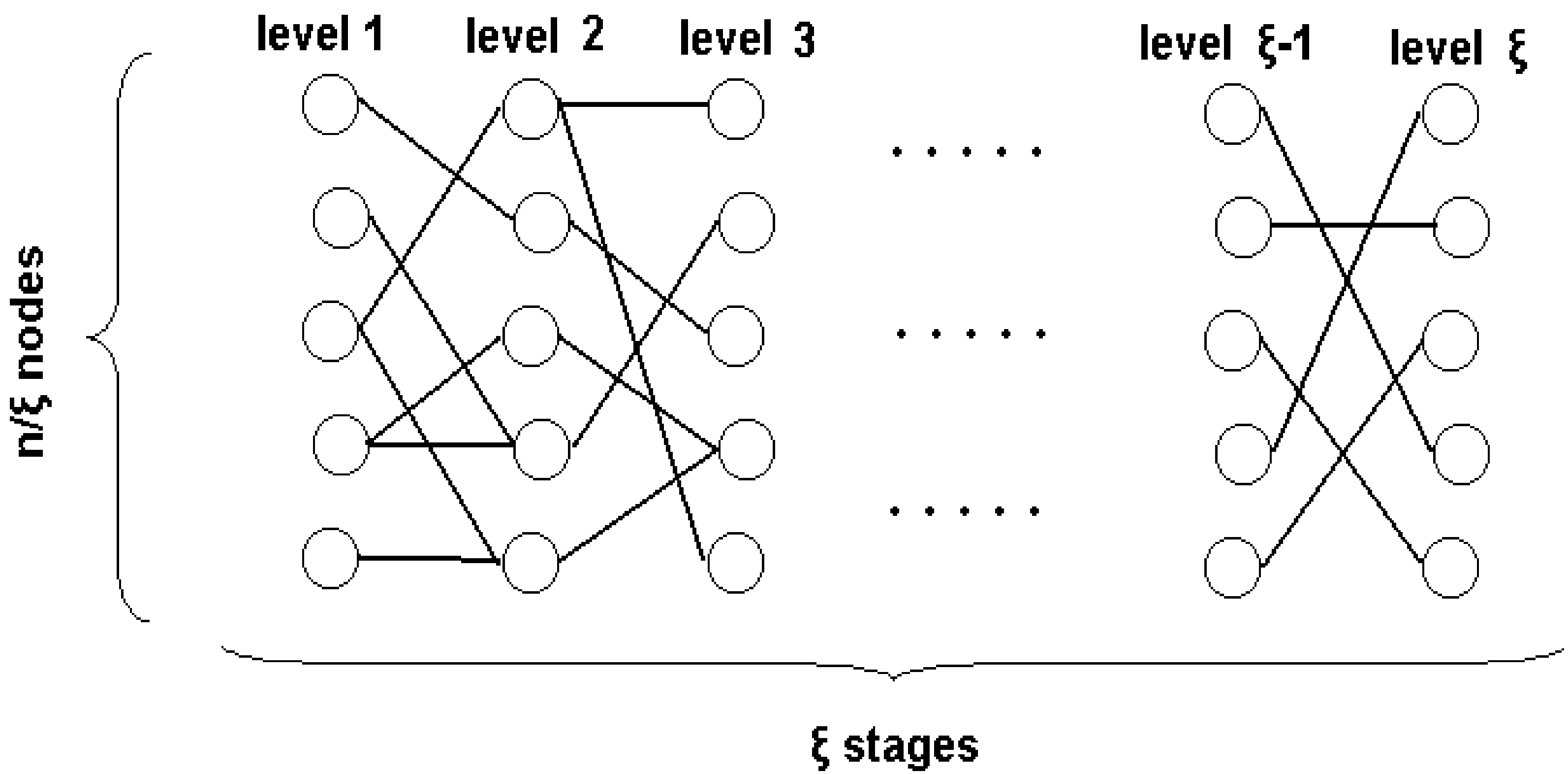
- LEDA data types
  - sets
  - lists
  - graphs

Experimental setup

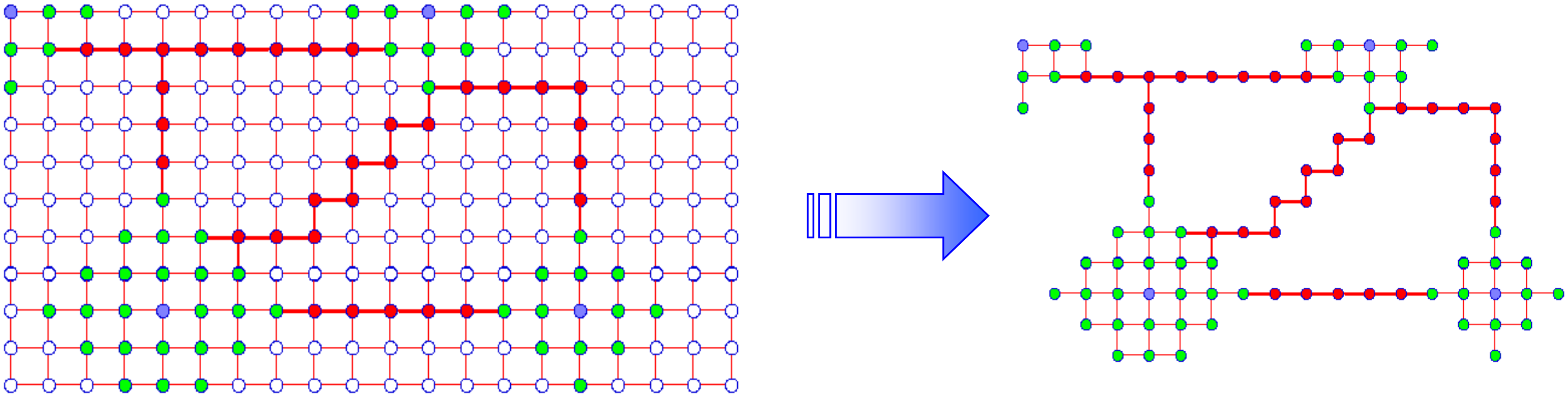
- Five (5) types of motion graphs
  - random graphs
  - 2-D grids
  - 3-D grids
  - bipartite multistage graphs
  - two-level motion graphs

- Number of nodes: $n \in [\,400, 6400]$
- Support size: $k \in [3, 45]$
- 1,000 users × 100 messages = 100,000 total transactions

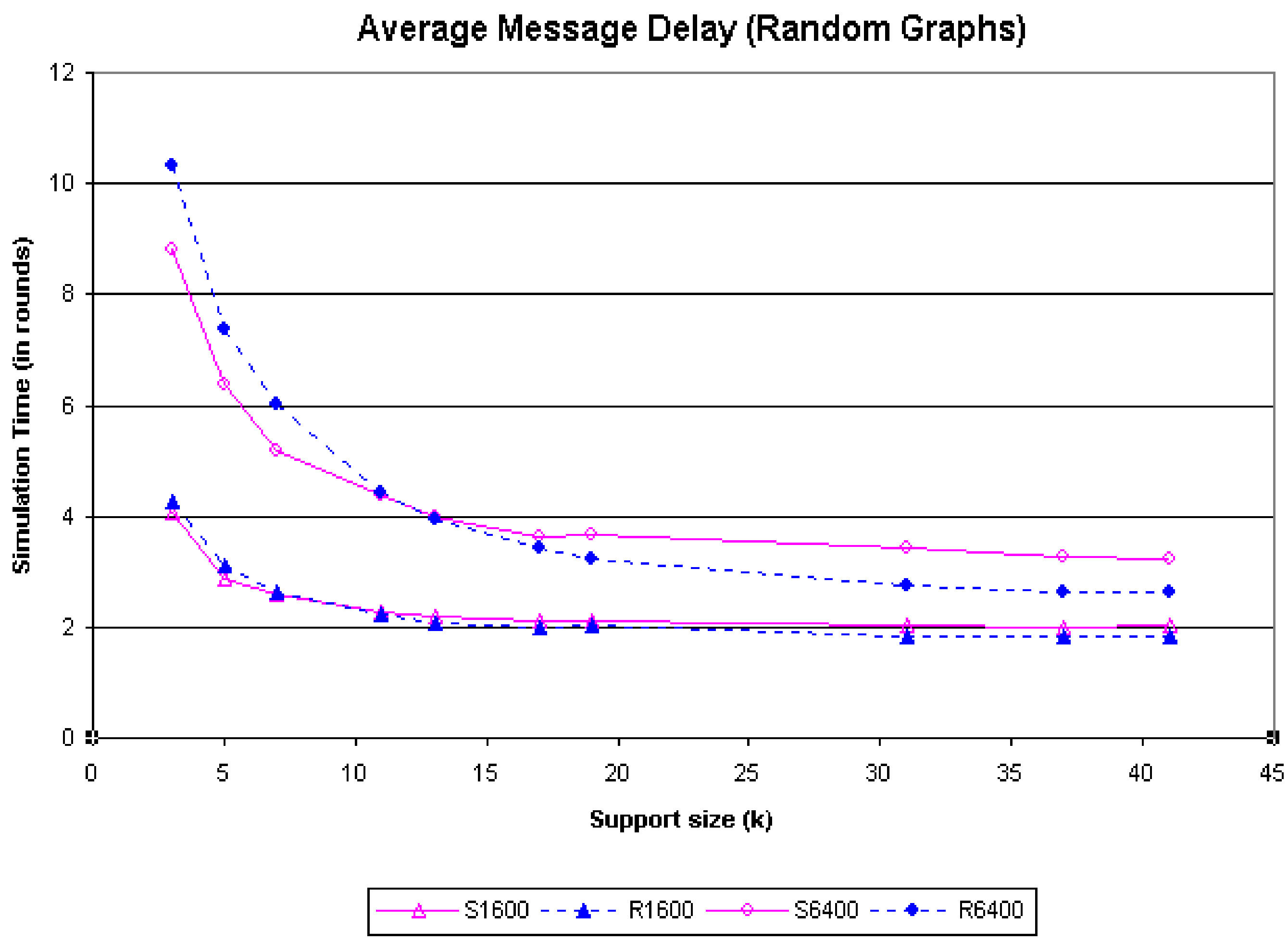
Bipartite multistage motion graphs



Two-level motion graphs

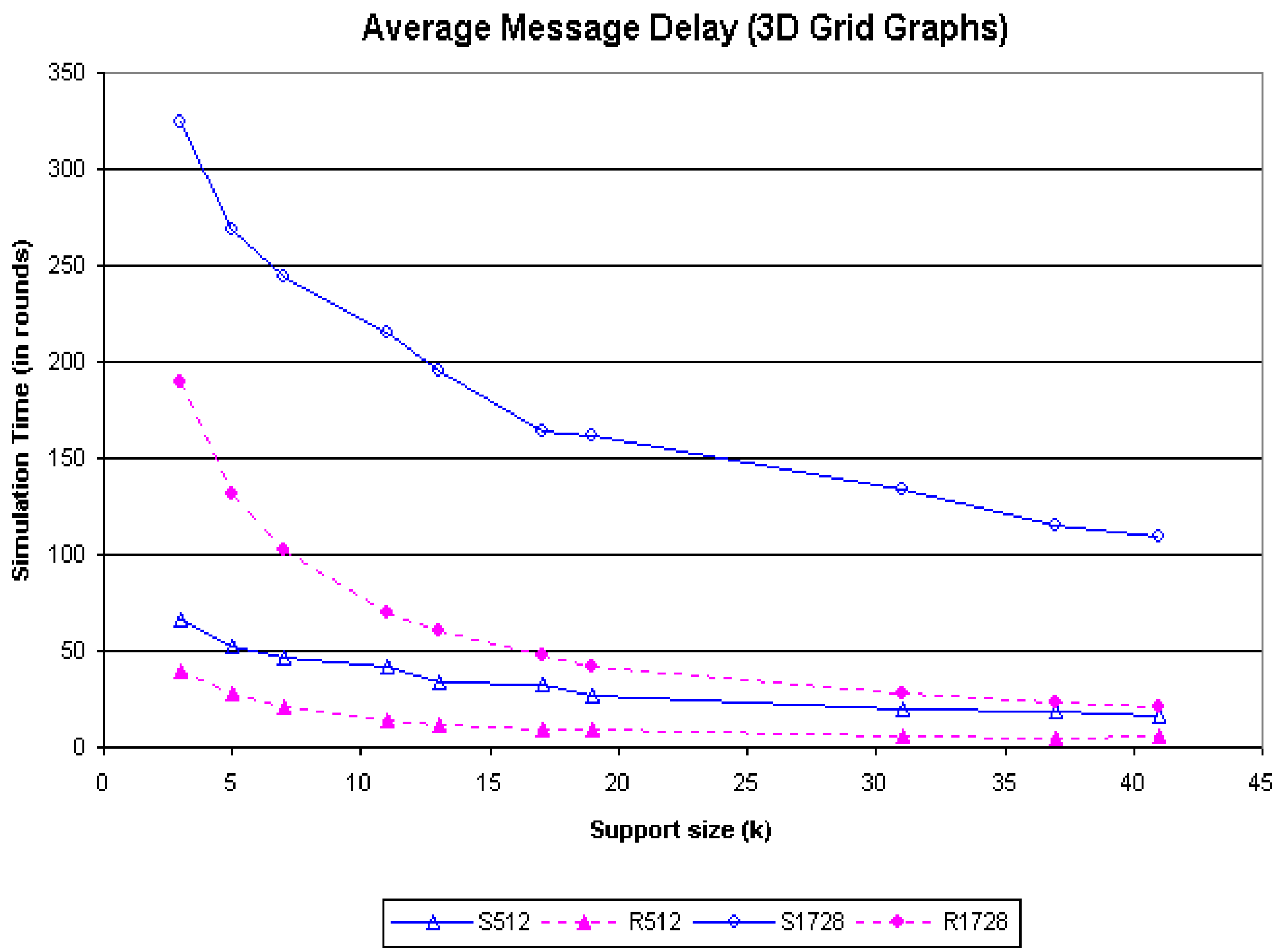


Results - 1/6



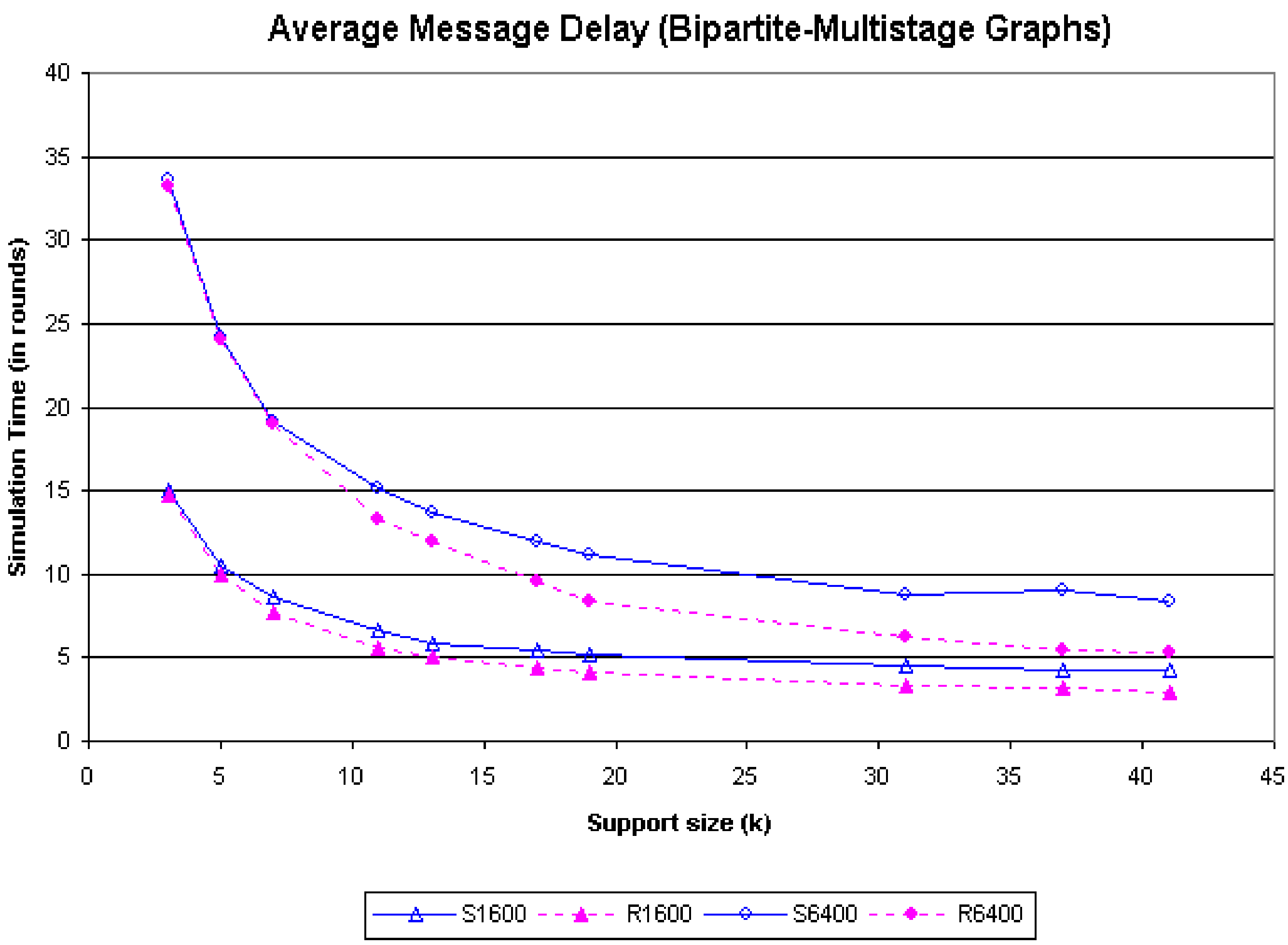
Average Message Delay (Random Graphs)
Simulation Time (in rounds)
0
2
4
6
8
10
12
0
5
10
15
20
25
30
35
40
45
Support size (k)
S1600
R1600
S6400
R6400

Results - 2/6

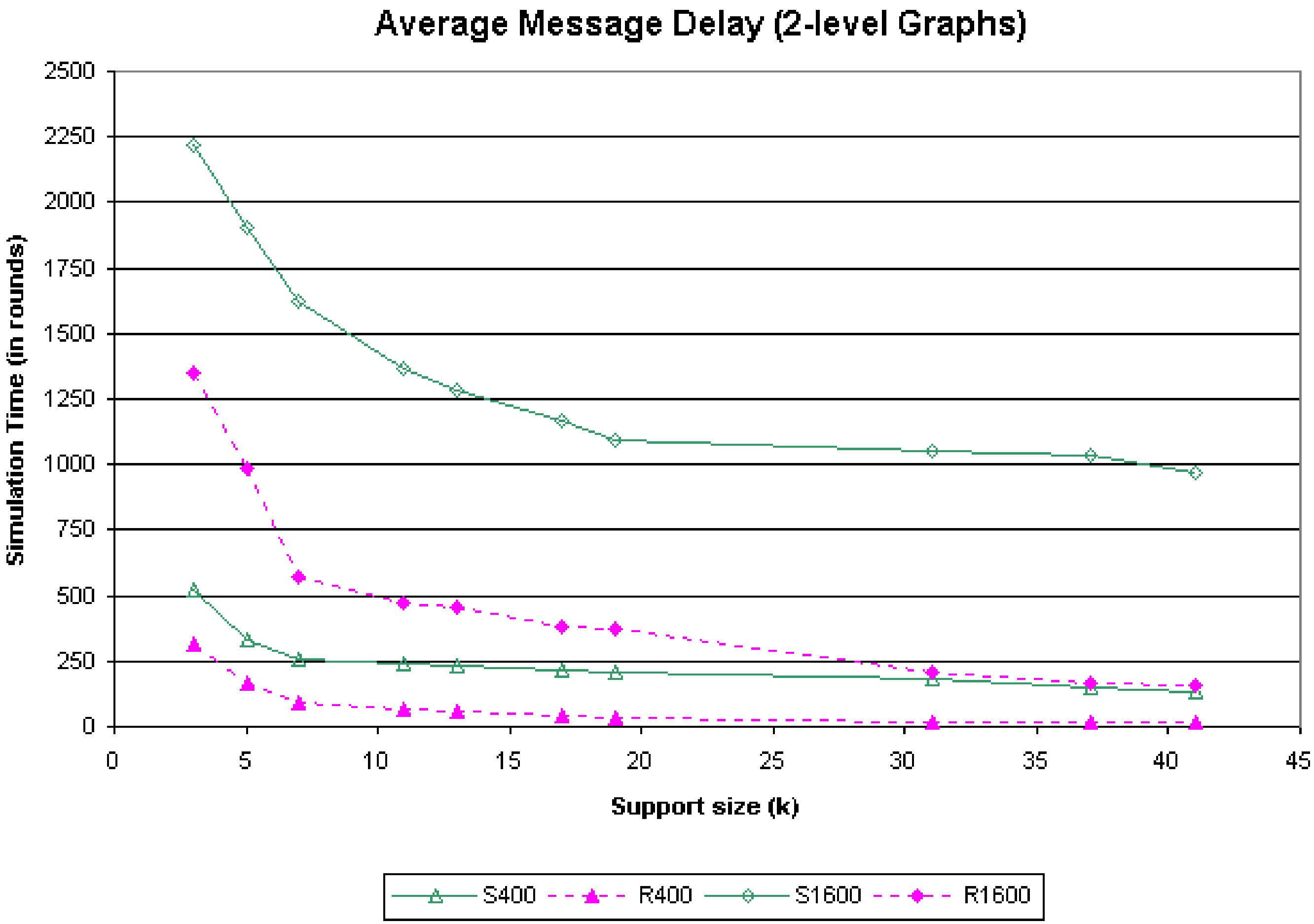


Average Message Delay (3D Grid Graphs)
Simulation Time (in rounds)
350
300
250
200
150
100
50
0
0
5
10
15
20
25
30
35
40
45
Support size (k)
S512
R512
S1728
R1728

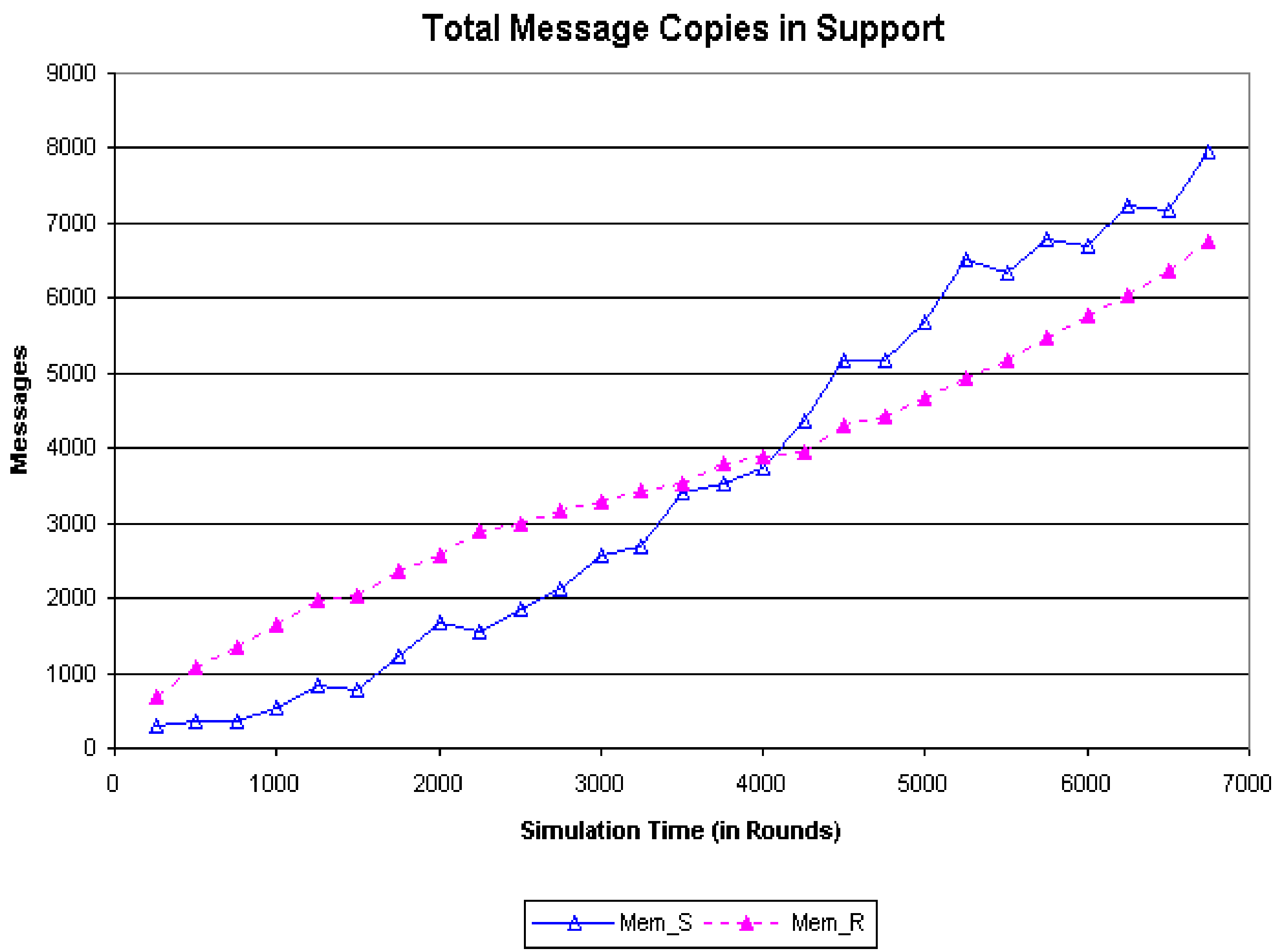
Results - 3/6


Average Message Delay (Bipartite-Multistage Graphs)
Simulation Time (in rounds)
40
35
30
25
20
15
10
5
0
0
5
10
15
20
25
30
35
40
45
Support size (k)
S1600
R1600
S6400
R6400

Results - 4/6

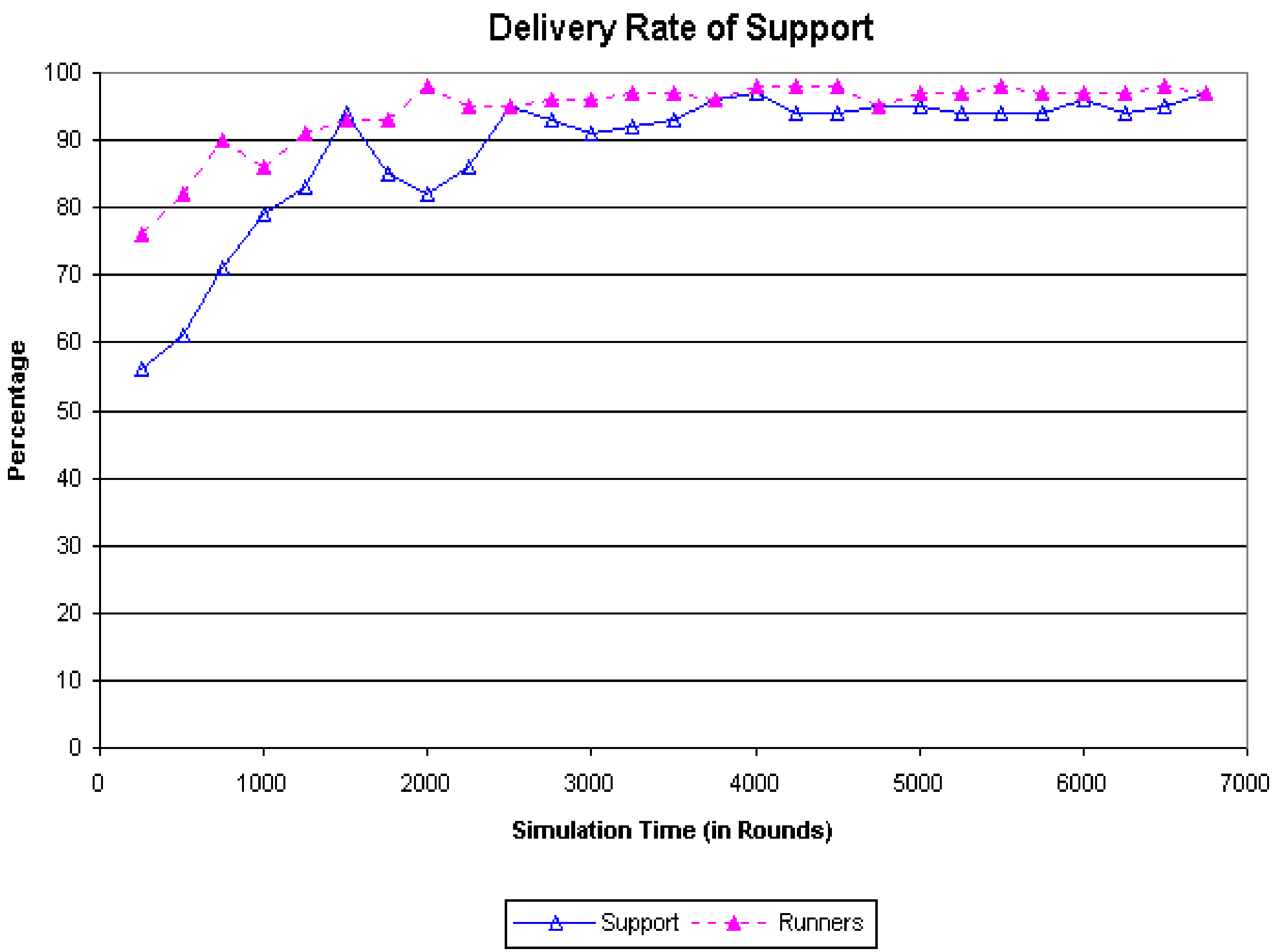

Average Message Delay (2-level Graphs)
Simulation Time (in rounds)
2500
2250
2000
1750
1500
1250
1000
750
500
250
0
0
5
10
15
20
25
30
35
40
45
Support size (k)
S400
R400
S1600
R1600

Results - 5/6



Total Message Copies in Support
Messages
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
0
1000
2000
3000
4000
5000
6000
7000
Simulation Time (in Rounds)
Mem_S
Mem_R

Results - 6/6



Delivery Rate of Support
Percentage
100
90
80
70
60
50
40
30
20
10
0
0
1000
2000
3000
4000
5000
6000
7000
Simulation Time (in Rounds)
Support
Runners

- Conclusions
  - Experimental study of *two basic communication protocols for ad-hoc mobile networks*
  - The *Runners* protocol outperforms the *Snake* protocol for almost every input considered

- Future Work
  - Theoretical analysis of the *Runners* protocol
  - Dynamic control of support population (births and deaths)