SmartGossip: an improved randomized broadcast protocol for sensor networks

Presented by

Vilas Veeraraghavan

Advisor – Dr. Steven Weber

Presented to the Center for Telecommunications and Information Networking

Abstract

- Four performance metrics for randomized broadcast protocols on sensor networks:
  - ✓ coverage, energy efficiency, per hop latency, and overhead.
- Focus - evaluate the extent to which the exchange of local information (either active or passive) can improve protocol performance.
- Three protocols were studied and their performance against the well known GOSSIP1 ([1]) protocol was compared.
- Explore the strengths and weaknesses of the above protocols
- Propose the new SmartGossip protocol.

Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

Outline of talk

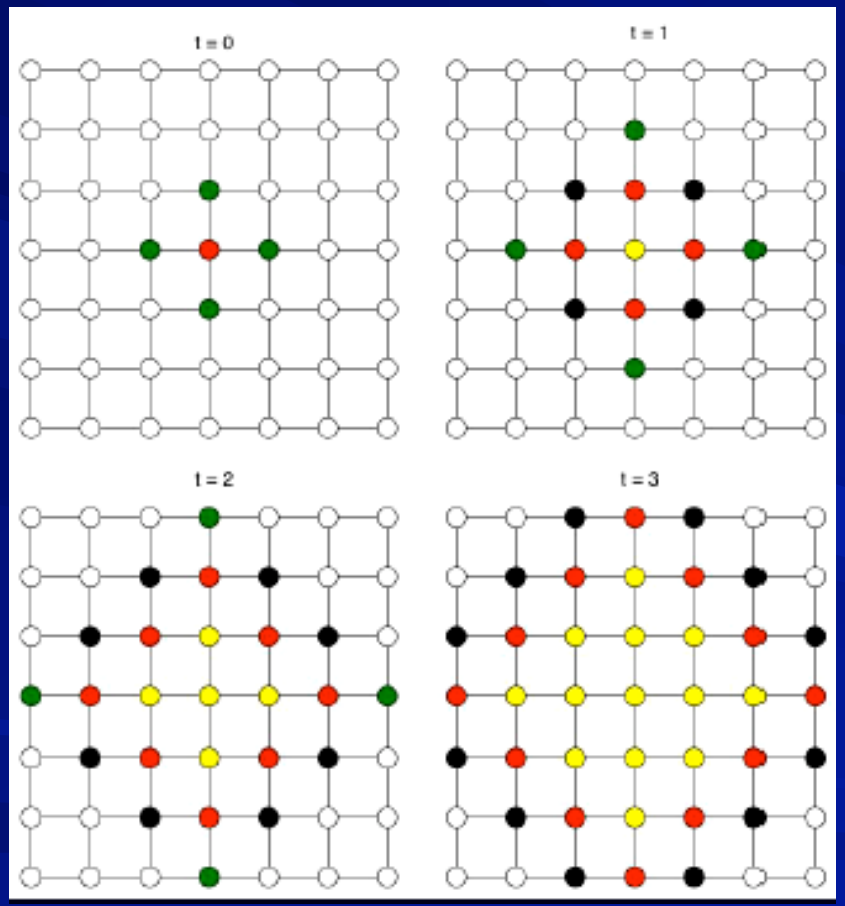
- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

Background

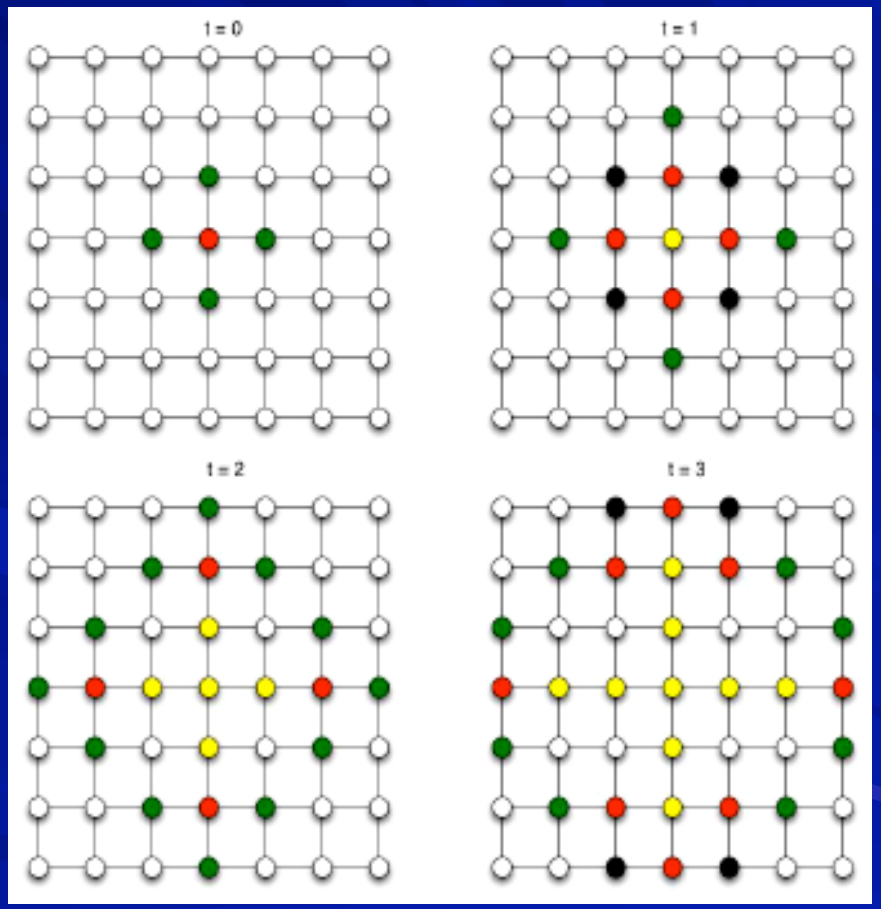
- Gossip protocols – Randomized broadcasting
- Simple and Distributed method to disseminate information in a network
- Assumption – Nodes on a grid
- Importance of state promulgation in wireless sensor networks
- Easiest way – Flooding
- Tradeoff between redundancy and robust coverage of network
- First pass to make things better – Controlled flooding

What is the difference?

Naïve Flooding



Controlled Flooding



Probabilistic Model

- Each node transmits the message on its first reception (only) with a probability *p*
- Low probability of transmission = low redundancy (efficiency)
- But high probability of transmission needed for higher coverage
- Synchronous versus asynchronous

Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

Related Work

- First mention of gossip by Hajnal, Miller, and Szemeredi [4] in 1972
- Gossip variants – SPIN[2], directional gossip[7], push-pull gossip[3], others
- Extensive Analytical work – spatial gossip analysis[8], percolation theory[9], copulas[5], epidemiology[6]
- Focus on gossip as a resource discovery mechanism e.g.[7]
- Protocols – from Z. J. Haas et al. for gossip, W. R. Heinzelman et al. for SPIN, and R. Karp et al. for push-pull

Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

Metrics

- Coverage – Fraction of nodes that receive the message
- Energy Efficiency – Lower redundancy
- Per Hop Latency – Rate of propagation of message across the network
- Control Overhead – Messages other than data messages

Coverage

- Network of n nodes
- N_r = random variable denotes number of nodes that have received the message
- If coverage is denoted as C then

$$C = E[N_r]/n$$

- Coverage is the primary goal especially in cases where the message is of critical importance

Energy Efficiency

- Goal – minimize redundant transmissions
- N_r = random variable denotes number of nodes that have received the message
- If Efficiency is denoted as η then

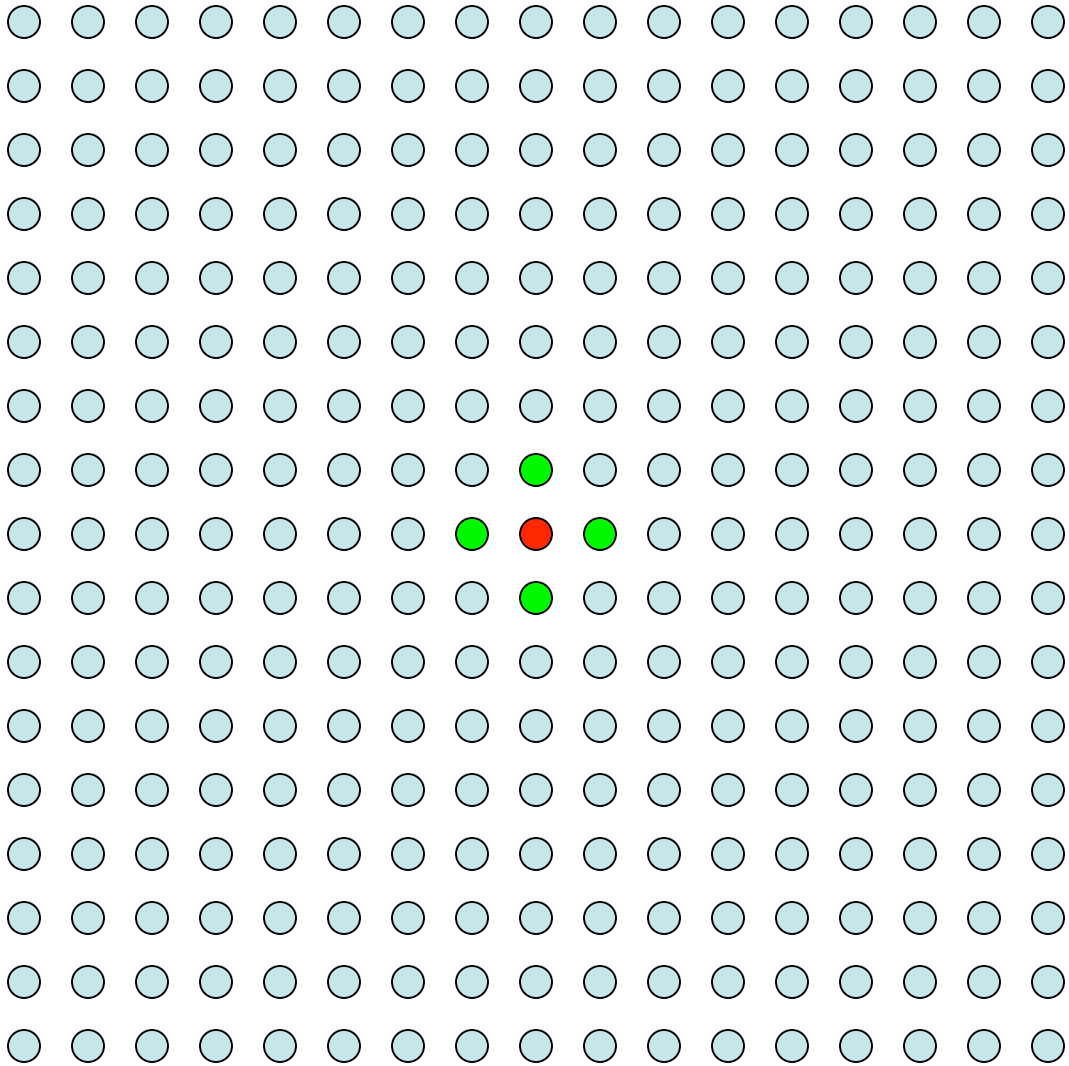
$$\eta = E[N_r]/E[N_t]$$

- Where N_t is the number of transmissions per time
- Motivation – Sensor has practical energy loss

Outline of talk

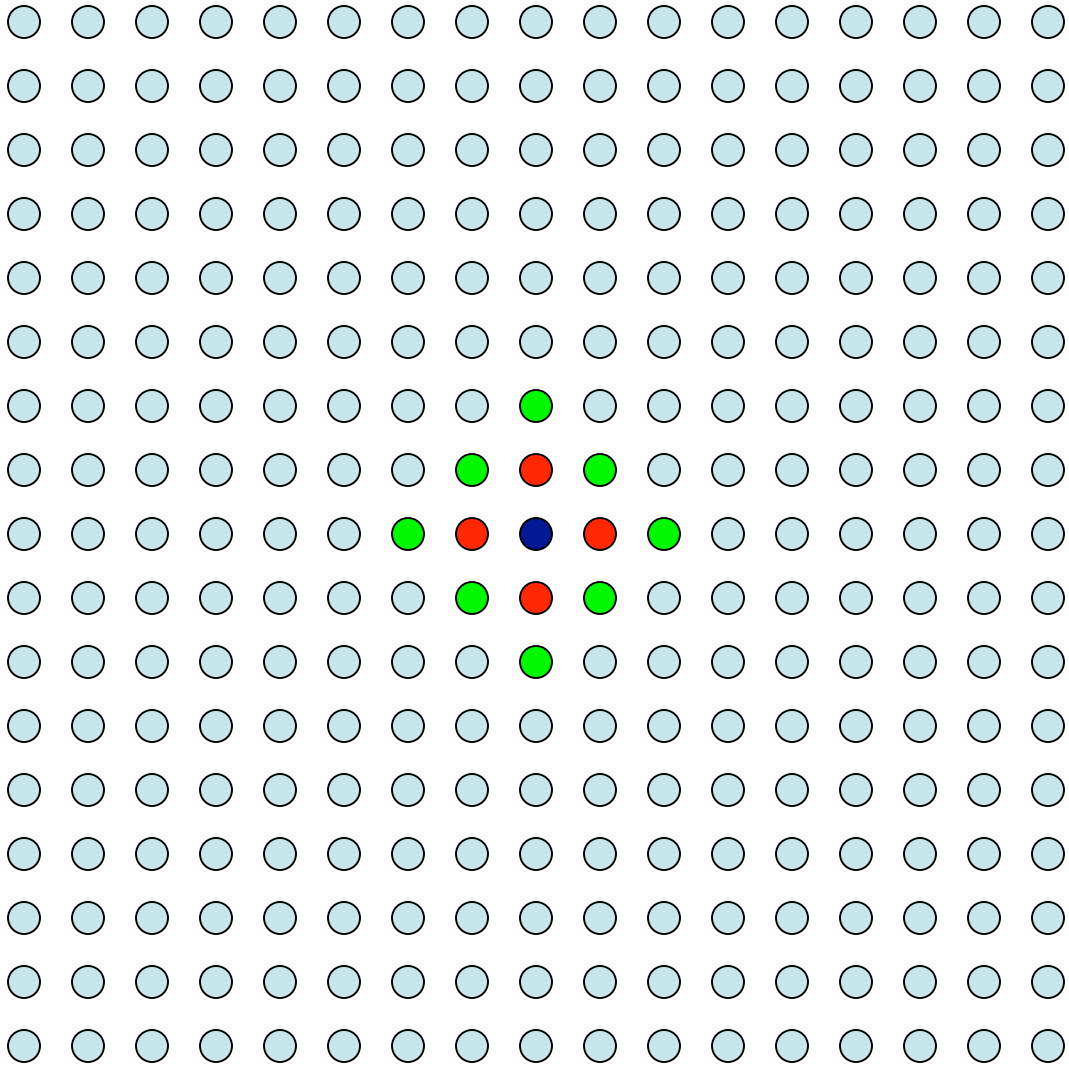
- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

GOSSIP1



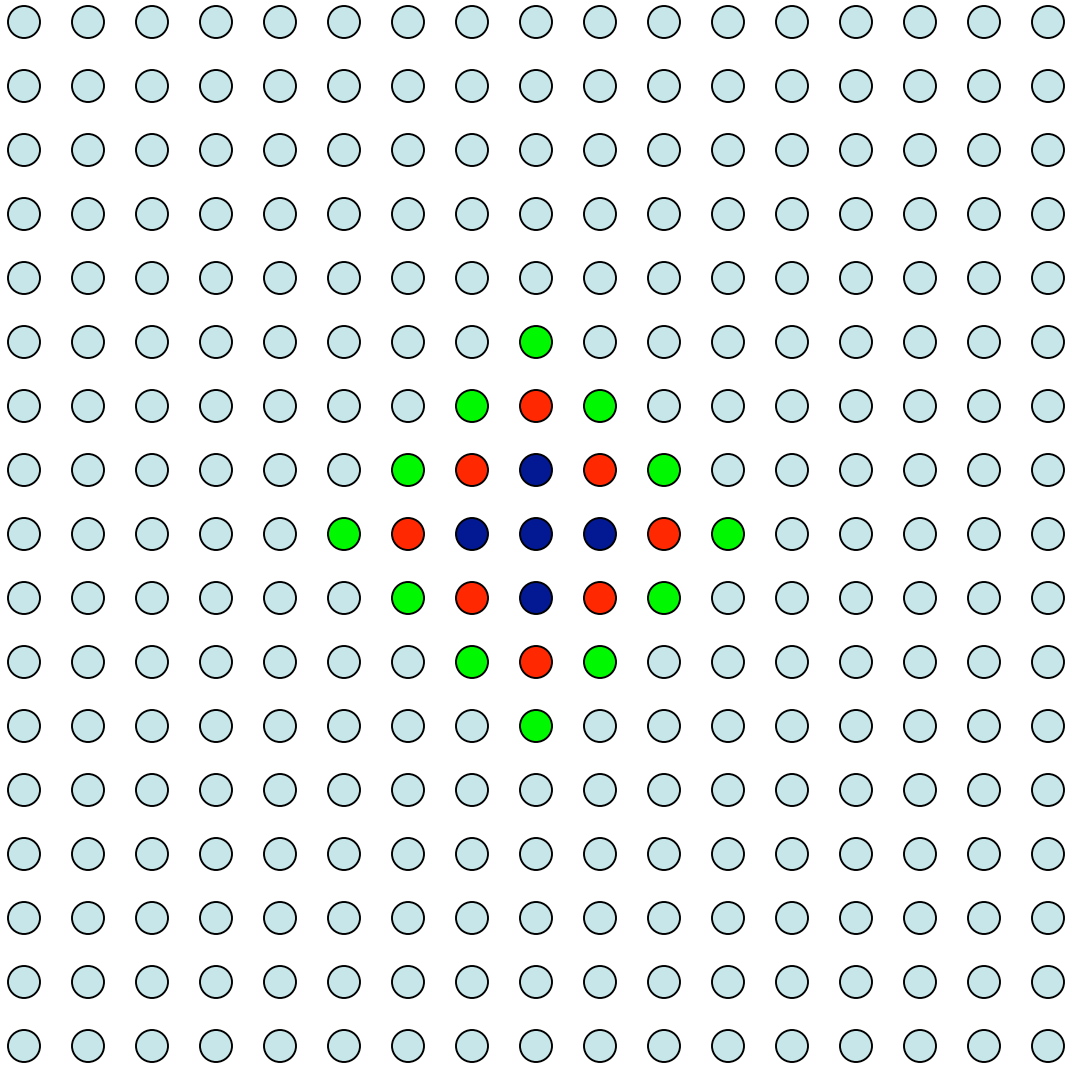
K=1

GOSSIP1



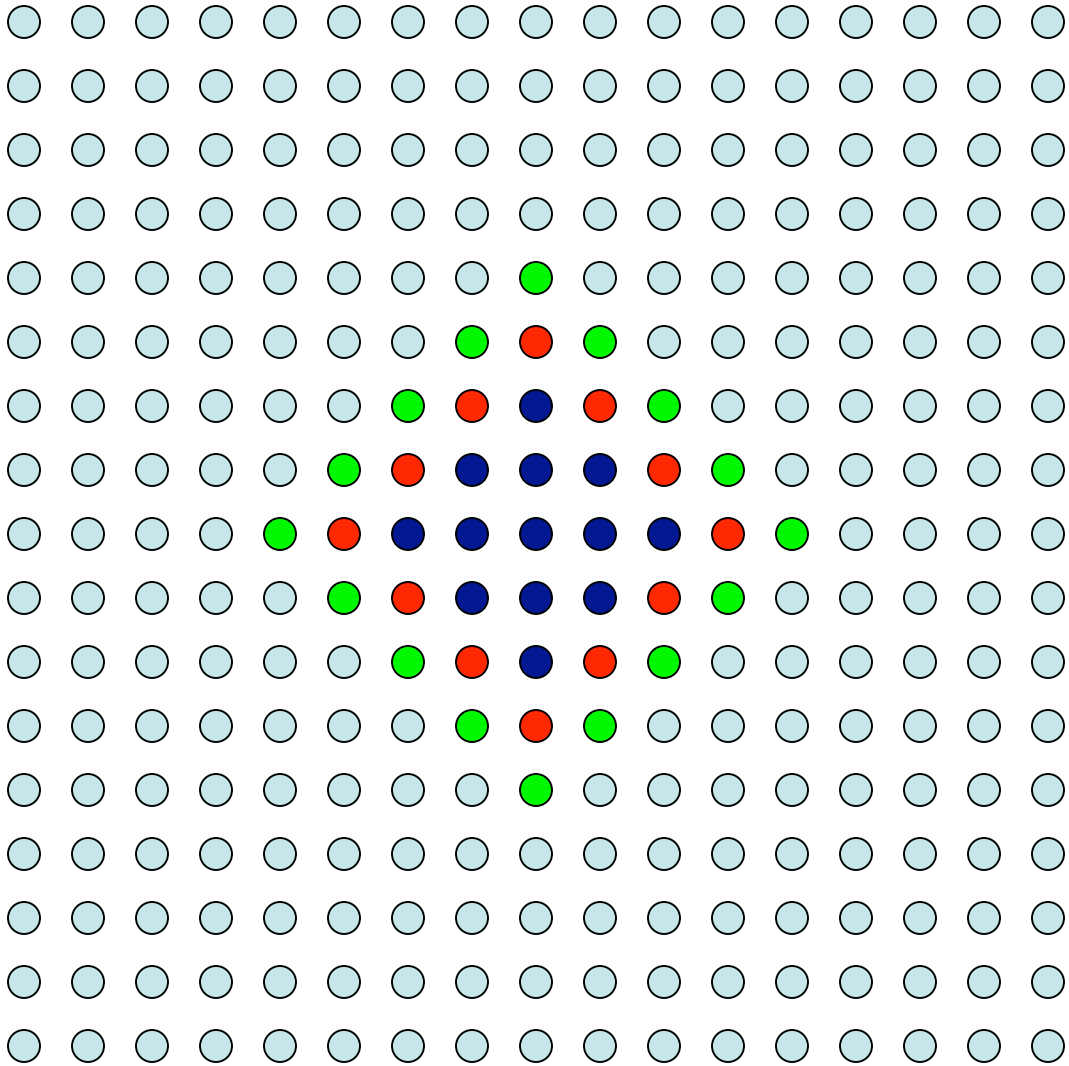
K=2

GOSSIP1



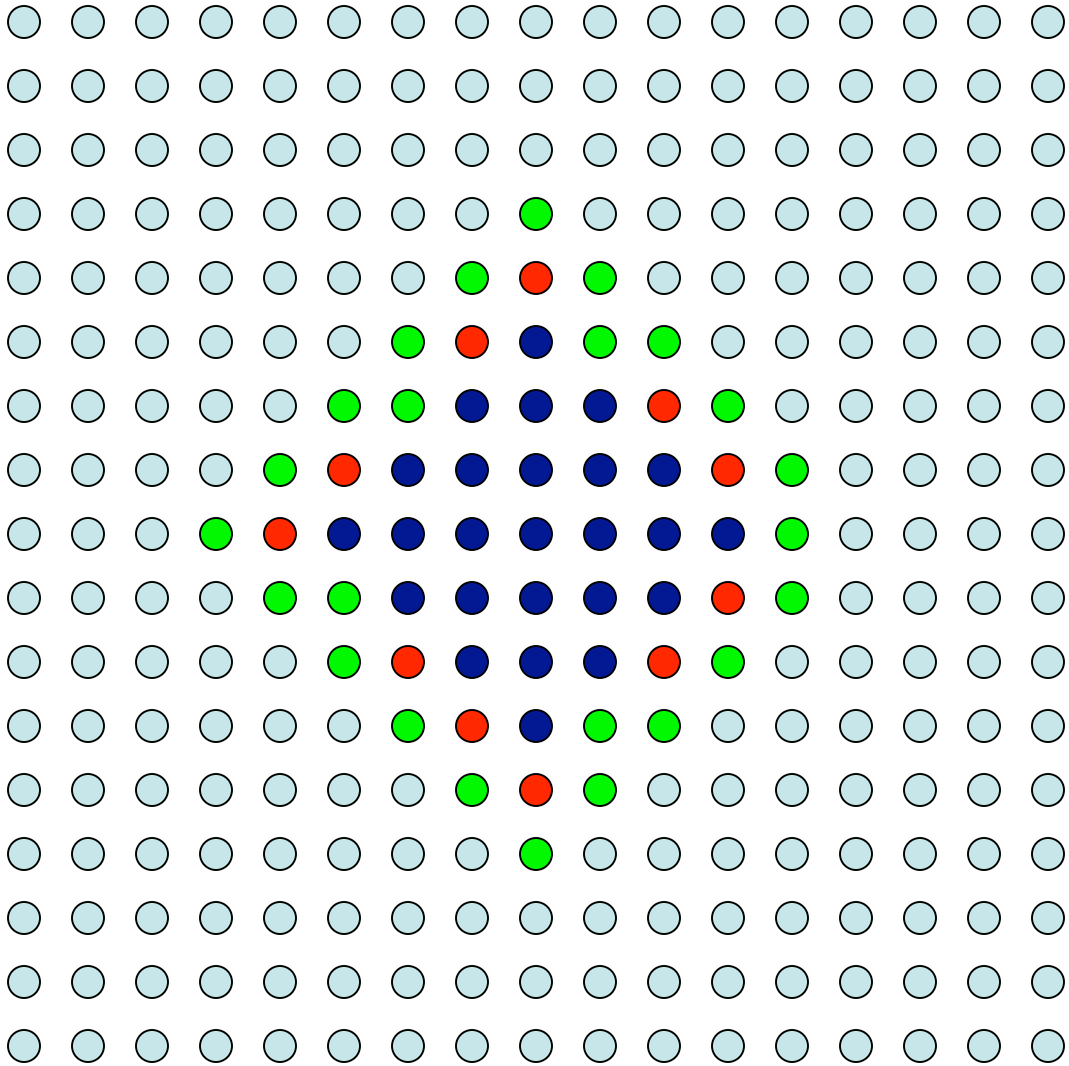
K=3

GOSSIP1



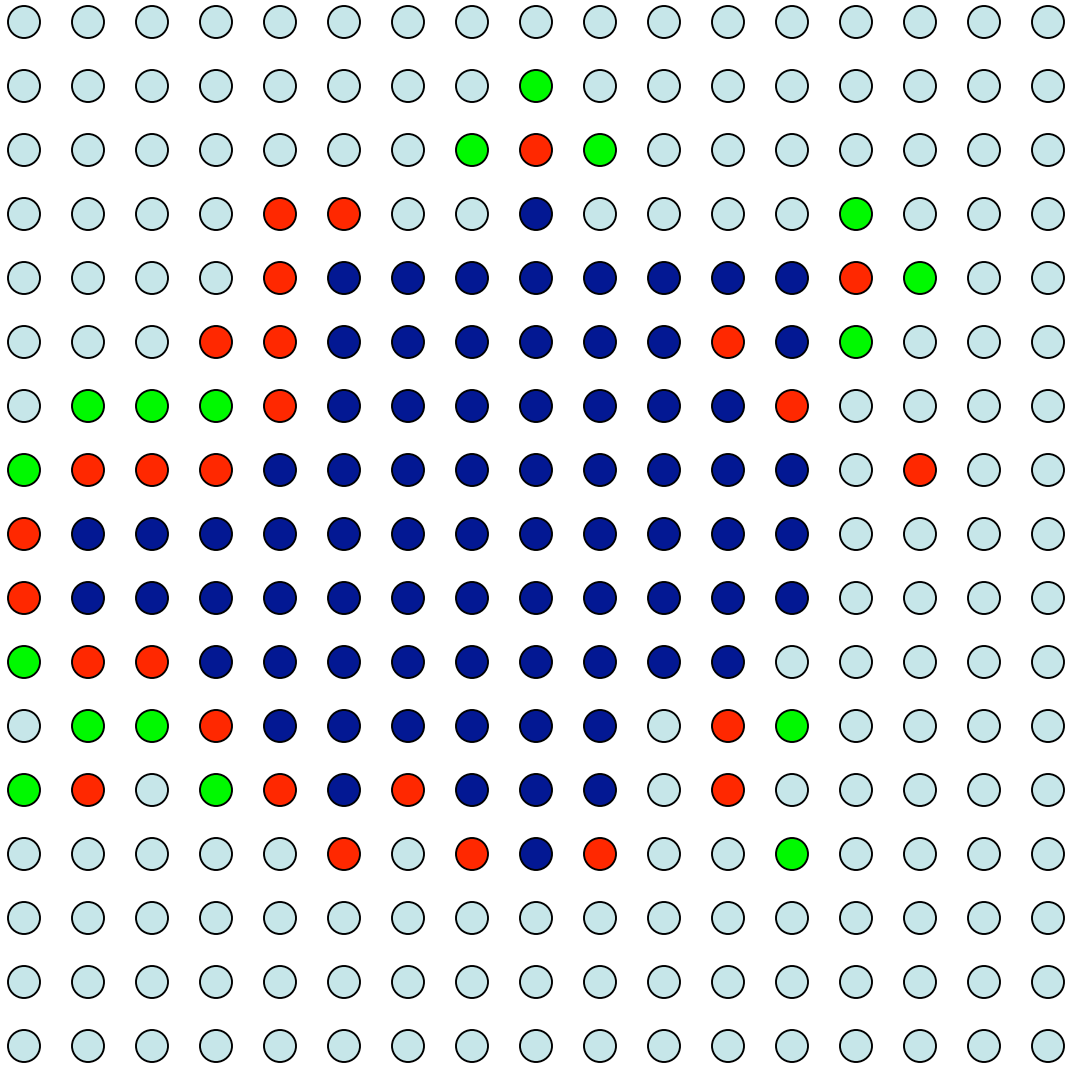
K=4

GOSSIP1



K=5

GOSSIP1



K=8

Concept of a timer

- Asynchronous versus synchronous
- Start a timer at time *t* when message is received for the first time
- Time scale changes from discrete slots to a continuous range
- Reduction of redundancy – major motivation
- Hardware complexity low

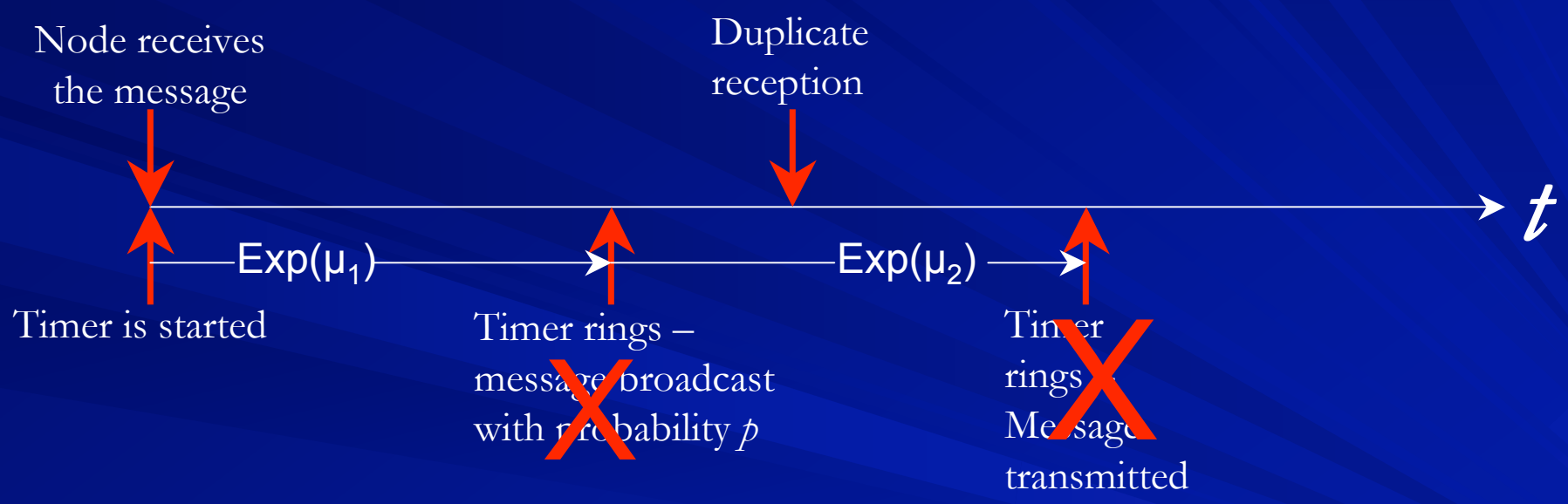
Modified GOSSIP1

- We let flooding run for k steps followed by information dissemination with probability p
- Make it asynchronous – that is node transmits at a random time given by

$$T_{new} = t + \mathrm{Exp}(\mu)$$

- In this case at time T_{new}, node transmits with probability p=1
- "Straw man" protocol
- No usage of local information!

GOSSIP3



- Employs local information
- But low control overhead

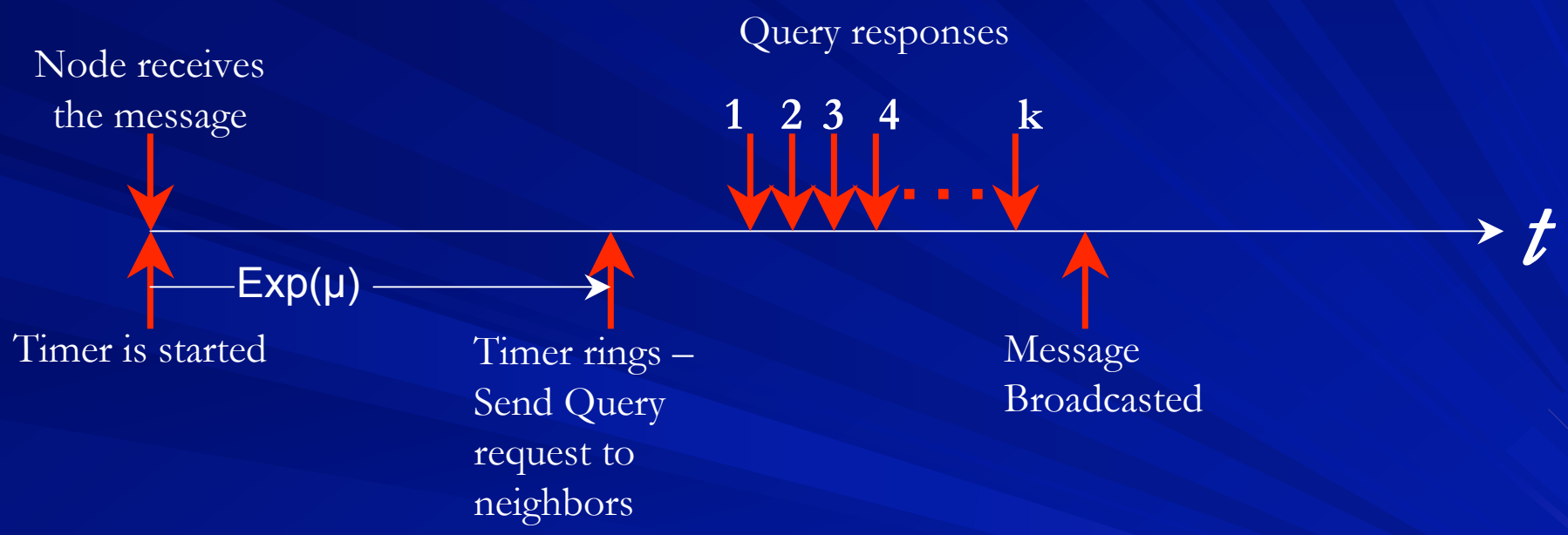
GOSSIP3 [1]

- If Node does not transmit on first reception, set timer and transmit unless atleast *m* neighbors informed
- Not transmitting – good decision or bad?
- Asynchronous version – Set another timer to go off before the first one
- If there is atleast one more reception, do not transmit, else transmit.

SPIN

- SPIN - *Sensor Protocols for Information via Negotiation [2]*
- Negotiation between neighbors.
- Transfer of control and status information
- Three way handshake
  - Node *n* issues ADV message (reception)
  - Nodes that don't have message send REQ
  - Upon receiving atleast one REQ, node *n* transmits

SPIN



Timing Diagram

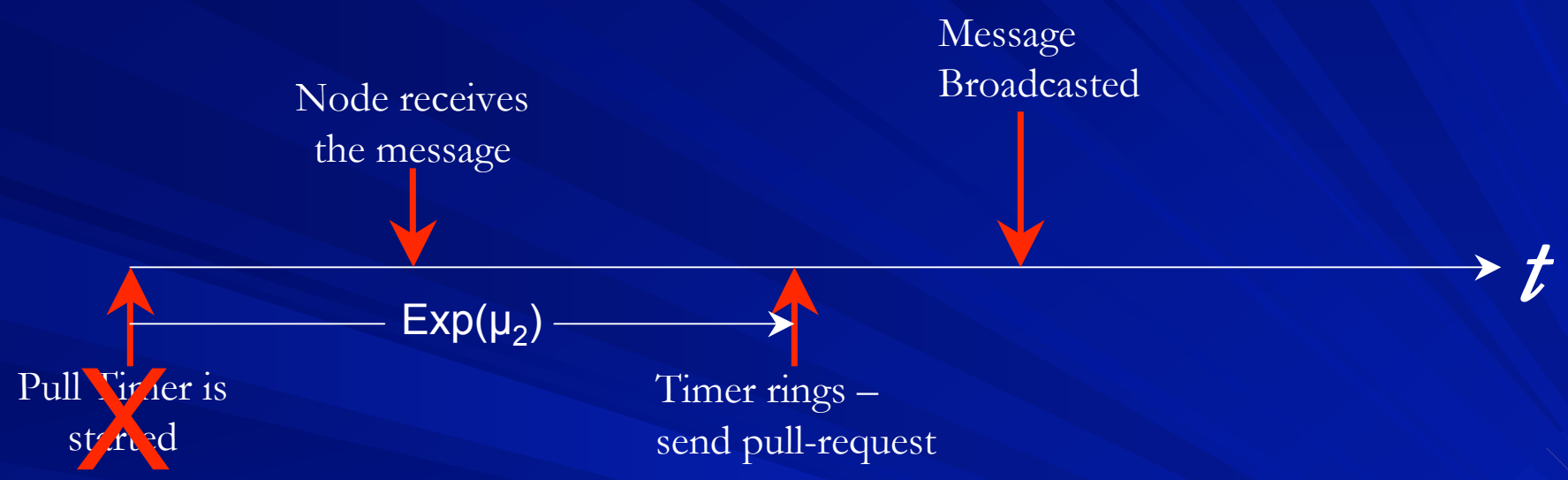
SPIN

- Our modification – separate time between reception and issuing query
- Atleast *k* REQ messages required for the node to broadcast message.
- Active information gathering
- "Asking" not "Listening"
- Control overhead can cause collision, channel contention, etc.

Push-Pull [3]

- Essentially two phases – Push and Pull
- Push phase – inform neighbors on reception
- Pull phase – ask neighbors for message
- When to stop PUSHing and start PULLing?
- Control Message overhead large
- Nodes cannot be sure if their neighbors have the message.
- Neighbor nodes must be synchronized

Push-Pull



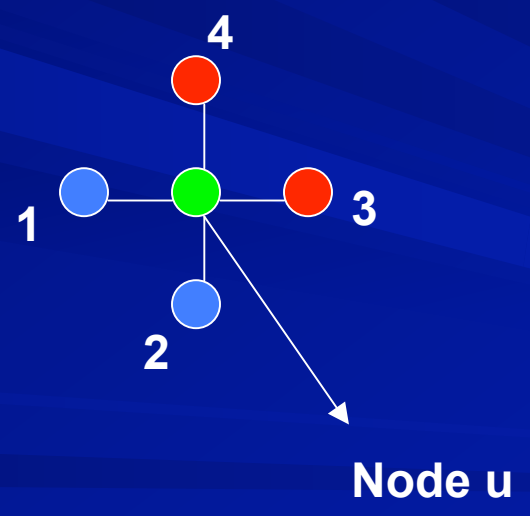
Timing Diagram

Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

SmartGossip

- Each node has a state vector *s* with entries 0 or 1 for each neighbor corresponding to message reception



S_u = [0 0 1 1]

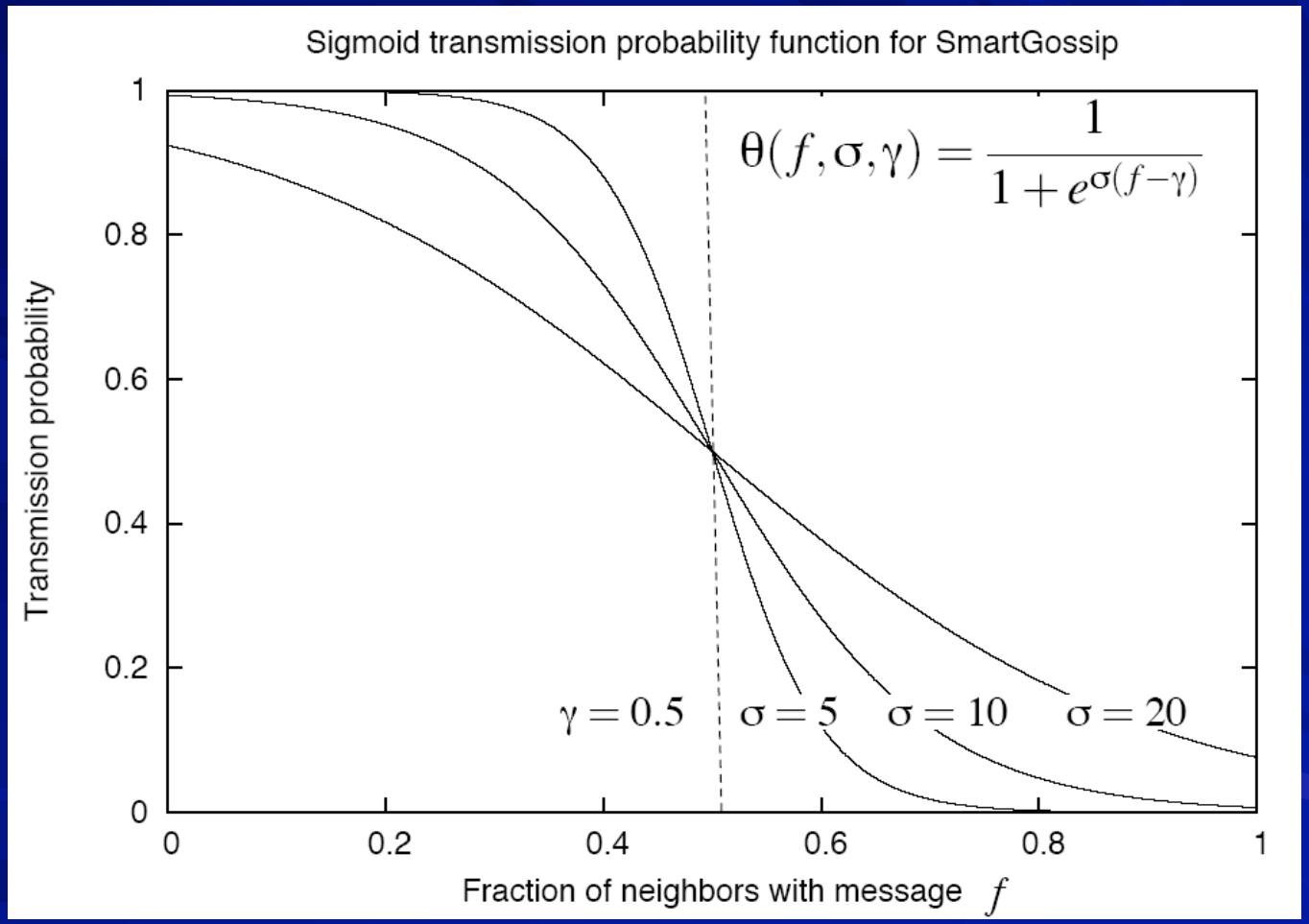
SmartGossip

- Time to transmit depends on whether node *u* is a target or not
- All messages contain a randomly chosen target
- Target nodes transmit earlier
- Probability is calculated by –

$$p = \theta(f,\sigma,\gamma) = 1/(1+\exp\{\sigma(f - \gamma)\})$$

- $\theta(\gamma,\sigma,\gamma) = 1/2$

SmartGossip



Importance of sigmoidal curve for choice of Probability

SmartGossip

- Message Received confirmation – inform neighbors to update local information
- Query Request message – targeted to specific neighbor
- Advantages of SmartGossip –
  - Knowledge of local state with minimal complexity
  - Directed transmissions reduce latency
  - Sigmoidal probability curve
  - Relatively low control overhead

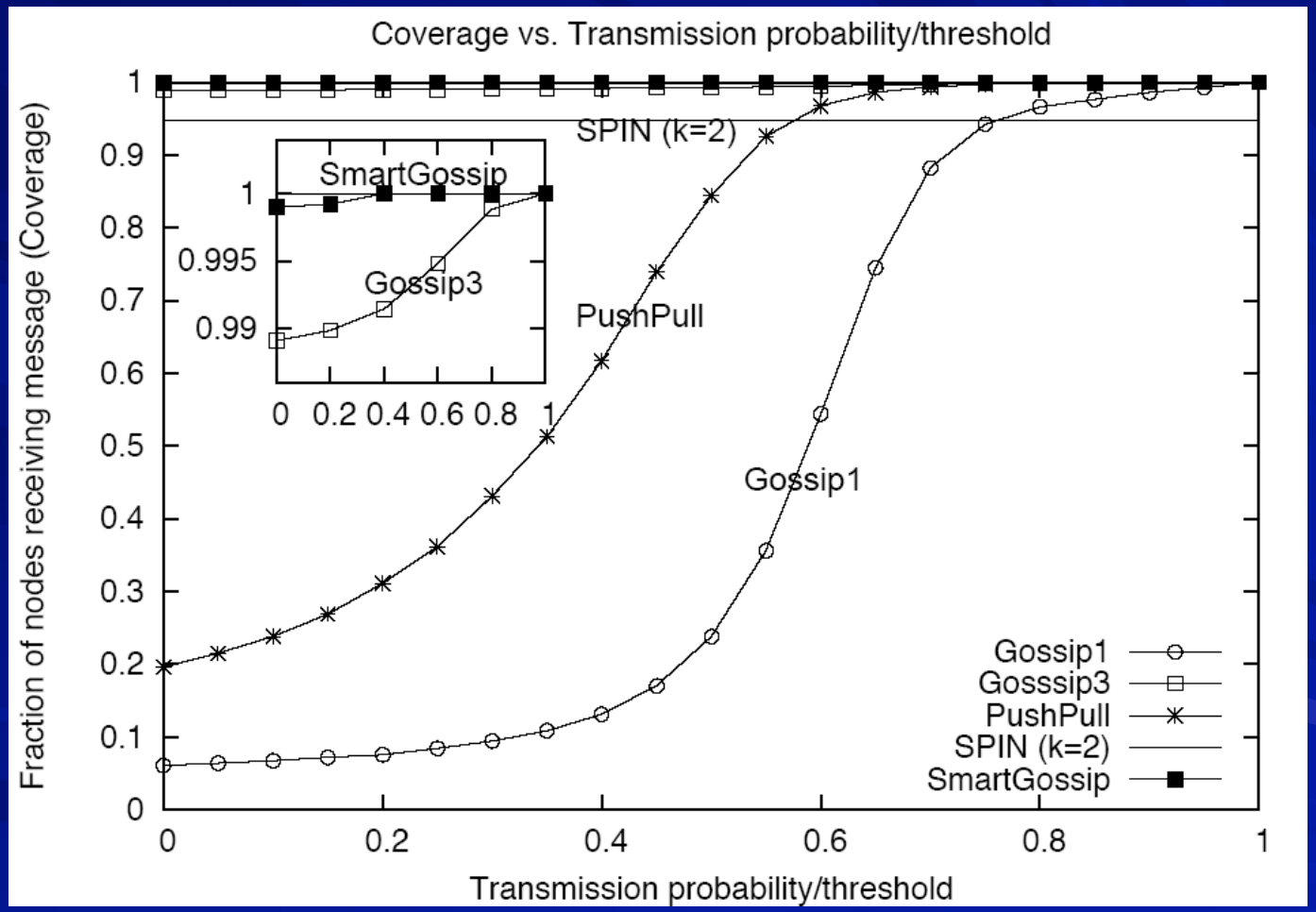
Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future Work

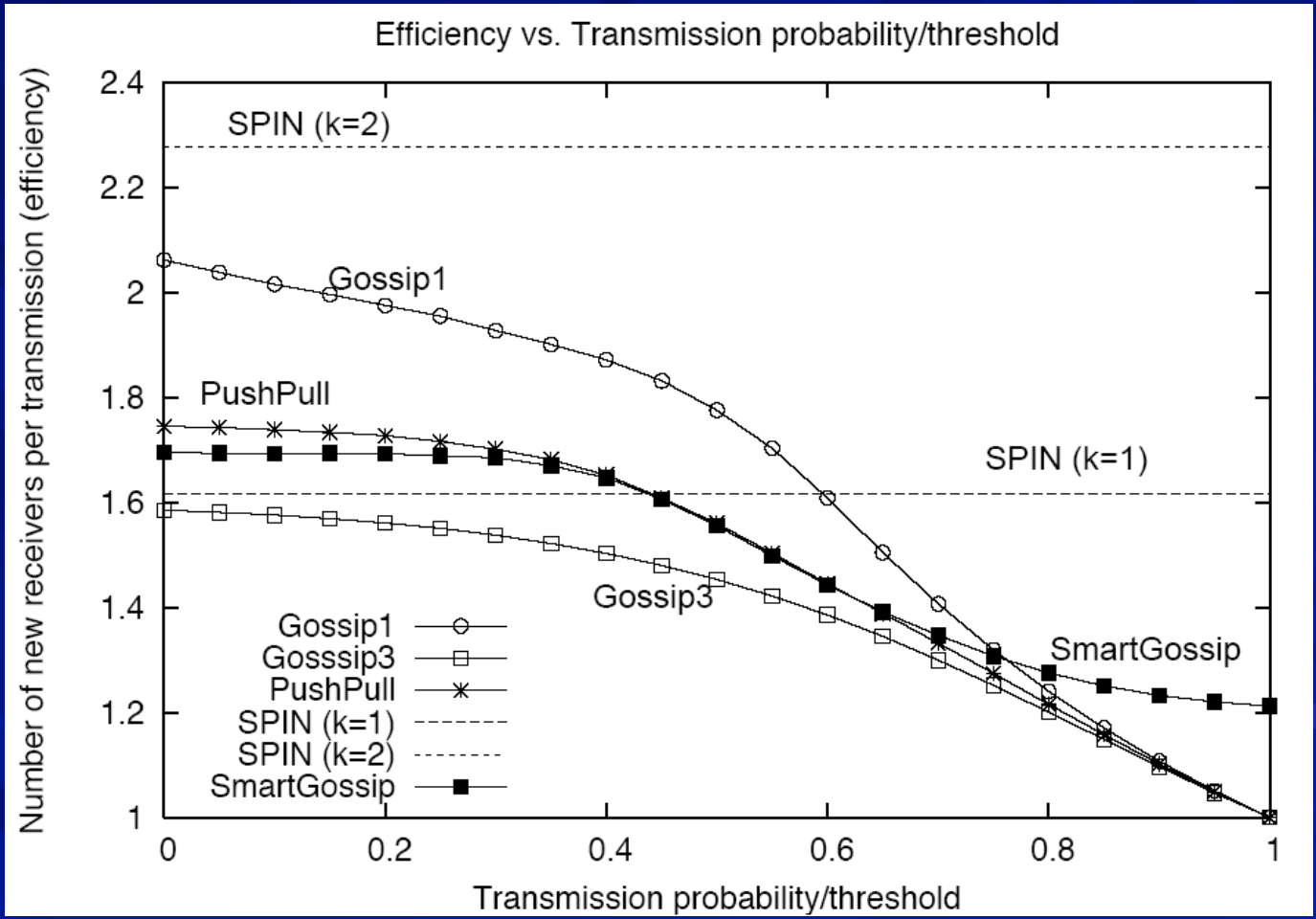
Simulation Setup

- Lattice of 400 nodes
- Simulations averaged over 1000 runs
- Randomly selected source node initiates simulation (message dissemination)
- Comparison of metrics obtained in various protocols
- Measure "quality" of protocol – more coverage, more efficiency and low control overhead and latency
- Optimal parameter values used for protocols

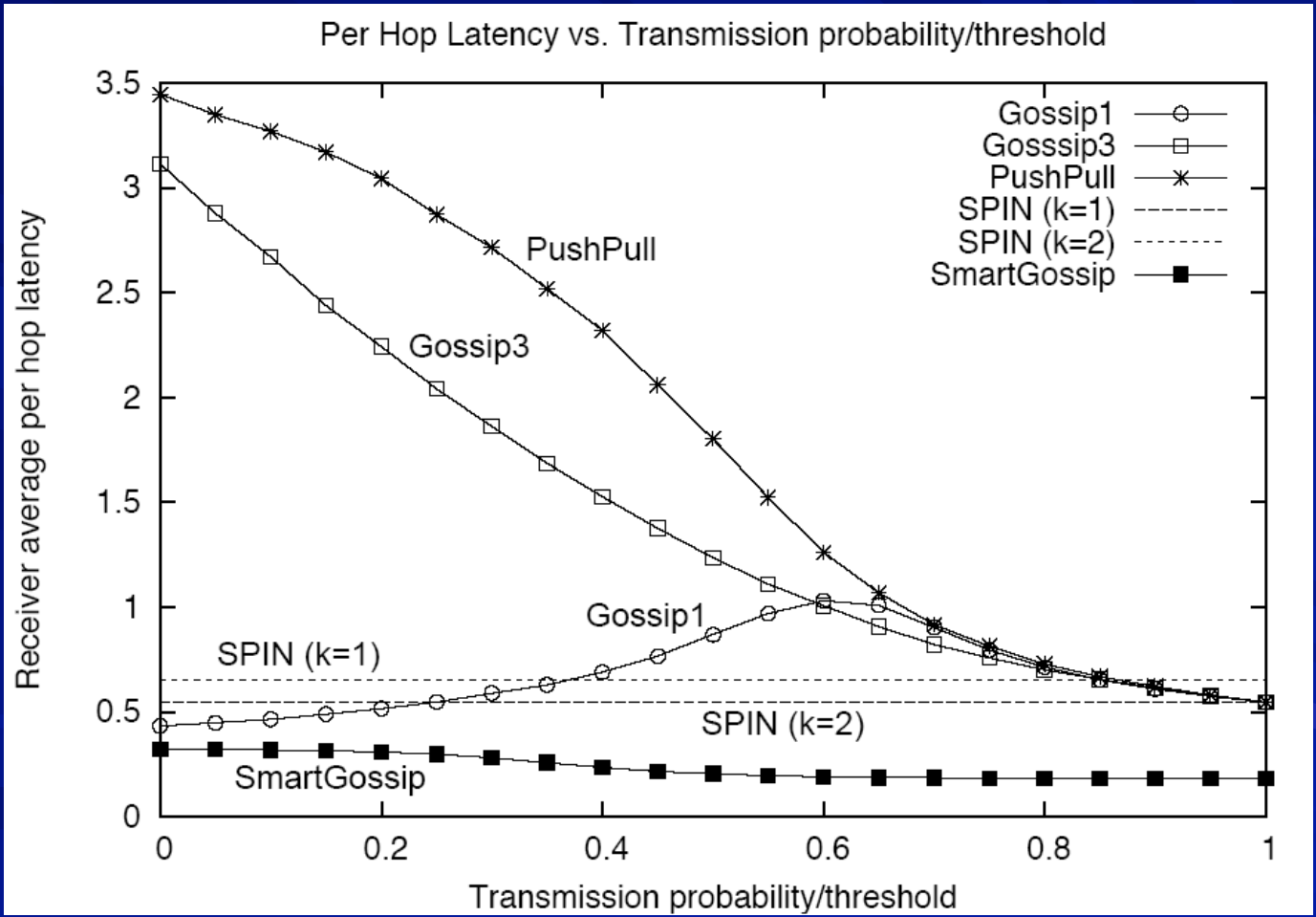
Simulation Results



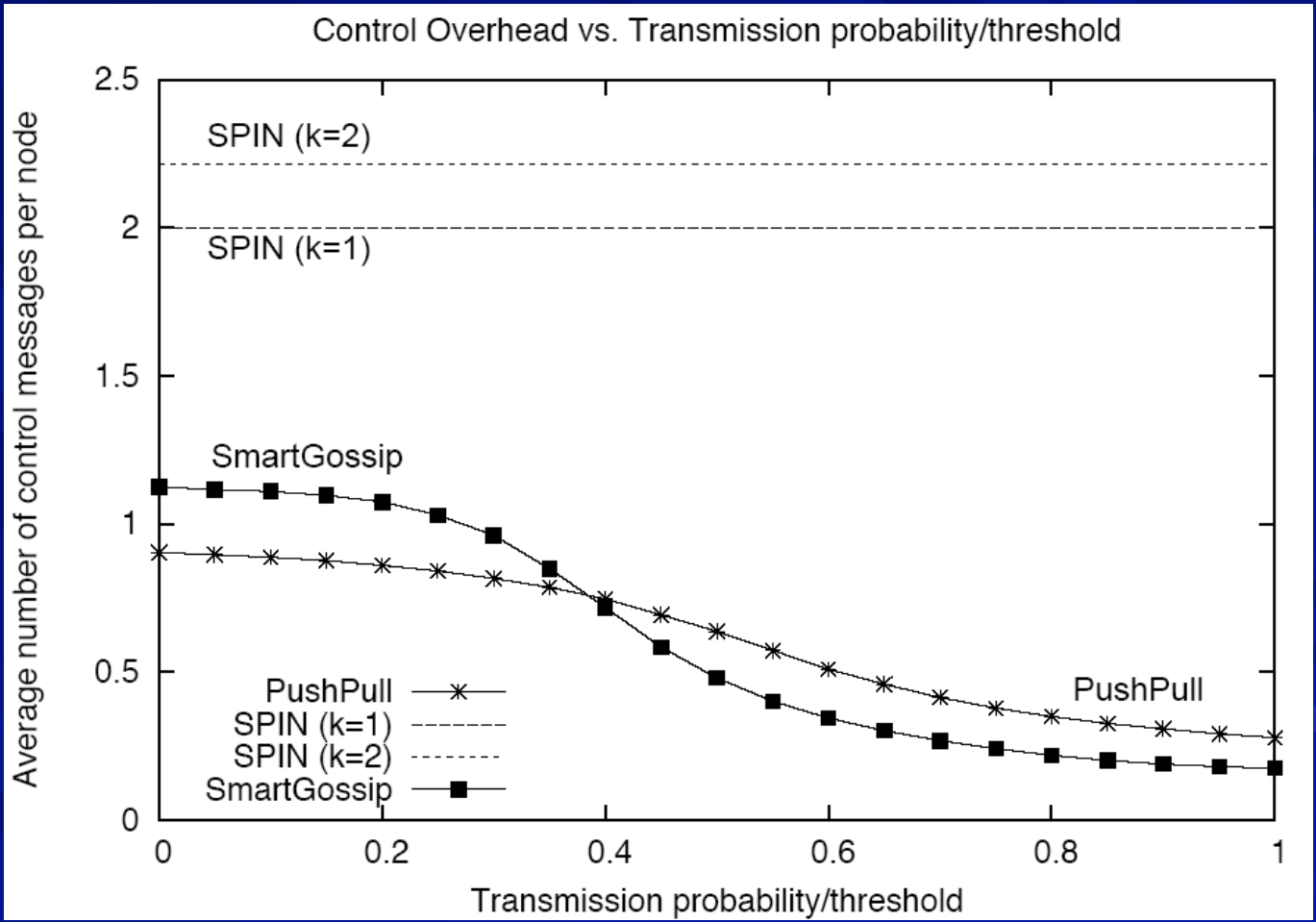
Simulation Results



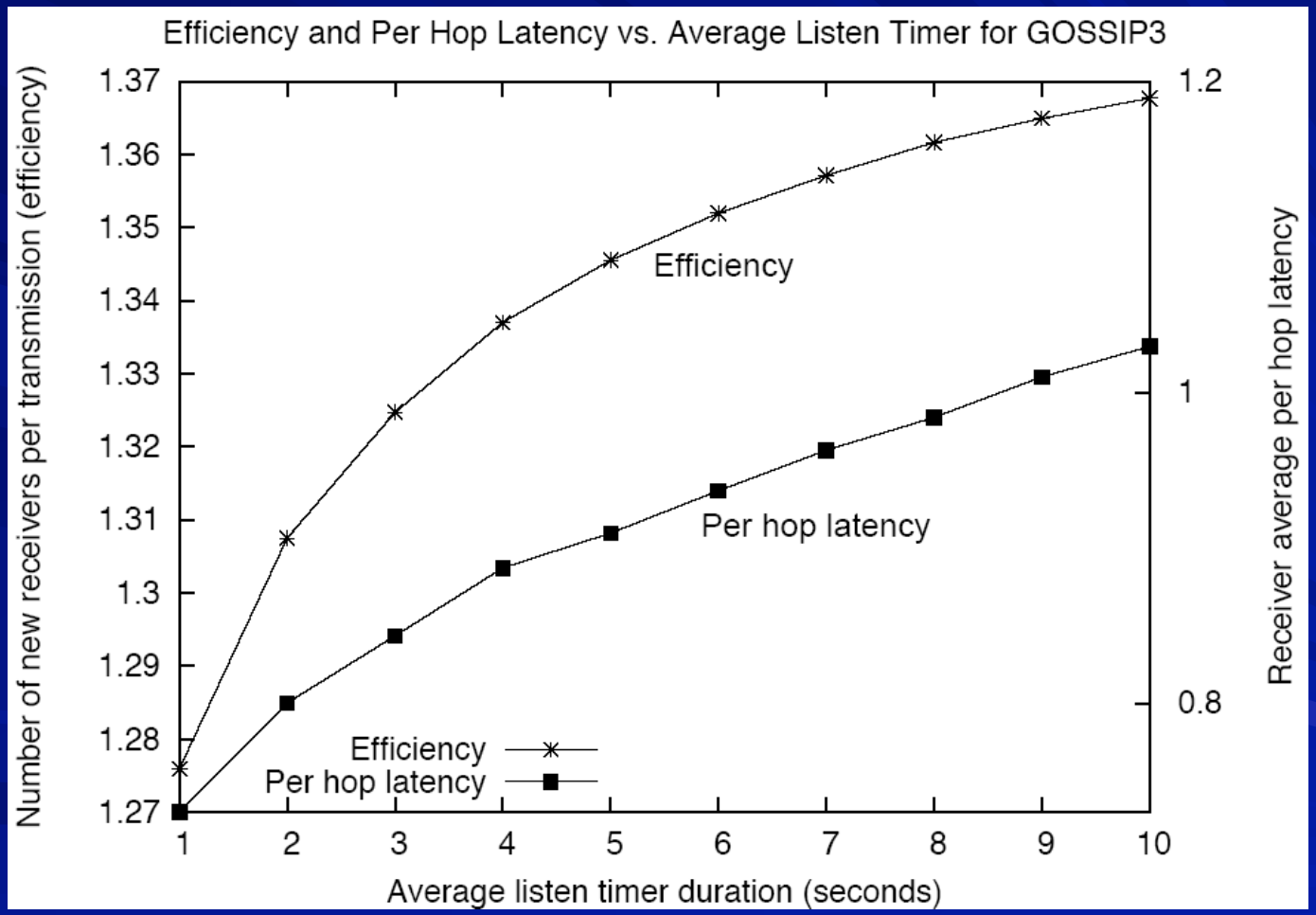
Simulation Results



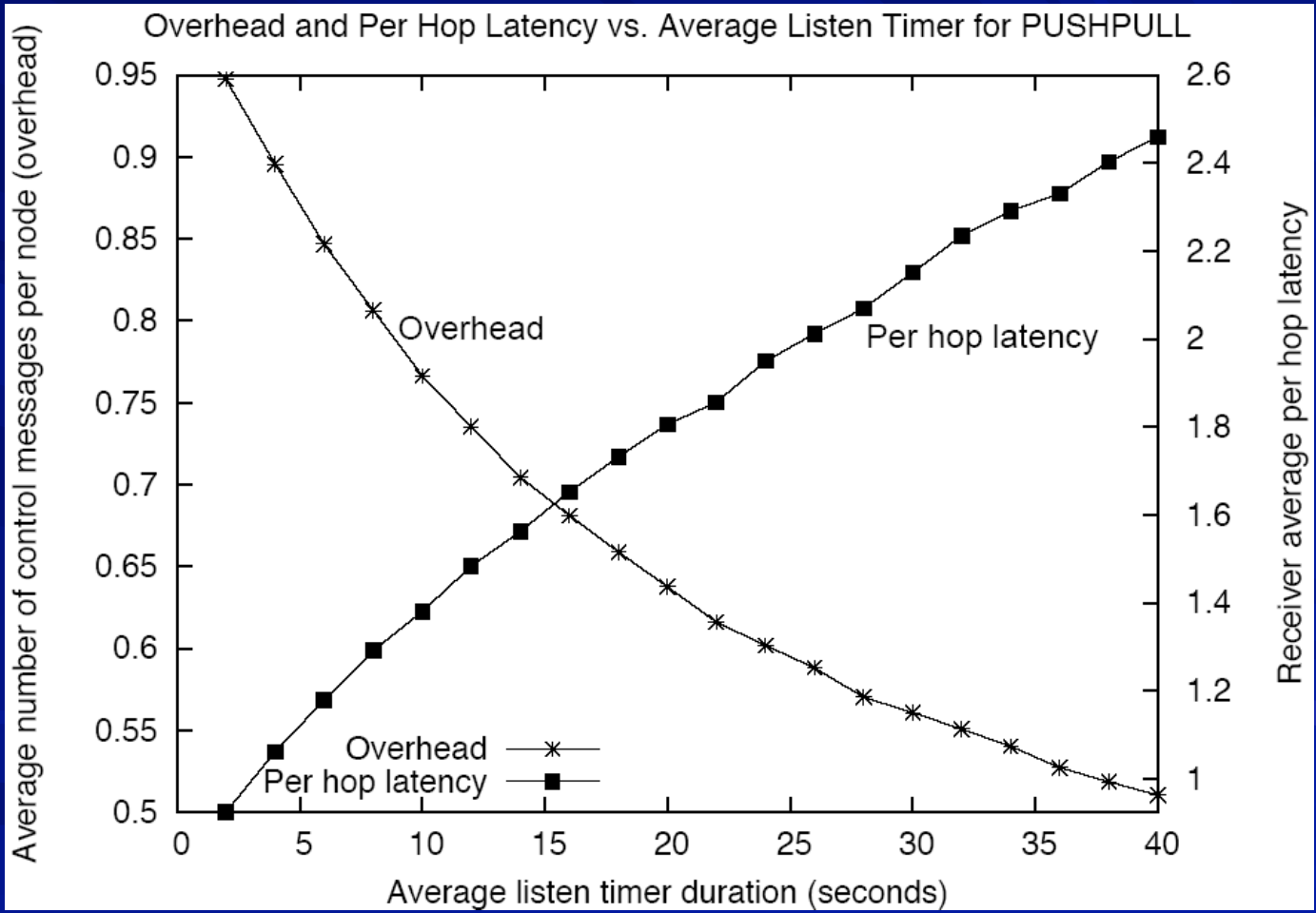
Simulation Results



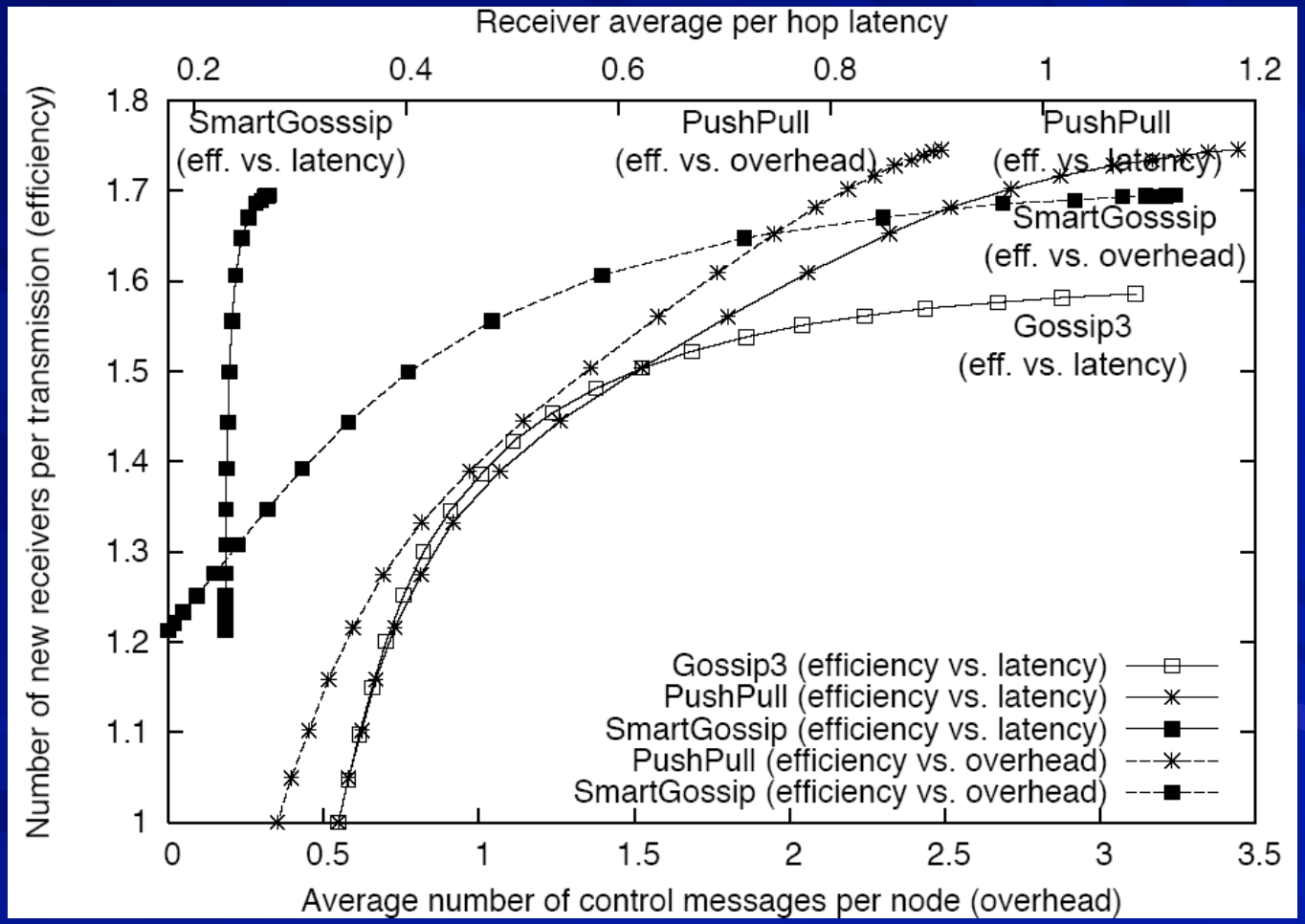
Simulation Results



Simulation Results



Simulation Results



Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

Conclusion

- There is a significant improvement with usage of local information
- Both high coverage and efficiency can be achieved but delays and overheads increase costs
- SmartGossip combines the best elements of known protocols
- Asynchronous transmissions make a difference

Outline of talk

- Background
- Related work
- Explanation of Metrics
- Protocols used
- What is SmartGossip?
- Simulation Setup and Results
- Conclusion
- Future work

Future Work

- Improvement of the SmartGossip protocol
- Active work is being pursued with mathematical analyses of metrics (coverage, efficiency, latency, overhead)
- NS-2 simulations will give us a better feel of the energy utilization
- Apply protocol on different topologies e.g. Random Graphs

Acknowledgments

- Dr. Steven Weber
- Ananth Kini
- Nikhil Singhal
- Other members of the Networking lab
- CTIN group
- ECE department

Relevant References

[1] Z. J. Haas, et al. *IEEE INFOCOM*, 2002, pp. 1707–1716.

[2] W. R. Heinzelman, et al. *MobiCom*, Seattle, WA, 1999, pp. 174–185.

[3] R. Karp, C. Schindelhauer, S. Shenker, and B. Vocking, *FOCS*, 2000.

[4] A. Hajnal, et al. , *Canadian Mathematical Bulletin*, vol. 15, pp. 447–450, 1972.

[5] S. Weber, V. Veeraraghavan, A. Kini, and N. Singhal, submitted to *WiOpt*, Boston, MA, April 2006.

[6] C. Moore and M. E. J. Newman, *Physical Review E*, vol. 61, 2000.

[7] M.-J. Lin and K. Marzullo, *EDCC*, 1999.

[8] D. Kempe, et al.*Proceedings of the 33rd ACM Symposium on Theory of Computing*, 2001, pp. 163–172

[9] G. Grimmett, *Percolation*. New York, NY: Springer-Verlag, 1989.

Questions?

Abstract

- We investigate four performance metrics for randomized broadcast protocols on sensor networks: coverage, energy efficiency, per hop latency, and overhead.
- Our focus is to evaluate the extent to which the exchange of local information (either active or passive) can improve protocol performance.
- We study via simulation three protocols from the literature that exploit local information and compare their performance against the well known GOSSIP1 ([1]) protocol which does not employ any local information in making transmission decisions.
- We study the strengths and weaknesses of the above protocols and propose the new SmartGossip protocol which combines several ideas from the above protocols, as well as several new mechanisms, to achieve superior performance.