SeNDORComm: An Energy Efficient Priority-Driven Communication Layer for Wireless Sensor Networks

Vinaitheerthan Sundaram, Saurabh Bagchi,
Yung-Hsiang Lu, Zhiyuan Li

SeNDOR – Sensor Networks: Detect, Optimize and Repair

Purdue University

Work partially supported by:
National Science Foundation

Outline

- Problem Definition
- Our approach – SeNDORComm
- SeNDORComm Design and Operation
- Evaluation: Experimental, Simulation, and Analytical
- Analysis: Code and memory size
- Related Work
- Summary of Our Contribution

Problem Definition

- Wireless Sensor Networks (WSN)
  - Energy conservation is critical
  - Bandwidth is very limited
  - RAM is precious
- Energy required for communication is much higher than that required for computation
- Therefore, reducing communication can improve energy efficiency as well as network utilization
- Combining messages is an appealing idea to reduce communication traffic. However, …

Problem Definition

- Messages in WSNs have priority. Examples:
  - Debugging Framework - Data messages vs. Debug messages
  - Surveillance Applications - Intruding motor vehicle vs. pedestrian
  - Indoor Climate Control - Harmful gas presence vs. CO_2 reading
- Moreover, in wireless environments, increase in packet size increases the chances of packet getting corrupted.
- Questions:
  - How to combine messages that have priority?
  - What is the trade-off between reliability and energy efficiency?
  - What layer in the radio stack should provide this functionality?

Our approach - SeNDORComm

- Terminology
  - Message priorities: 1 byte priority or 0(highest) – 255(lowest)
  - *Immediate* messages are the messages with the highest priority
  - *Deferred* messages are messages that are not immediate
  - *Packets* are messages in the network layer and below

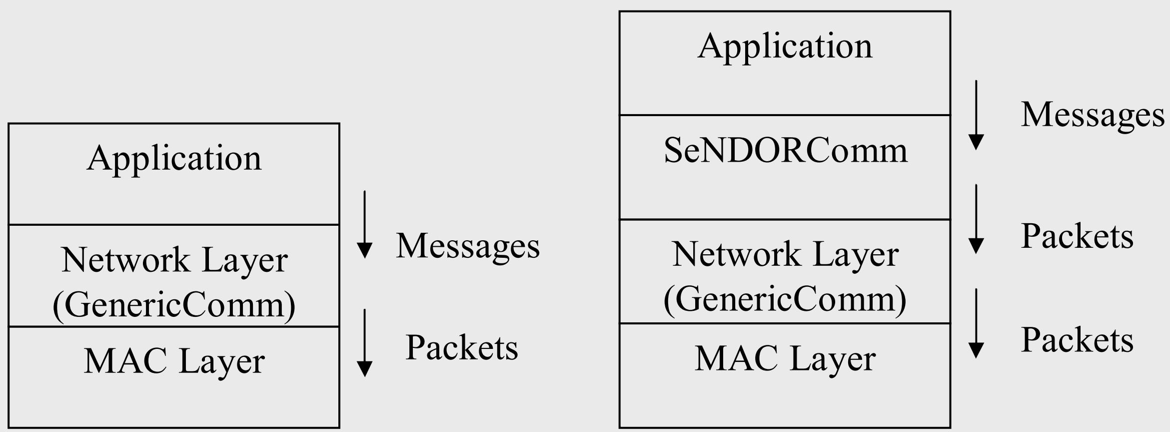
- Key Observations
  - Predominant traffic pattern in WSNs is from motes to the base station
  - Every WSN is designed to send *immediate* messages
  - Bursty traffic exists in WSNs
  - Multiple application components can use the priority-driven communication layer

Our approach - SeNDORComm

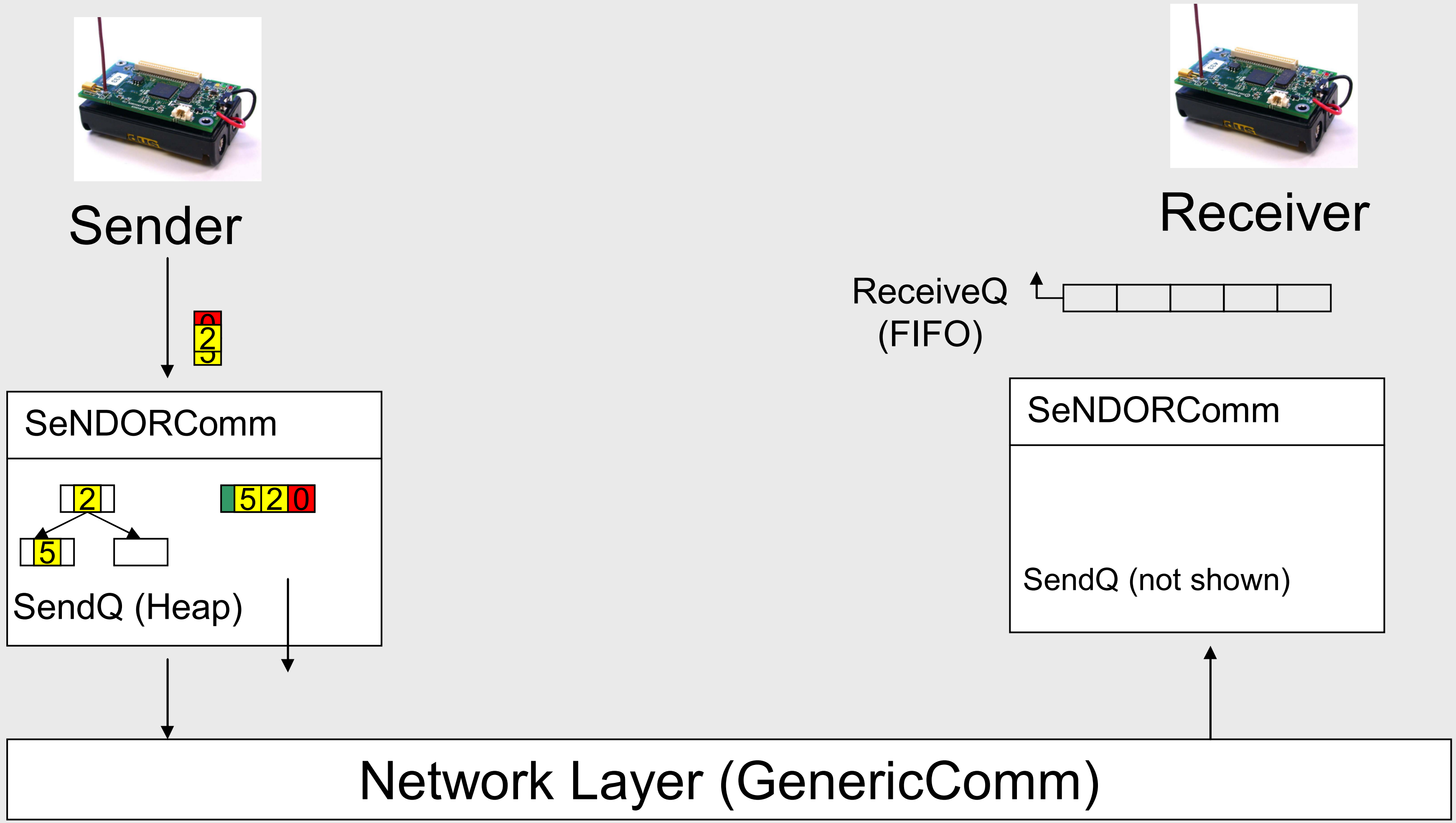
- Design Goals
  - Reduce deferred message traffic to conserve energy
  - Send critical messages promptly
  - Keep the interface close to GenericComm, default communication layer in TinyOS
- SeNDORComm - a priority driven communication layer that satisfies the above design goals
- At the heart of SeNDORComm is the policy for deciding "when to send a message"
  - Immediate messages are sent without buffering
  - Deferred messages are buffered in a priority queue (Q). Later, they are sent out along with immediate messages or as an explicit packet in priority order

SeNDORComm – Where does it fit in the radio stack?

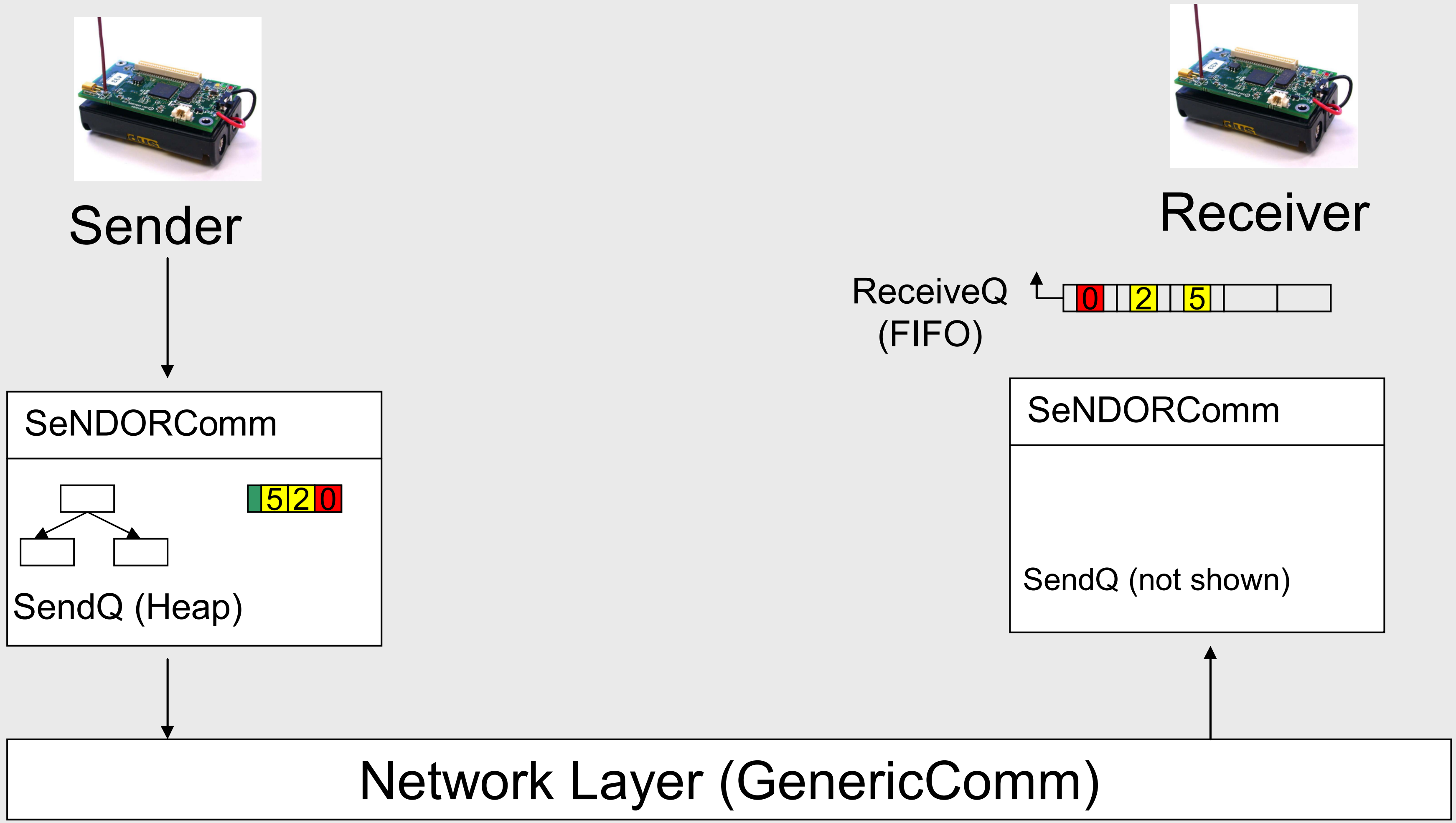
- SeNDORComm is between application and network layer
  - Example: TinyOS Applications Radio Stack
- Why separate communication layer?
  - Useful for many application components
  - Preserves the end-to-end nature of priority
  - Avoids repacking at each intermediate node
  - Can work with any network layer



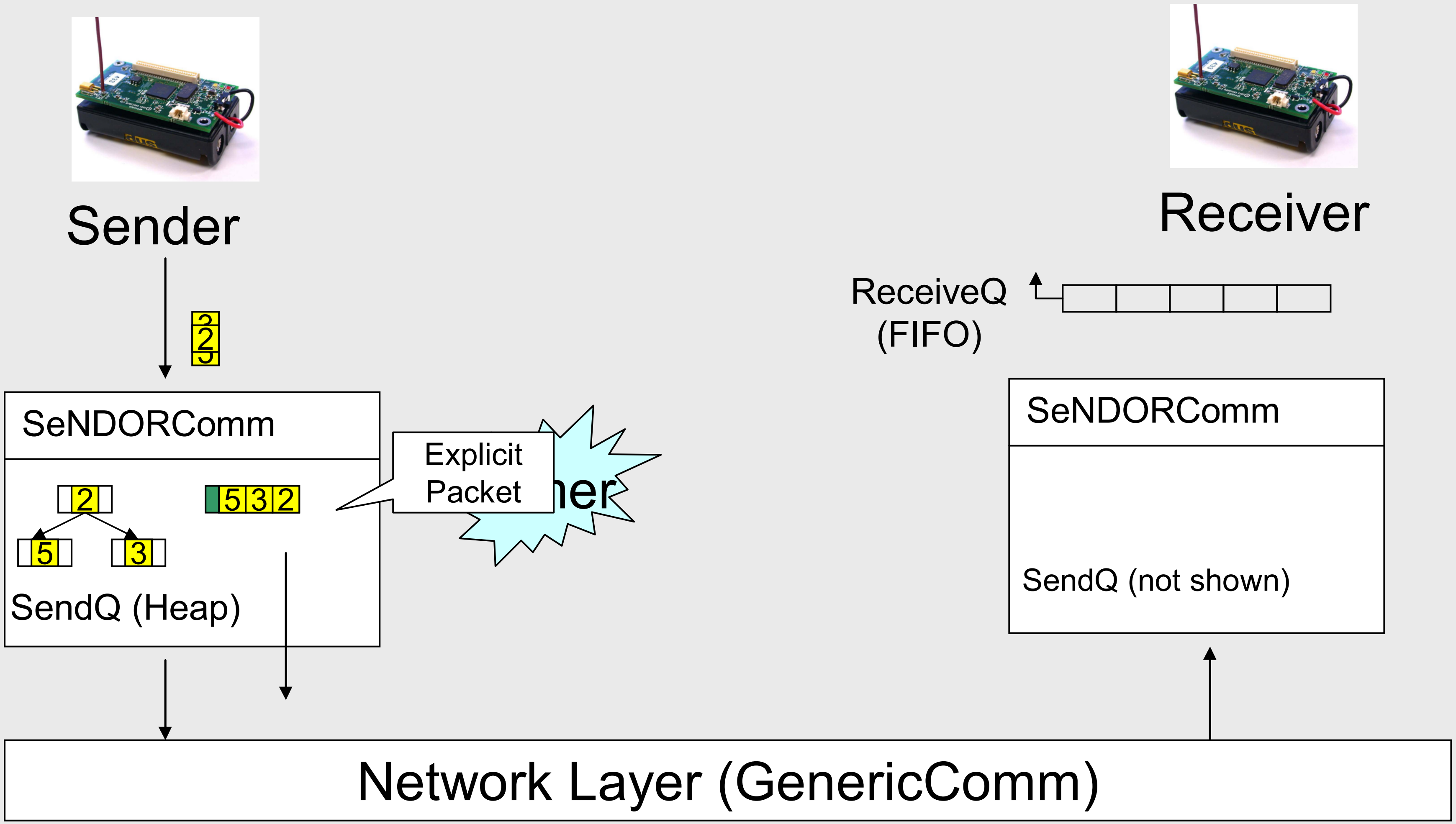
SeNDORComm Operation - Send



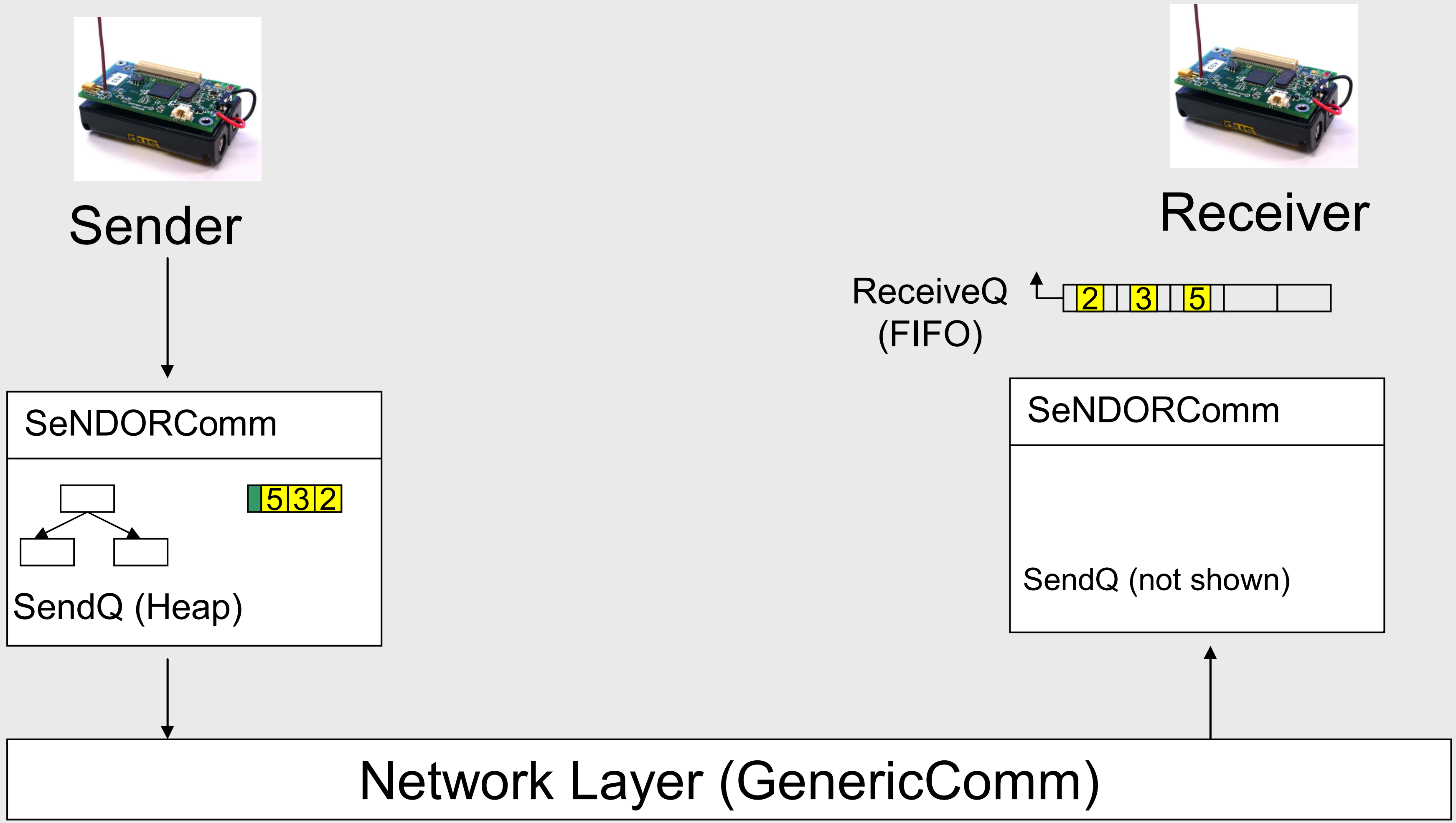
SeNDORComm Operation - Receive



SeNDORComm Operation - Timer Fired



SeNDORComm Operation - Timer Fired



Interface and Implementation

- Same as GenericComm interface, but
  - Send function takes an additional parameter: urgency or priority value
  - Receive function's return value semantics allows application to not return a message buffer

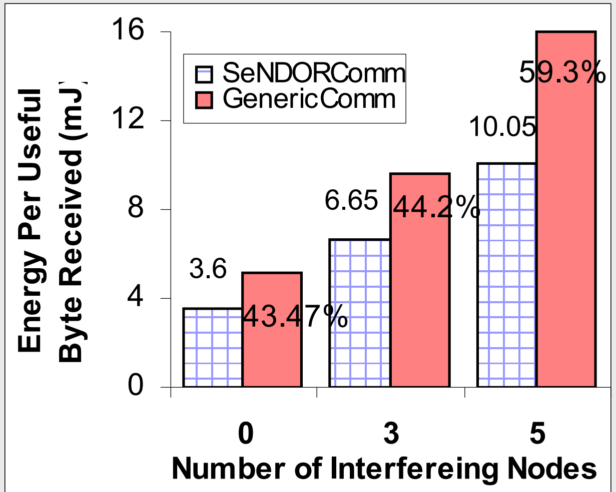
- Priority queue is implemented as a heap of pointers.
  - Internal memory management
  - Only one memory copy per heap update
  - Each heap element has 4 additional bytes

Experimental Evaluation

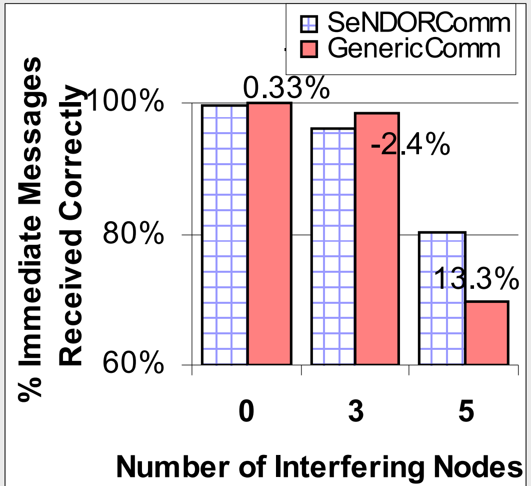
- Goal:
  - To quantify energy efficiency and improvement in message reliability
  - Capability to handle bursty traffic
- Experiment 1: Energy Expenditure under Interference
- Experiment 2: Improvement in Network Utilization
  - A real-world case study using LEACH (a clustering protocol) and HSEND (a debugging framework)

Experiment 1: Energy Expenditure under Interference

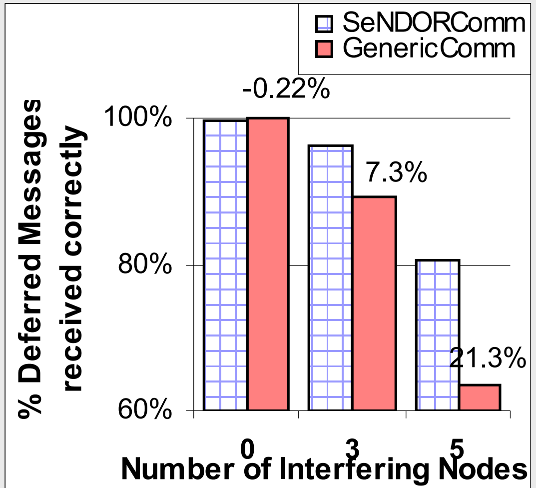
- Two Mica2 motes are kept at 5 meters apart and at 1 meter height
- The sender mote sends 200 messages to the receiver mote
- Interfering nodes send messages at maximum speed possible
- BMAC LPL (Low Power Listening) mode is set to 3 (11.5% duty-cycle)
- Three sets of sub experiments with 0, 3 ,5 interfering nodes



(a)



(b)



(c)

Experiment 2: Network Utilization

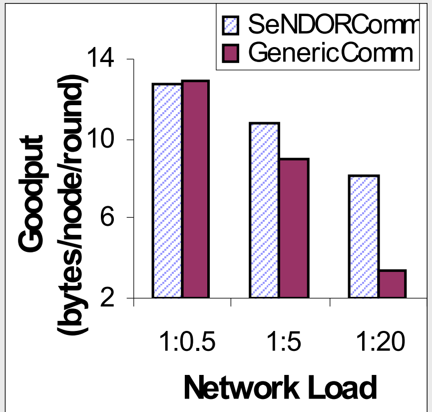
- Real-world case study using LEACH and HSEND
- LEACH [Heinzelman IEEE Trans. Wireless Comm. 2002] - Clustering protocol
  - Nodes organize themselves into clusters, with one node in each cluster acting as the cluster head for one round.
  - Each round has
    - election timeslots - used to elect a cluster head
    - data timeslots - used to send data to the cluster head.
  - In election timeslots, the self-elected cluster heads advertise their status. Nodes that are not cluster heads choose one of the cluster heads to join based on received signal strength.
- HSeND [Herbert IEEE SUTC 2006] - Invariant- based Error Detection Framework
  - Sends alert messages to base station when error is detected
  - Sends information messages to clusterhead to detect a global invariant

Experiment 2: Setup

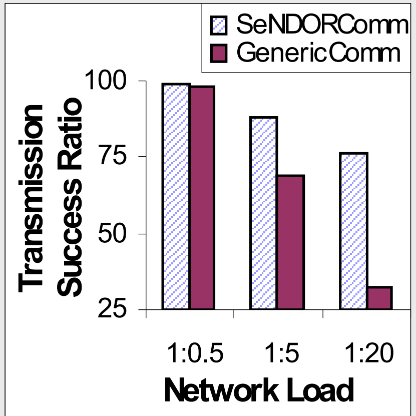
- A 21 node Mica2 motes arranged in 2x1 grid
- Leach parameters:
  - Round = 27 slots - 20 data slots and 7 election slots
  - Each slot is 2 seconds
  - 2 clusters - 9 nodes on average per cluster
  - 20 rounds per experiment
- H-SEND
  - **An invariant that generates an error message with priority value 3 if the rate of successful transmission of sensed data (immediate messages) at each node is below a certain threshold**
  - We set the threshold to be slightly higher than the node's normal sensed data rate so that on average a debug message is generated at every check.
  - We vary the frequency of checking the invariant to vary the load in the network.

Experiment 2: Metrics and Results

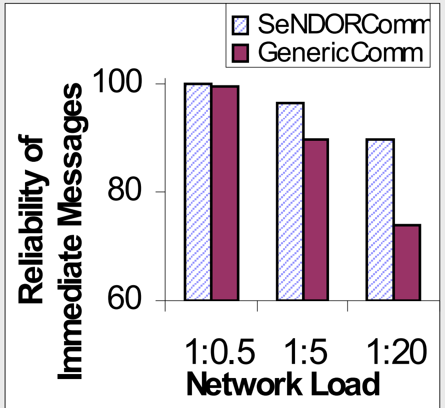
- *Metrics -*
  - *Goodput* - the rate of immediate messages that reaches the base station
  - *Transmission success ratio* - the ratio of the number of messages received by nodes in the network to the number of message sends attempted by nodes including retransmission
  - *Reliability of immediate (deferred) messages* - the ratio of immediate (deferred) messages received by nodes successfully to the total number of immediate (deferred) messages sent by nodes



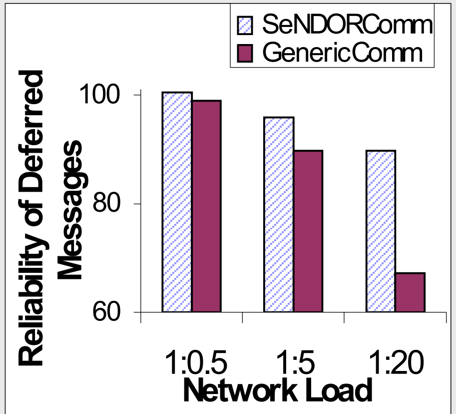
(a)



(b)



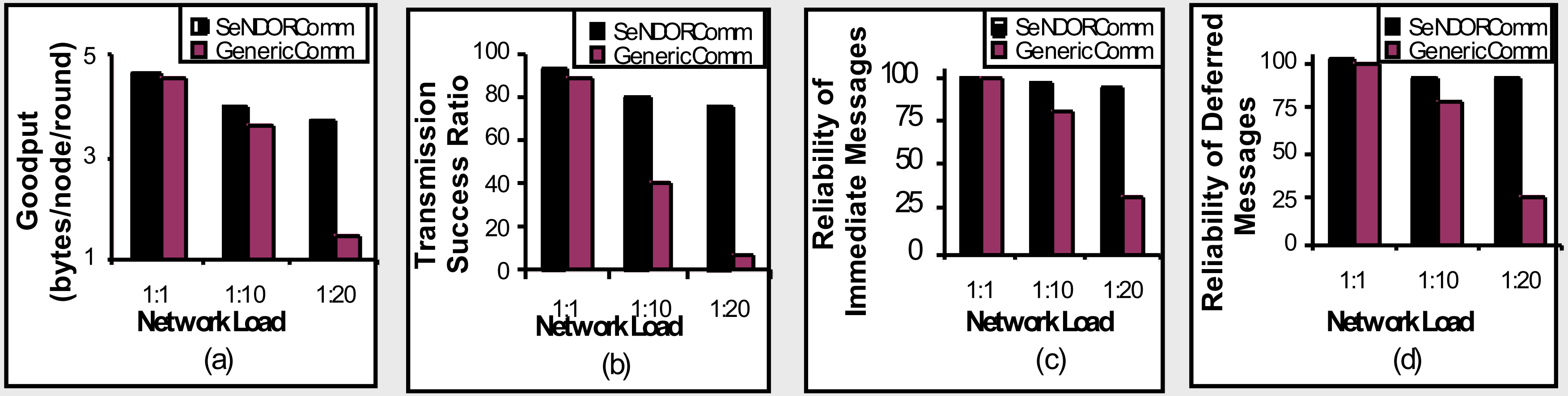
(c)



(d)

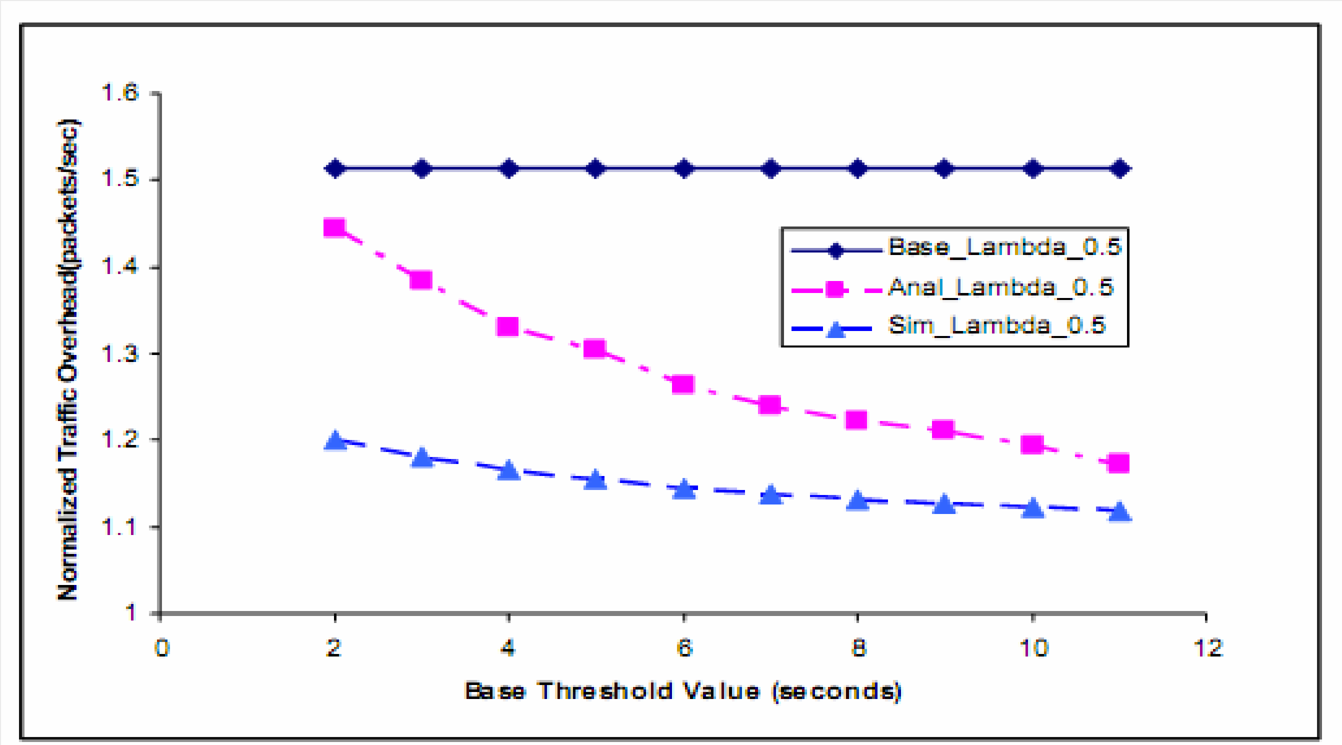
Simulation Evaluation for large networks

- Goal: To show SeNDORComm scales well to large networks
- We simulated experiment 2 for a large network with 100 nodes using TOSSIM (Tiny OS Simulator)
- Results follow a similar trend as in the Experiment 2



Analytical Evaluation

- Goals - Derive an upper bound on the additional traffic injected into the network
  - Guides in choosing the threshold value for the deferred messages



Analysis – Code and Data Memory

- Buffers Size - Runtime Memory Required for data structures maintained

Components	Program Size	Memory Size	Buffers Size
LEACH with GenericComm and Debugging	17884	811	0
LEACH with SeNDORComm and Debugging (1 buffer priroityQ, 2 buffer receiver list)	21812	1118	38
LEACH with SeNDORComm and Debugging (10 buffer priroityQ, 4 buffer receiver list)	21812	1596	676

Related Work

- Priorities: RAP [Lu RTAS 02]
  - Uses priority to do velocity monotonic scheduling
  - Doesn' t consider message combining
- Message Pooling: AIDA [He TECS 03], BMAC [Polastre Sensys 04]
  - Don't use priority
- Congestion Control: [Hull Sensys 04] [He ICDCS 05]
  - Works at mac/network layer
- These works do not consider message combining and application priorities like we do

Discussion

- SeNDORComm guarantee covers passing the message to lower layer in the radio stack
  - The guarantee doesn't cover delivery or even successful send attempt and it's weak because of practical reasons such as predicting wireless channel condition and contention for channel is very hard
- Any-to-any communication in the network
  - By combining deferred messages going to the same station, SeNDORComm can improve energy efficiency in such cases too.
- Congestion can still form under very high load
  - SeNDORComm's admission control can stop the application sending explicit messages to alleviate congestion

Summary

- Energy conservation is critical in Wireless Sensor Networks (WSN)
- Combining messages is an appealing idea
- Challenges
  - Different Priorities for messages exist in WSN
  - Increased packet size reduces reliability in wireless environments
- SeNDORComm
  - A new communication layer that combine messages based on priority without violating timing constraints
- Significant advantages over default communication layer
  - conserves energy, increases network utilization, handles bursty traffic, and prevents congestion
- Future Work: We are working on problem diagnosis in WSN

Q&A

Thank you for your attention!