CS649
Sensor Networks
Lecture 22: Transport Protocols I

Andreas Terzis
http://hinrg.cs.jhu.edu/wsn06/

Outline

- Background
  - Loss rate in WSNs
- Application reliability requirements
- Where should reliability be implemented?
  - MAC
  - Transport layer
- Examples
  - RMST
    - Another example: Wisden

Loss rate in WSNs

- [GKW+],[ZG03] have shown that WSNs face harsh conditions
  - Loss rates can be up to 30%-50% between direct neighbors
  - Loss rates are variable
  - Loss rates cover a large range of values

Application Requirements

- Application requirements vary
  - *Sample-and-collect*: can tolerate errors
    - Scales to large networks? (cf. GDI report [ZG+04])
  - *Network reprogramming*: needs to reliable deliver a (fairly) large object over all the network nodes
- Related issues
  - Fragmentation/Reassembly
  - Fairness (next lecture)
- **As always** power consumption is the primary concern

Where should reliability be implemented? (1)

- MAC layer
  - ARQ mechanisms
    - No ARQ (Benefits?)
    - Unicast messages in 802.11, S-MAC use RTS/CTS/DATA/ACK mechanism for limited retransmissions (Stop-and-Wait)
    - Same mechanism can be extended to bcast/mcast (multiple copies)
    - Selective ARQ

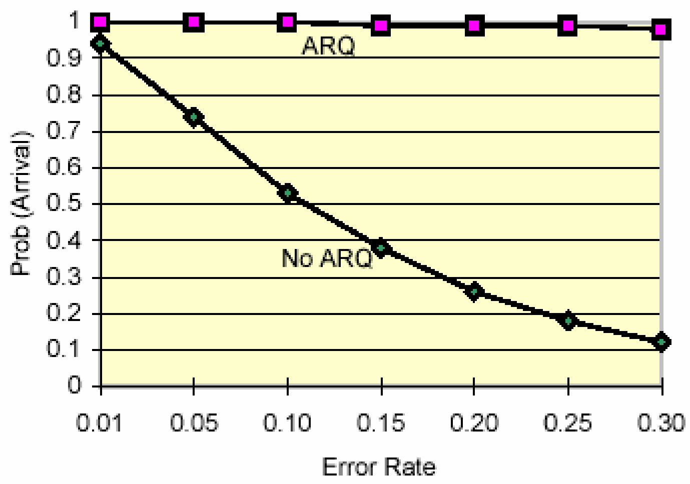
Where should reliability be implemented? (2)

- Transport layer
  - End-to-End retransmission requests
  - Hop-by-Hop NACKs and local repair
- Application layer
  - End-to-End Positive ACK

Analysis of MAC layer retries

- Assuming delivery probability *p* and *R* retransmissions
  - Successful delivery probability across a single hop:
    $$p_h = 1 - (1 - p)^R$$
  - Successful delivery probability across *H* hops:
    $$p_e = p_h^{\ H}$$

Effect of *p*



Probability that message arrives at destination over a 6-hop path

- ARQ with 3 retries operates well over large range of loss rates while performance of No ARQ deteriorates quickly

End-to-End Transport Layer performance

- Assuming large object is divided in *M* fragments and transmitted across *H* hops to destination
- Expected number of fragments arriving at the destination:

$$E\left[f(M,H)\right] = \sum_{m=1}^{M} m \cdot p_e{}^{m} \cdot (1 - p_e)^{M-m}$$

- Expected number of hops that failed packet will travel:

$$E\left[f_h(H)\right] = \sum_{n=1}^{H} n \cdot p_e{}^{n-1} \cdot (1 - p_e)$$

- Approx cost of link-wise fragment transmissions with E2E transport layer:

$$H \cdot E\left[f(M,H)\right] + E\left[f_h(H)\right] \cdot \left(M - E\left[f(M,H)\right]\right)$$

Hop-by-Hop Transport Layer performance

- Data is cached at each hop and transport layer recovery happens on a per-hop basis
- Expected number of retries to successfully transmit a fragment over single hop:

$$E[r(K)] = \sum_{k=1}^{\infty} k \cdot p_h \cdot (1 - p_h)^{k-1}$$

- Number of link-wise transmissions to send *M* fragments over *H* hops:

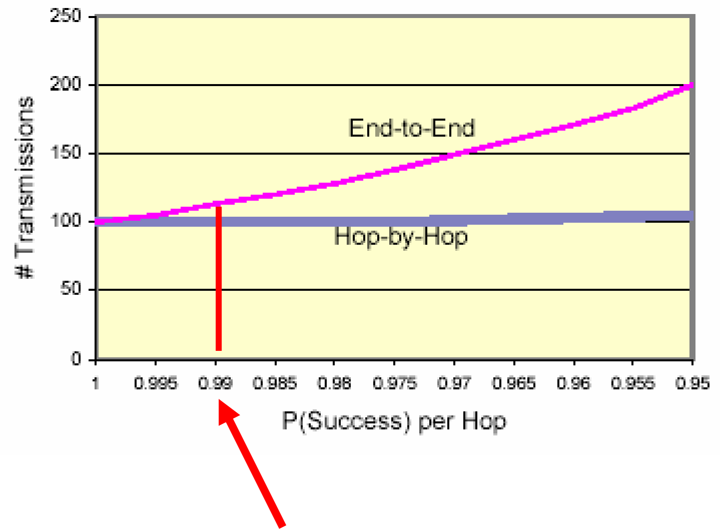
$$E[Tx(H, M)] = M \cdot H \cdot E[r(K)]$$

Comparison

Number of total transmissions to send *M* Fragments over *N* hops (*p=0.9)*

Frag-ments	5 Hops		10 Hops	
	Cache	No Cache	Cache	No Cache
5	27.77	42.33	55.55	143.39
10	55.55	84.67	111.11	286.79
20	111.11	169.35	222.22	573.59

Effect of per hop success probability

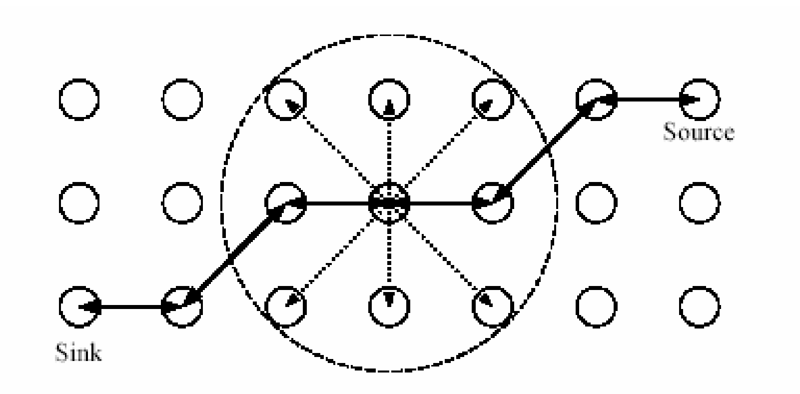


If per-hop loss rate is > 1% then E2E requires many more retransmissions

RMST Overview

- Assumption: Works with Directed Diffusion
- Model: Eventual delivery of fragments to the destination(s)
  - No in-order delivery
  - No delay guarantees
- *Receivers* detect when fragment needs to be re-sent
  - Final destination or intermediate nodes can be *receivers*
- Receivers detect losses by “holes” in sequence numbers or by using timers
- After loss is detected, receiver send NACK to upstream node (using directed diffusion route)
  - Multiple requests can be sent in a single NACK

Simulation Evaluation



- Environment
  - 802.11 MAC layer
  - Directed Diffusion
  - ns-2 simulation
- Parameters
  - Link Error Rate
  - Hop Count (6)
  - Number of MAC retries (4)
  - Message size (5K broken in 50 100-byte fragments)
- Metric
  - Ratio of **total** number of messages normalized to the cost of sending the message without ARQ or transport layer overhead

Simulation Results (1)

- Baseline E2E positive ACK
  - At low-loss rates MAC introduces overhead
    - Selective ARQ more efficient
  - When loss rate increases No ARQ requires many retransmissions
- RMST with Hop-by-Hop Recovery
  - At medium loss rates transport recovery offers very little added benefit over MAC recovery
  - At p=0.1 No ARQ 15% better than Selective ARQ

PHY Error Rate	No ARQ	ARQ All	Selective ARQ
0	.93 (.07)	.57 (.03)	.65 (.03)
.01	.51 (.04)	.56 (.03)	.61 (.05)
.10	.21 (.05)	.47 (.09)	.54 (.06)

Table 2: End-to-End Positive ACK
Normalized byte transmissions required for diffusion to transfer 50 fragments of 100 bytes across 6 hops without any transport layer

PHY Error Rate	No ARQ	ARQ All	Selective ARQ
0	.99 (.05)	.60 (.06)	.68 (.06)
.01	.95 (.06)	.57 (.06)	.67 (.07)
.10	.76 (.07)	.48 (.07)	.61 (.07)

Table 3: Hop-by-Hop Selective NACK and Caching
Normalized byte transmissions required for diffusion to transfer 50 fragments of 100 bytes across 6 hops with hop-by-hop caching and repair

Simulation Results (2)

- RMST with E2E recovery (NACK)
  - At p=0.1 very high loss
    - Hop-by-Hop recovery mechanism at transport of MAC is necessary
  - Little difference between E2E and hop-by-hop NACKs on top of MAC-level recovery
- High Loss
  - Compare schemes that performed best at p=0.1
  - No-ARQ breaks (directed diffusion cannot create paths)
  - E2E-RMST and HBH RMST perform similarly over Selective ARQ

PHY Error Rate	No ARQ	ARQ All	Selective ARQ
0	1.0 (.05)	.61 (.08)	.67 (.07)
.01	.90 (.06)	.60 (.10)	.66 (.07)
.10	n/c	.49 (.09)	.61 (.07)

Table 4: End-to-End Selective NACK
Total byte transmissions required for diffusion to transfer 50 fragments of 100 bytes across 6 hops with end-to-end repair.

PHY Error Rate	Hop by Hop RMST NoARQ	Hop by Hop RMST Sel ARQ	End to End RMST Sel ARQ
.20	.48 (.19)*	.40 (.18)	.40 (.17)
.30	n/c	.24 (.23)	.27 (.25)

Table 5: High Error Rate Test
Total byte transmissions required for diffusion to transfer 50 fragments of 100 bytes across 6 hops with high error rates.

Wisden

- A wireless multi-hop sensor network based data acquisition system for structural health monitoring.
  - Reliable data delivery over multiple hops.
  - Time-synchronized data delivery from multiple sensor nodes.
  - Data compression at the source node to relieve bandwidth bottleneck.
  - Ease and flexibility of deployment.

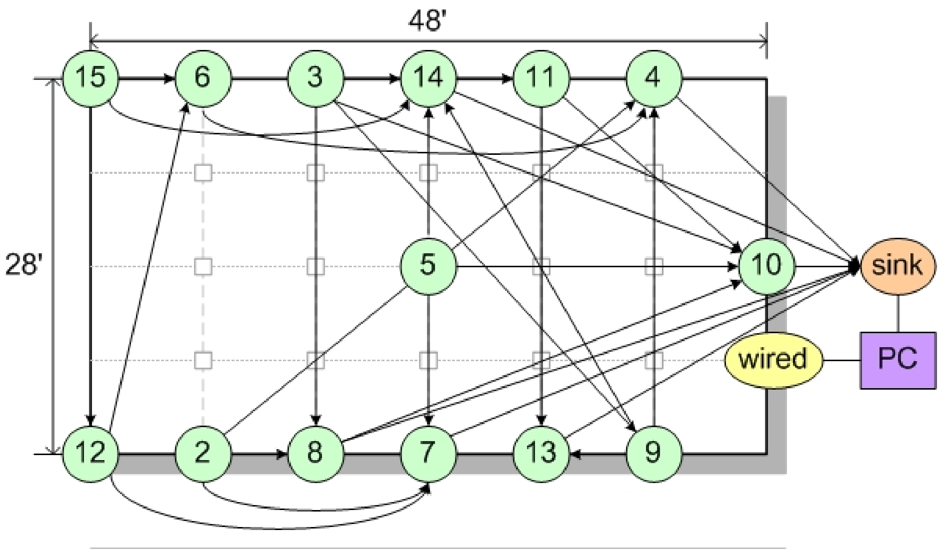
---

*“**A Wireless Sensor Network for Structural Monitoring**”*, Ning Xu, Sumit Rangwala, Krishna Chintalapudi, Deepak Ganesan, Alan Broad, Ramesh Govindan, Deborah Estrin, In Proceedings of the ACM Conference on Embedded Networked Sensor Systems, Nov.2004

Wisden Overview (Software)

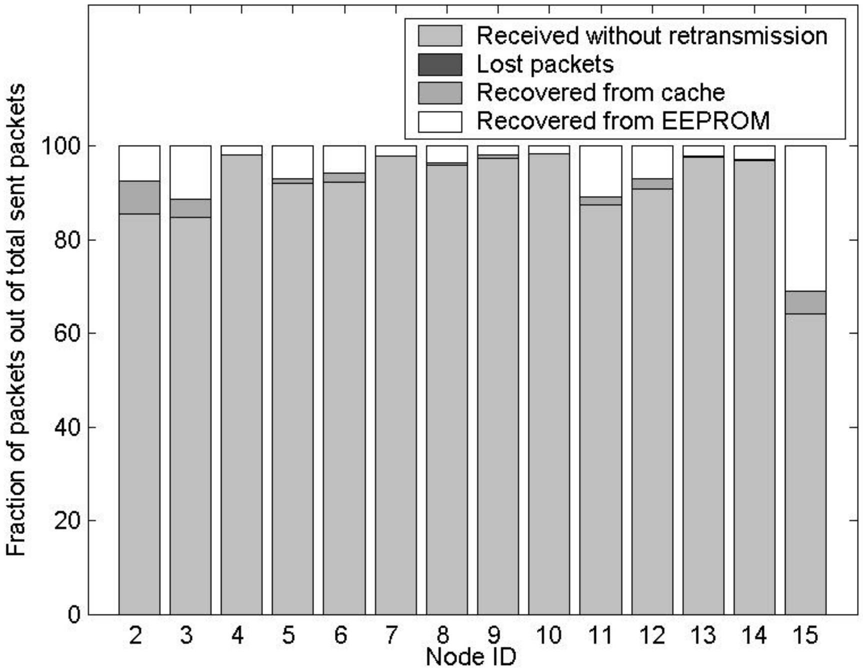
- Reliability
  - Application layer NACK mechanism
  - Hybrid hop-by-hop and end-to-end loss recovery over self-configured multi-hop tree topology
- Data Synchronization
  - Calculate residence time of a packet within each node.
  - Time-stamp data at the base station by estimating the generation time.
- Data Compression
  - Lossy run-length encoding for silence suppression
  - Required to reduce data rate and relieve the bandwidth limitations of the motes

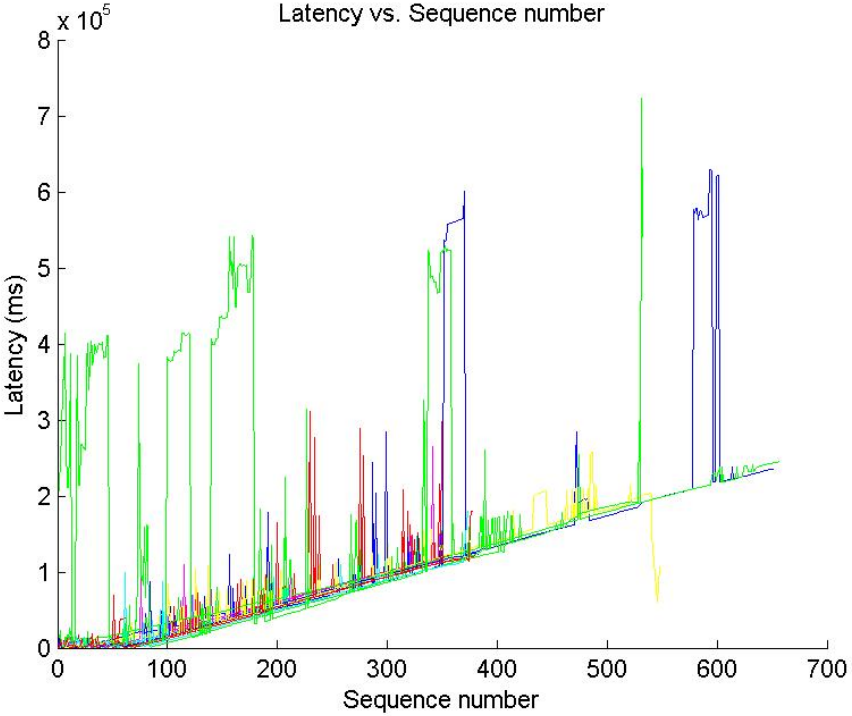
Deployment Setup



- 14 MicaZ node network
  - 2~4 hop: multi-hop network
  - 200Hz, single-axis sampling
- 5 minute experiment with 40 seconds of forced vibration

System Evaluation





- Achieved 100% delivery
  - With 9.5% of the packets being retransmitted

Discussion

- Error Model
  - Bursty loss
  - Location dependent loss
- It’s clear we need MAC-level reliability, but how much does transport-level reliability help?
- Other metrics
  - Message latency