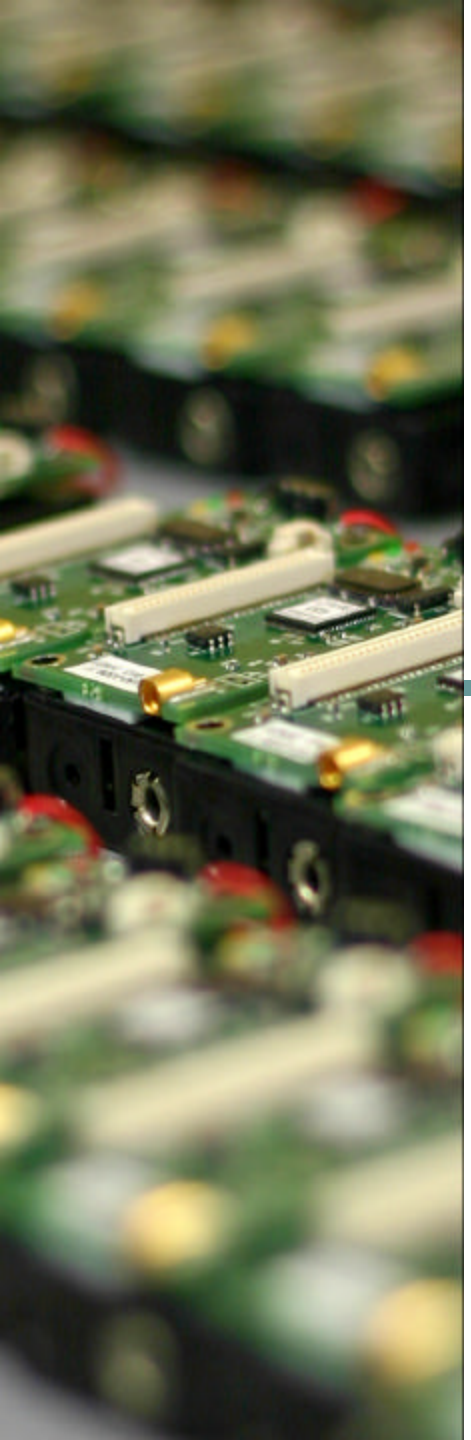


Reliable Multihop Transfer on Wireless Sensor Networks

Rodrigo Fonseca, Sukun Kim, David Culler

University of California Berkeley

IEEE SECON – October 2004

Motivation

- Some sensornet applications require 100% reliability over multiple hops
  - Structure monitoring
  - Logging (development, deployment)
  - Auditing
- This has proven to be non-trivial

Challenges

- Wireless communications
  - Low power radios
  - Asymmetric, changing links
  - Interference, etc.
- Resource constrained hardware
  - Memory, computational power, energy

Problem Scope

- Design options for achieveing high reliability over multiple hops
- Traffic pattern:
  - One destination, large data (in comparison to pkt)
  - Focus on convergence and point-to-point
- Assumption:
  - Routing layer provides a path, or set of paths to the destination

Design Options

- *Only a fraction of the transmissions goes through any given link*
- We can improve reliability by increasing
  - Number of packets injected
  - Probability of success
- Redundancy
  - Retransmissions -- End to End, Link Level, Both
  - Erasure coding
- Probability of Success
  - Path selection, Alternate paths
  - Congestion Control

Outline

- Introduction
- Alternatives for Reliability
  - Retransmission
  - Erasure Coding
  - Alternate Routes
- Experimental Results
- Conclusions

End-to-End Retransmission

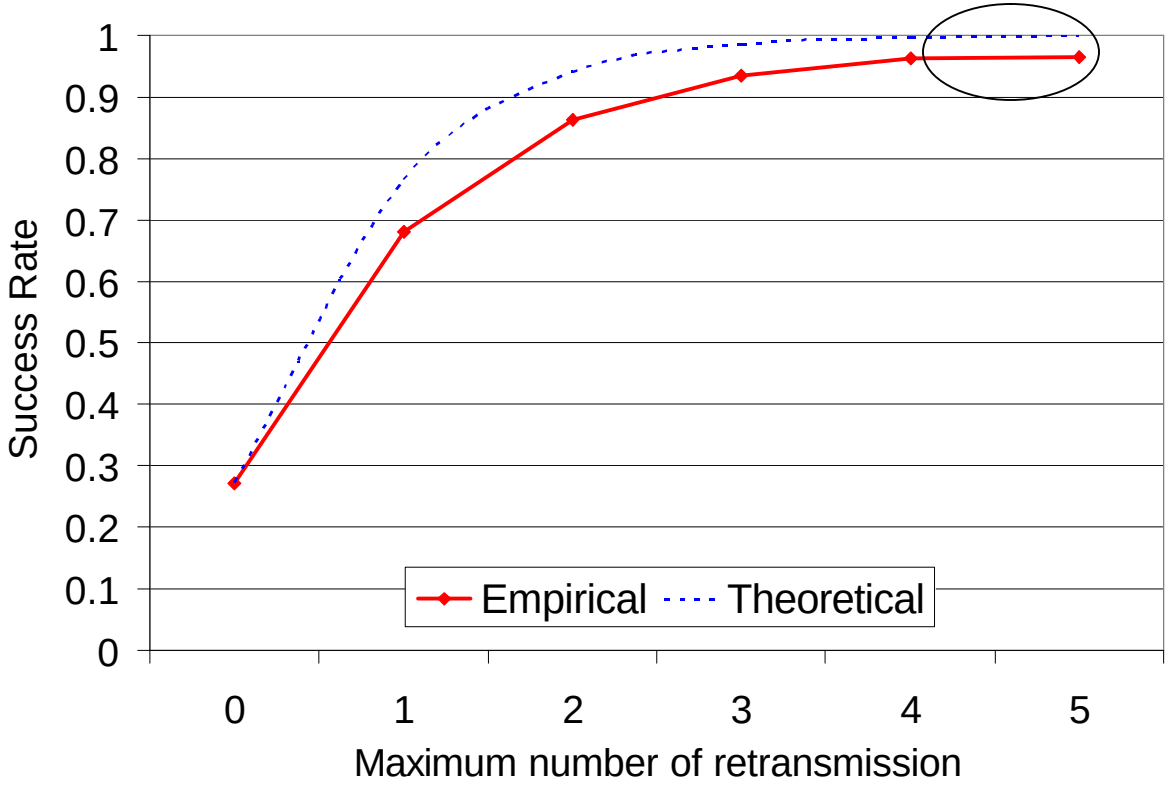
- Probability of success over multiple hops decreases rapidly
- Wasted effort
  - Fail at hop n, n-1 wasted transmissions
- E2E path may not exist at all times
- Reverse path may not exist
  - Although for 100% reliability source must receive some signal from destination

Link Level Retransmission

- Found to be very efficient in increasing reliability
  - Effect of boosting each link success probability
  - Local repair

Link Level Retransmission

- Testbed experiment, 5 hops, avg link quality 77%



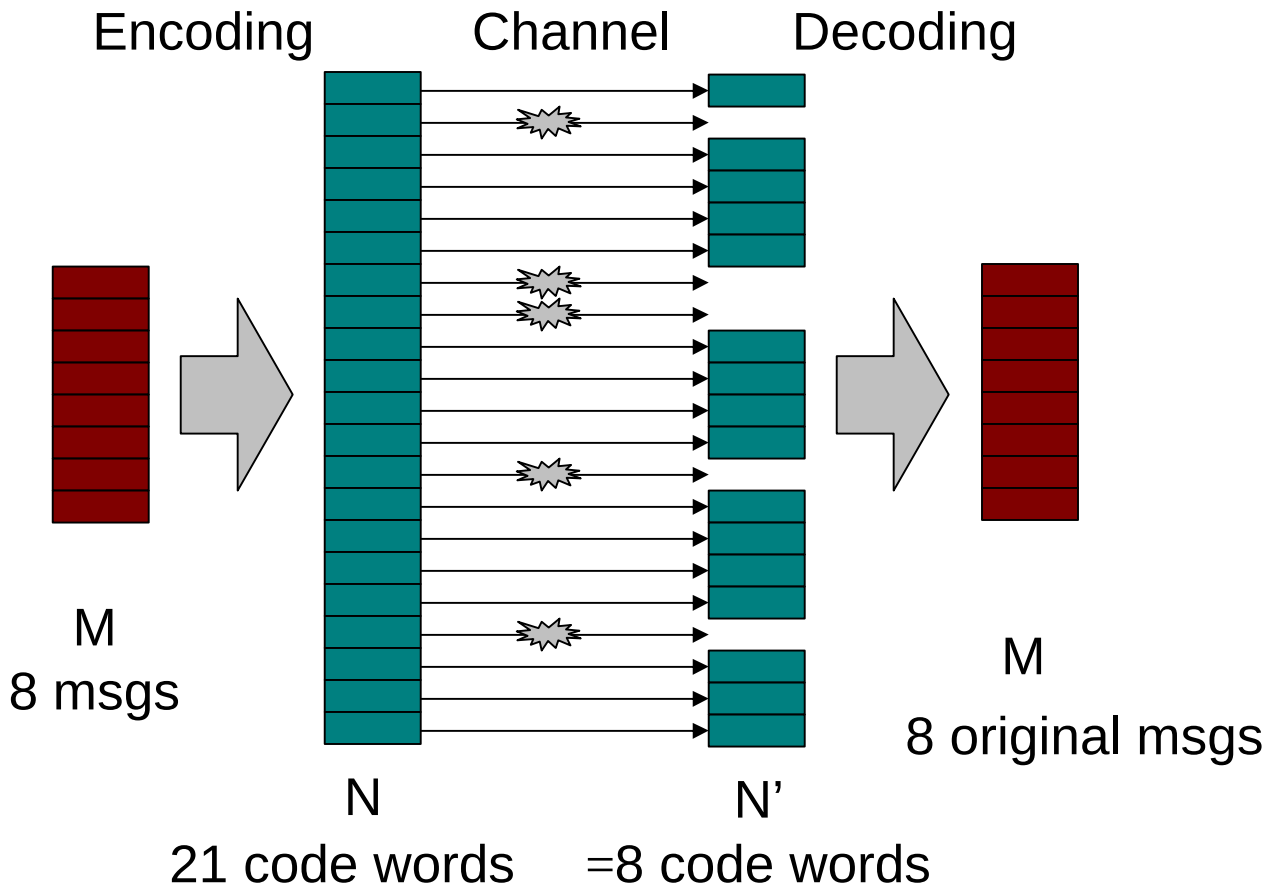
Link Level Retransmission

- Found to be very efficient in increasing reliability
  - Effect of boosting each link success probability
  - Local repair
- However, still fails to reach 100% reliability
  - Bursty loss pattern
- Cost of achieving even higher reliability may become very high

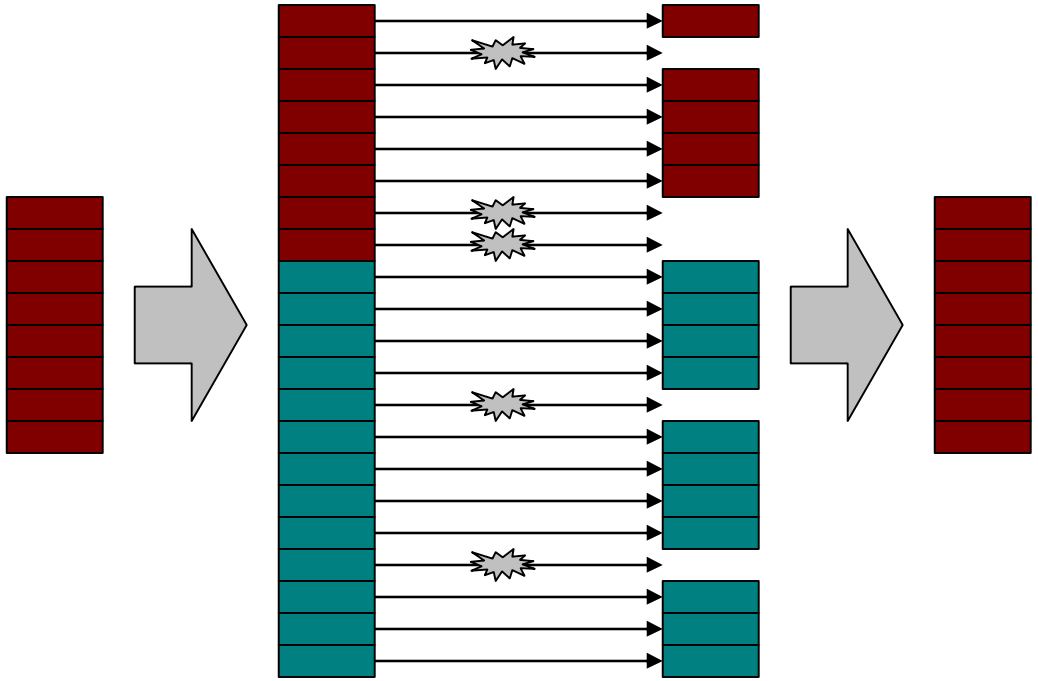
Erasure Codes

- What if we can tolerate the loss of a few percent of the packets?
- Transmit redundant information
- Erasure codes allow *M out of M+N* packets to be recovered
- Fraction of redundancy is called rate of code
  - There are rateless codes, which can produce unlimited redundancy, but may be expensive

Erasure Codes



Systematic Codes



Benefit: if receiver has codes containing original messages

- Encoding, Decoding are faster
- Even if receiver get less than 8 packets, we don’t lose every message

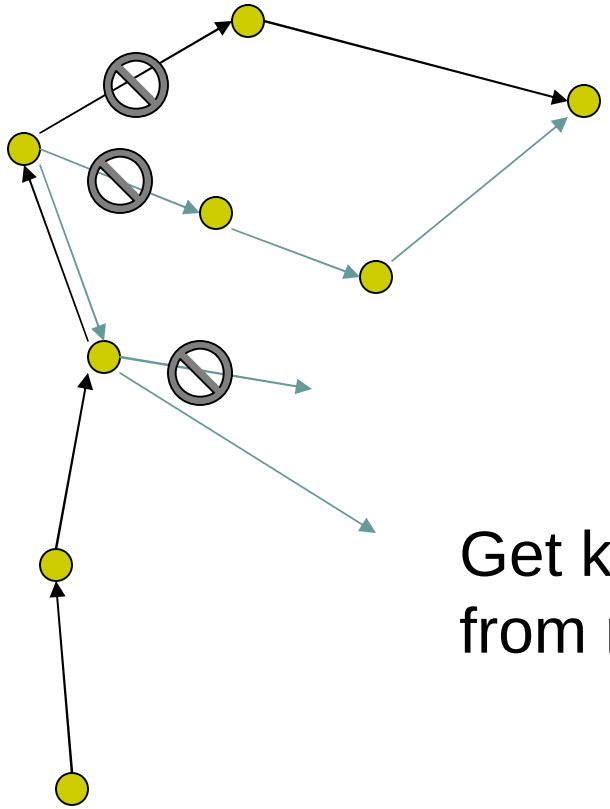
Implementation on TinyOS

- We use systematic codes
- No memory overhead for encoding or decoding
  - Codewords generated on the fly
  - Reception can stop once M pkts received
- Real time operation on Mica2 motes
- Available for TinyOS

But Losses are Bursty...

- If we loose more than N-M packets, can’t recover the entire data
- Codes introduce a fixed redundancy overhead
- So, depending on the loss process
  - Waste bandwidth on all packets
  - Not effective when needed...

Alternative Routes



Find Alternative Route: a form of ‘spatial retransmission’

But this may get tricky if we get a lot of failures

Get k best candidates for the next hop from routing layer, and try from the best

Alternative Routes

- Dynamic alternative route selection
  - Provides immediate reaction to failed route
- We change the routing layer to provide possible next hops, instead of one
- Successively try alternatives
- May still drop packet if no possible route works

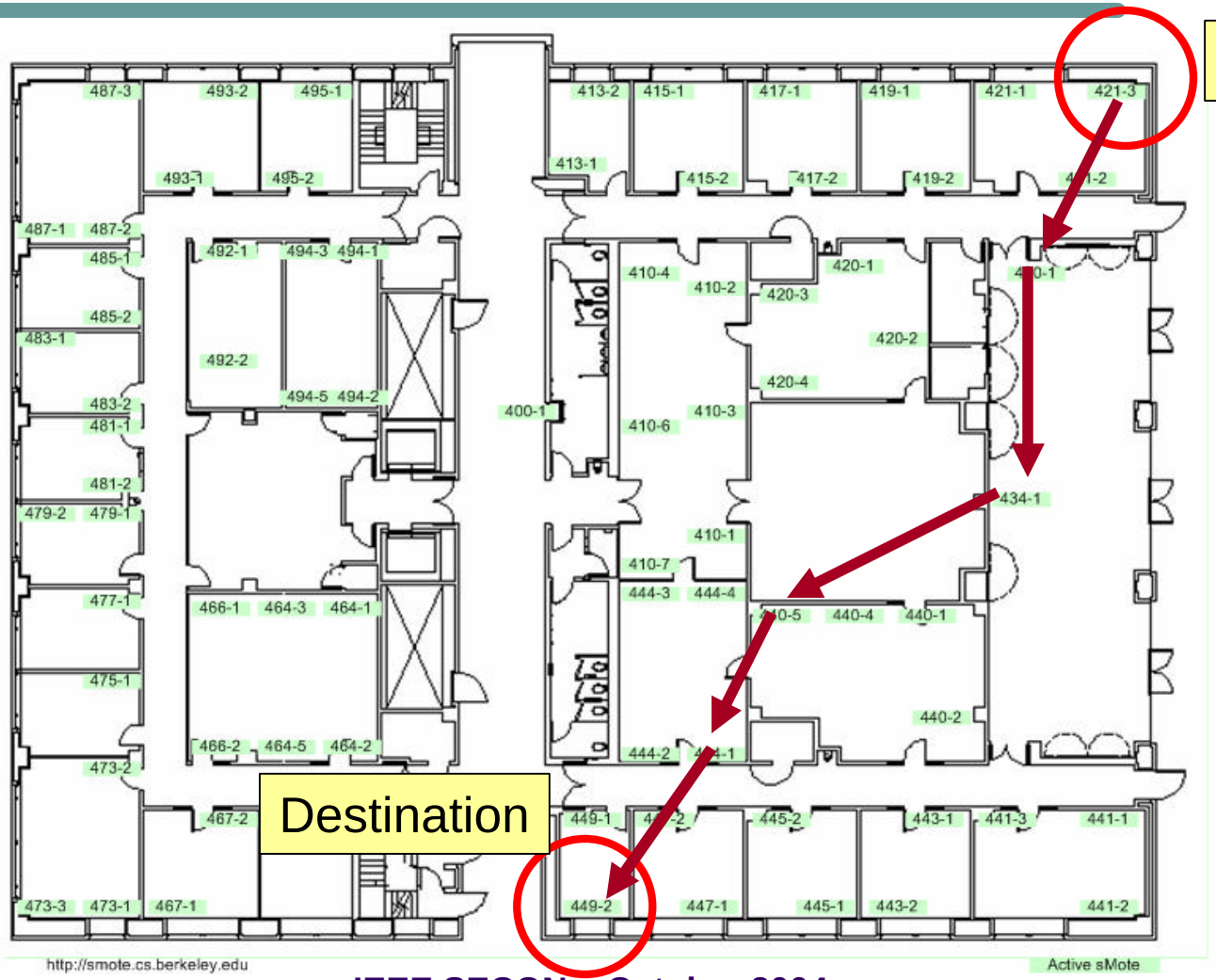
Outline

- Introduction
- Alternatives for Reliability
  - Retransmission
  - Erasure Coding
  - Alternate Routes
- Experimental Results
- Conclusions

Experimental Setup

- Point-to-point routing
  - Beacon Vector Routing used to provide routes, remained stable
  - Soda Hall Testbed, 78 motes
  - 1 pair of nodes at a time, 300 packets @ 1/s
- Results shown:
  - 1 pair of nodes, 300 packets @ 1/s
  - Average route 5 hops
  - Other pairs similar results

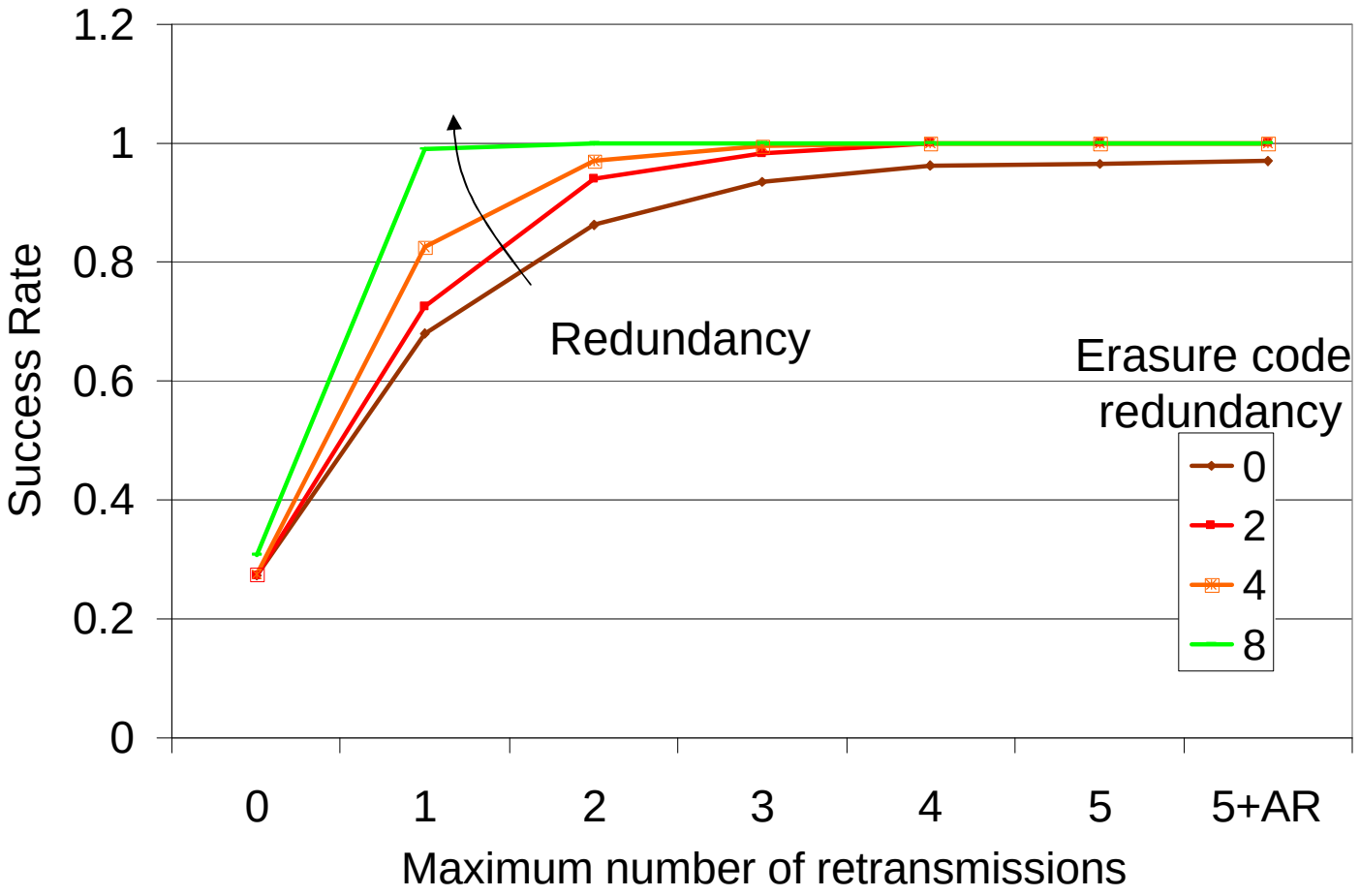
Testbed


Source
Destination
http://smote.cs.berkeley.edu
Active sMote

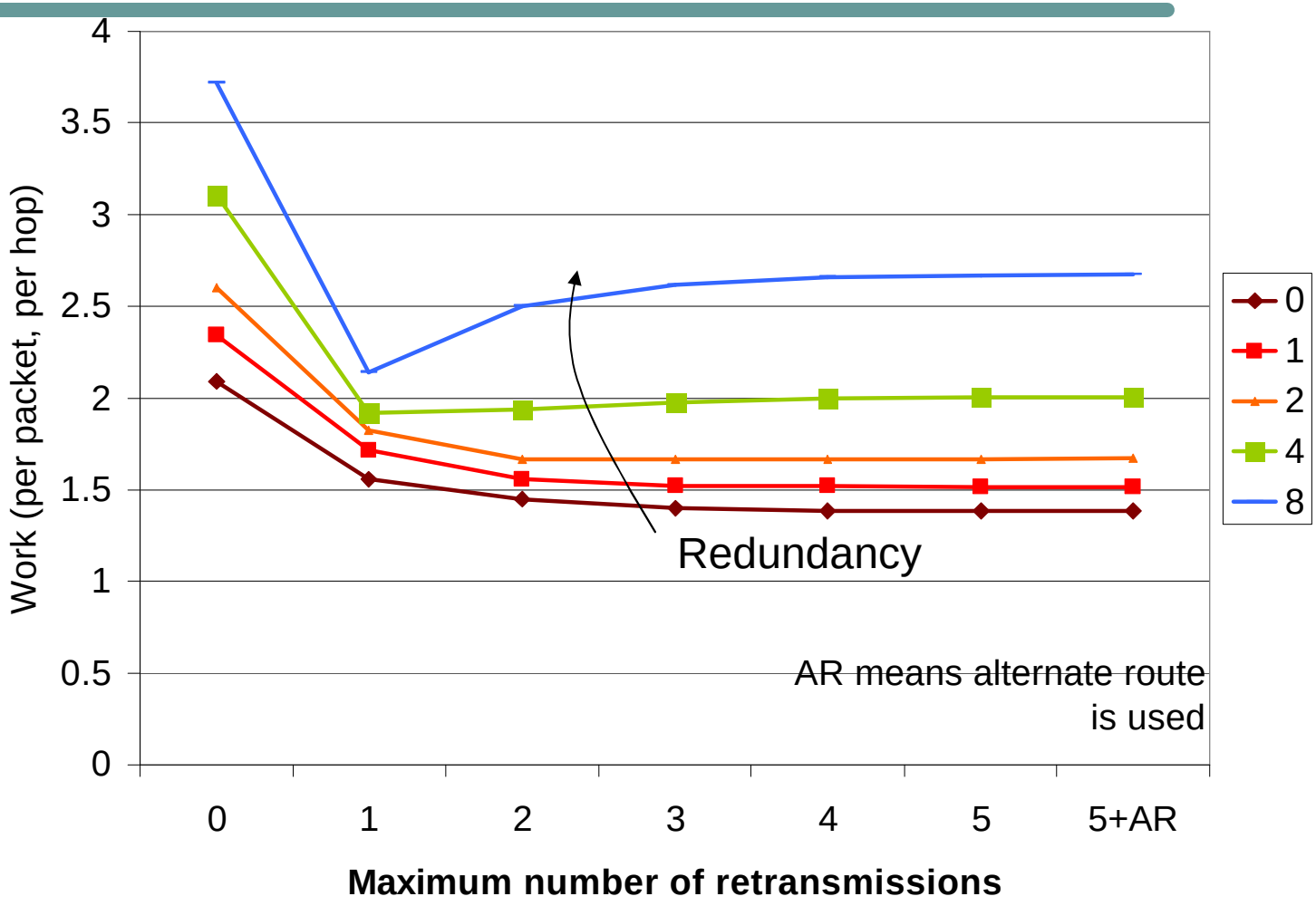
Metrics

- We are mainly concerned with two metrics:
  - Reliability
    - Fraction of application data packets that are received by the destination
  - ‘Work’
    - Number of transmissions per successfully received packet, per hop
    - Ideally, 1 transmission per hop per message

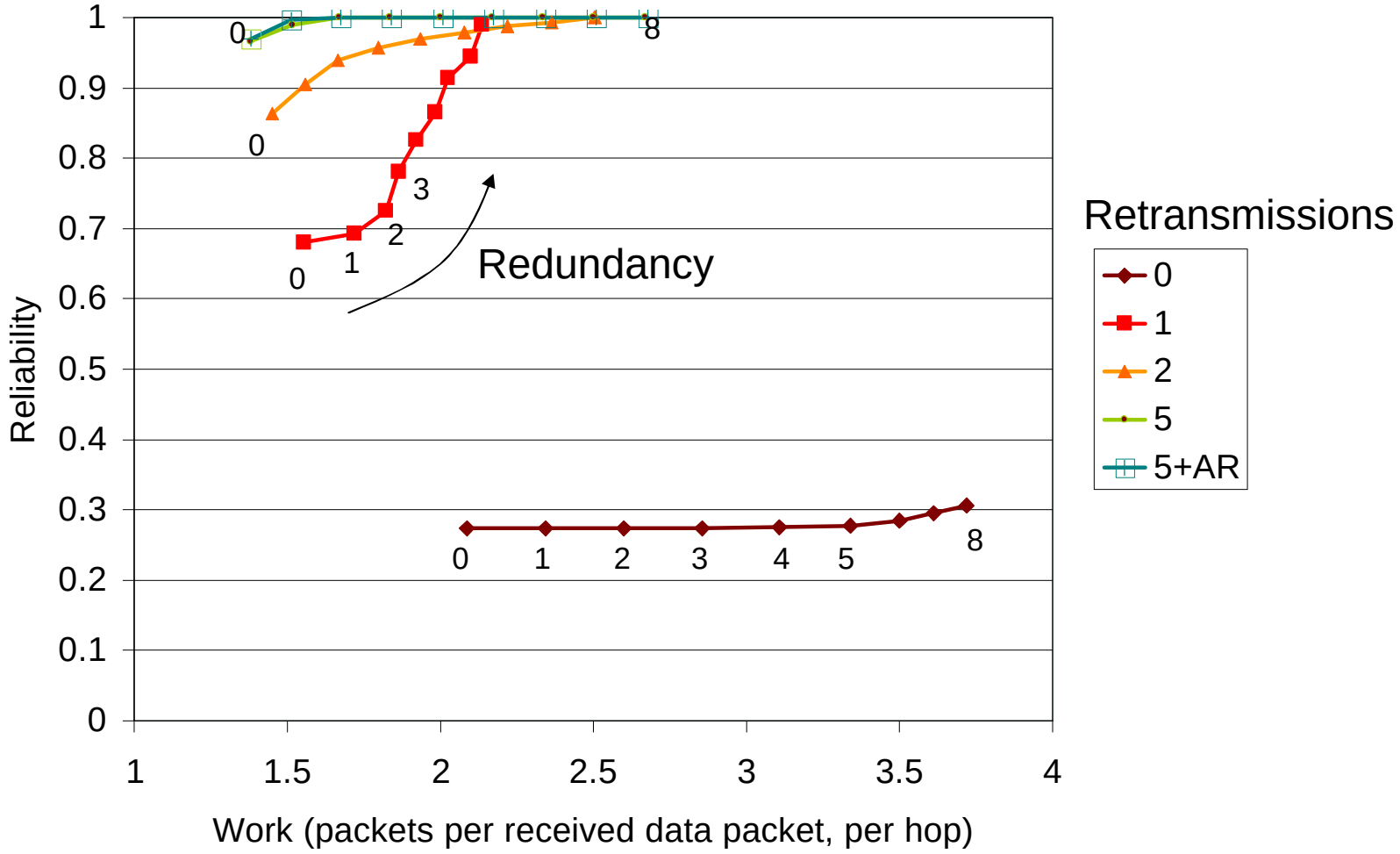
Reliability



Work per packet, per hop



Reliability versus work



Finding the best combination

Given a threshold reliability requirement, what is the retransmission /redundancy combination that has the smallest overhead?

Threshold	Retransmission	Redundancy	Work
90%	5+RF	0	1.381
95%	5+RF	0	1.381
98%	5+RF	1	1.512
99%	5+RF	1	1.512
99.9%	4	2	1.663

Conclusions and Future Work

- Main Contributions
  - We evaluated different combinations of options for multihop reliability
  - Implementation of real time erasure coding on TinyOS
- Combination of options yields best results
  - Erasure coding allows packet drops
  - Alternate route makes the loss process more amenable to erasure coding
  - Important for routing layer to quickly detect and route around failure
- Future work
  - Throrough characterization of loss patterns in other settings
  - Experiments with different routing algorithms