FlowCode: Multi-Site Data Exchange for Wireless Ad-Hoc Networks using Network Coding

H.T. Kung, Chit-Kwan Lin, Tsung-Han Lin
Stephen J. Tarsa and Dario Vlah

Harvard University
School of Engineering and Applied Sciences
Motivation

- This work was motivated by some questions raised at MILCOM 2008:

  1. What are the gains of network coding?
  2. How do we achieve these gains in a real system?
  3. How big are the gains?

- We focus on these questions in a wireless ad-hoc network context, where there are some interesting opportunities
Agenda

• Categories of Network Coding Gain
• The FlowCode System
• Simulations
• Field Experiments
• Future Work
• Concluding Remarks
Example Wireless Mesh Scenario

- S transmits data to D
- $k$ redundant paths in the middle
  - Each independent, with loss rate $p$
Opportunity: Path Redundancy

- $k$ redundant paths provide $k$ chances to successfully transmit a packet
  - Packet delivery failure = $p^k$
  - Contrast with a single path, as in routing
Opportunity: Path Diversity

- What if each of the $k$ paths could deliver a different packet simultaneously?
  - Potentially reduces expected no. of transmissions by factor of $k$
Enter Network Coding

- **Path Diversity** can be conveniently achieved with network coding
  - Coding inside network means it’s possible for $k$ links to deliver $k$ innovative packets to $D$

- **Path Redundancy** can be achieved naturally within network coding rubric
A Categorization of Network Coding Gain

- **Capacity Gain** (COPE)
- **Fault Tolerance Gain**
  - Path diversity
  - Path redundancy
Multi-site Data Exchange Scenario

- Sources A and B wish to exchange data
- Three middle links
  - Each independent, with loss rate \( p \)
The FlowCode System

• A system-level approach to achieving network coding gains
  – High-loss wireless scenarios
  – Data exchange (not just distribution)

• Two main design criteria:
  – Fault Tolerance:
    • Aggressively exploit multiple alternative paths (even if transient)
  – Capacity:
    • Orchestrate where flows meet (coding opportunity)
    • Orchestrate when flows meet (coding schedule)
FlowCode Mechanisms

- Layered Topology
  - Allows us to distinguish *direction* of traffic, relative to root layer
  - *Coding opportunities* exist at nodes where upstream and downstream flows meet
FlowCode Mechanisms

• **Operand-Driven Transmission**
  – Allows us to achieve an optimal schedule without a global scheduler
  – *Coding is scheduled* when upstream and downstream flows meet

<table>
<thead>
<tr>
<th>Peripheral Node Rules</th>
<th>Root Nodes Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>• On upstream arrival</td>
<td>• Wait for upstream and downstream arrivals</td>
</tr>
<tr>
<td>• code and send immediately</td>
<td></td>
</tr>
<tr>
<td>• On downstream arrival</td>
<td>• After waiting short period ( \tau ), send regardless of coding opportunity</td>
</tr>
<tr>
<td>• wait for upstream operand arrival</td>
<td></td>
</tr>
</tbody>
</table>
How FlowCode Works

• “Maximize innovativeness of each transmission but don’t starve upstream nodes by waiting too long”

• Layers and operand-driven transmission automatically provide fault tolerance
  – Innovative packets flow on multiple paths (diversity) in parallel (redundancy)
    • Broadcast advantage enables multiple nodes to operate independently and simultaneously
Simulation: Two-site Data Exchange

- Sources A and B
- Vary $k$, the number of lossy middle links ($p = 0.8$)
- Compare:
  - FlowCode vs. Random Uncoded Protocol
- Measure:
  - Fault tolerance gain
    - Max no. of source transmissions
  - Overall gain
    - Total transmissions in largest collision domain
      (approx. of best achievable completion time under CSMA)
Simulation: Fault Tolerance Gain

Network coding achieves optimum with 5 middle links, whereas uncoded can’t, even with 20.
Simulation: Overall Gain

Total Transmissions in the Largest Collision Domain

Number of Middle Links

Uncoded
Network Coding

Enough paths exist for minimal transmission at k=4; add’l links increase channel contention
Simulation: Breakdown of Gains

Fault Tolerance Gain
- Fault tolerance gain is higher in high loss scenarios and generally plays a more prominent role in overall gain.

Overall Gain
- Capacity gain is low under high loss because of asymmetric flows.

Capacity Gain

Fault Tolerance Gain

Overall Gain

Capacity Gain
Field Experiments

- Deployed FlowCode on **outdoor testbed**
  - 12 Nodes
  - Mobile Internet Devices (MIDs)
    - Intel Atom 800MHz
    - 512Mb RAM
    - 4Gb flash disk
    - Marvell SD8686 SDIO 802.11b/g
    - Linux
- Wi-Fi
  - Ad-hoc mode
  - 1Mbps modulation
Topology Construction

- Topology must consistently produce:
  - Wide range of faulty conditions
  - Highly dynamic link loss/outage

- **Problem:** Tuning is too hard!

- **Solution:** Exploit radio ground effects
  - Place nodes on ground, 35ft apart
    - Can’t hear each other
  - Induce communication by having human subjects (grad students) walk near nodes
    - Controlled tours
    - Irregular surface of human body reflects signal randomly
Topology Construction
Calibration: PDR as a Function of Human Location

- 0% delivery rate baseline when no human is present

- Calibration tells us where we can stand to induce reflections between transmitter and receiver

- We can induce flaky links in a controlled way
Experimental Setup: 1-Path Scenario

• 4 data sources (4-way exchange)
• Blue arrows show human tours
• FlowCode vs. BitTorrent-like Uncoded Protocol
2-Path and 3-Path Scenarios
Field Experiment Performance

Coded shows a 4X gain over uncoded due to capacity and fault tolerance gains.
Future Work

• How to do layer assignment?
  – Link quality-based methods won’t capture short-lived links that we want to exploit
  – Use geographic information (GPS or via range-based localization), which is more stable

• How to do dynamic layer adaptation?
  – Need to accommodate mobile nodes
  – By keeping a history of heard upstream/downstream operands, a node can decide its optimal layer assignment by dynamically choosing the one that maximizes throughput
Recap

• Network coding gain is comprised of
  – Capacity gain
  – Fault tolerance gain

• FlowCode is a system that achieves these gains via layered topology and operand-driven transmission mechanisms

• Fault tolerance gain is larger than capacity gain, in high-loss situations
Questions?
Experiment Topology/Link Qualities