Network Coding in ICNs
IRTF, Nov 2013

M. Gerla, YT Yu, J. Joy
CS Dept, UCLA
## Named Content Networks (NCN)

| Unique named segmented objects | • Object/segment is identified by names  
|                               | • e.g. /spiderman3/1, /spiderman3/2, ...
| In-network caching            | • +Resistance to mobility  
|                               | • +Scalability
| Pull-based transport          | • Data consumer pulls data by expressing interests  
|                               | • + Good for consumer mobility
| Name-based forwarding         | • +Possibility of multipath/multisource retrievals  
|                               | • +Exploit wireless broadcast nature
Content-Centric Networks (CCN)

• Replace IP layer
• Interest forwarding: broadcast (flood) on an interface
• Data forwarding: breadcrumb routing
Named content retrieval in MANETs

• Data discovery
• Interest propagation
• Data forwarding

• The tradeoff between
  – Robustness
  – Efficiency
Data discovery and interest propagation
Data retrieval with Network Coding

Content is split into three blocks (fragments)

Interest/Query triggers download of content

Linear Combination with Random Coefficients

Wait for any three linearly independent blocks to reconstruct original file

Three Coding options

• **Unrestricted coding**
  – All nodes encode available blocks.
  – Highest block diversity but vulnerable to pollution attack.

• **Source only coding**
  – Only the source of the file (publisher) may generate new encoded blocks. Intermediate nodes only cache and forward encoded blocks.
  – Low block diversity => weak robustness to channel losses

• **Cache coding**
  – Caches that have reconstructed the full files can (like the source) encode.
  – Caches (like source) sign the newly encoded blocks
Why Cache coding

• Increases code diversity

• Protected from pollution
  – Each cache node signs (no repudiation)
  – Easier than homomorphic signatures..

• Performance approaches “unrestricted coding” bound for popular files (with several caches)
Simulation Experiments

- Qualnet 6.1 simulator
- Multiple publishers disseminate files using broadcast.
- The network is intermittent due to interference, packet loss, and, in the dynamic case, mobility.
- Receiver can download from multiple caches in parallel.
  - Due to the broadcast mode, there are no retransmissions.
  - Redundancy is provided by multiple paths.
  - After timeout, the decoder discards file that cannot be decoded.
Static Scenario - Corridor Model

Single source/single destination

30% packet loss on links
Corridor scenario – Files Delivered within deadline

Files delivered within delay

- UNRESTRICTED MIXING
- FULL CACHE CODING
- SOURCE ONLY CODING
- NO CODING

Single source/single destination
30% packet loss on links
ICEMAN (Information CEntric Mobile Ad-hoc Networking)

• DARPA CBMEN Project led by SRI (2012-14)
• Hybrid ICN scheme:
  — flooding of interests (a la CCN), and;
  — epidemic dissemination of int. and content (like Huggle)
• Implemented in the Haggle framework
• Goal: deliver situation awareness content to teams of mobile agents (soldiers, tanks, etc)
  — Intermittent connectivity, heavy interference, loss
  — Mobile agents cooperatively download
  — Cache coding is applied
Network Coding for Content-Based Intermittently Connected Emergency Networks

Joshua Joy, Mario Gerla
UCLA
{jjoy,gerla}@cs.ucla.edu

Samuel Wood, James Mathewson
UC Santa Cruz
{sam,james}@suns-tech.com

Mark-Oliver Stehr
SRI International
ICEMAN Demo (cont)
DEMO Experiment

• Android Nexus S phones
• Topology: 10-node cluster
• nine subscribers and one publisher
• The publisher published six files with sizes ranging from 43.9 KB to 354 KB, staggered by 25 seconds every 90 seconds.
10 node Test bed results

<table>
<thead>
<tr>
<th>Transport</th>
<th>% Delivered</th>
<th>Delay (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCAST-CODING</td>
<td>95.57</td>
<td>19.5925</td>
</tr>
<tr>
<td>BCAST-NOCODING</td>
<td>95.43</td>
<td>35.6922</td>
</tr>
</tbody>
</table>
Conclusions

• ICN caches offer unique NC opportunities
  – Full cache blocks can be signed after coding
  – No repudiation, no pollution attacks
  – Eliminate need for expensive Homomorphic signatures (still prohibitive for mobile phones)

• Cache code performance approaches unrestricted code performance

• Future studies will focus on civilian applications, eg VANET downloading and file sharing