Routing in NDN

Lan Wang
Outline

• What’s new compared to IP routing?
• What has been done?
  • NLSR
  • Hyperbolic routing
  • Geohyperbolic routing
NDN has smart forwarding, so routing can be dumb.

- **PIT state** -> stop loops and enable measurements
- **Forwarding strategy** can choose next hop
- **Routing**
  - should produce *multiple* next hops
  - can be more *relaxed*
New Routing Semantics

Superset of IP semantics

1. Routing to names (address is a special case)
2. Multiple next hops (>= 1)
3. Not all the next hops lead to all the data.
   - Producer may not have all the data.
   - Links may fail.
   - Nodes may move.

D’s FIB: /nytimes/art  A, B, E
New Design Pattern

Any routing scheme in IP can be used in NDN, but needs to adopt NDN’s design pattern.

• **Use Interest/Data to retrieve routing information.**

• **Naming**: names facilitate network management and trust derivation.

• **Security**: routing data is signed by originator and verified by receivers.

• **Sync mechanism**: a new notion of transport to ensure multiple parties have the same information.
  • efficient way of set reconciliation
Named-data Link State Routing (NLSR)

- **Naming:** follow the hierarchy within a network
  - Router: `/<network>/<site>/<router>`: e.g., `/ndn/memphis/rtr1`
  - Data: `/<network>/NLSR/LSA/<site>/<router>/<process>/<type>/<version>`
- **Keys:** derived from the associated entity’s name
- **Routing security and trust model**

<table>
<thead>
<tr>
<th>Entity</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root key</td>
<td><code>/&lt;network&gt;/key</code></td>
</tr>
<tr>
<td>Site key</td>
<td><code>/&lt;network&gt;/&lt;site&gt;/key</code></td>
</tr>
<tr>
<td>Operator key</td>
<td><code>/&lt;network&gt;/&lt;site&gt;/&lt;operator&gt;/key</code></td>
</tr>
<tr>
<td>Router key</td>
<td><code>/&lt;network&gt;/&lt;site&gt;/&lt;router&gt;/key</code></td>
</tr>
<tr>
<td>NLSR key</td>
<td><code>/&lt;network&gt;/&lt;site&gt;/&lt;router&gt;/NLSR/key</code></td>
</tr>
<tr>
<td>Data</td>
<td><code>/&lt;network&gt;/NLSR/LSA/&lt;site&gt;/&lt;router&gt;/&lt;type&gt;/&lt;version&gt;</code></td>
</tr>
</tbody>
</table>
NLSR Development Status

• NLSR 0.1 was released on 8/25/14.
  • Supports both link state and hyperbolic routing
  • Uses ChronoSync to synchronize routing data
  • Uses a hierarchical trust model for routing within a single administrative domain (validation rules are configurable).

• Deployed on NDN testbed from 8/14 to 3/17.

• Code and doc:
  • http://named-data.net/doc/NLSR
Routing Scalability in NDN

• Already large number of content names in today’s Internet

• We want to bound the size of routing state while supporting an unbounded namespace
  • Small FIB size
  • Low number of routing updates
  • Comparable performance to shortest path routing algorithms
Greedy Geometric Routing

Observable network topology

Hidden metric space
Hyperbolic Embedding

From Boguna et al. “Sustaining the Internet with hyperbolic mapping” (2010)
Greedy Forwarding in HR

To forward a packet:

- Find the neighbor closest to the destination
- Forward the packet to that neighbor

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>{A, cost=10}, {B, cost=30}</td>
</tr>
</tbody>
</table>

Neighbor’s coordinates \((\theta, r)\)
Destination coordinates \((\theta', r')\)
\[ \Delta \theta = |\theta - \theta'| \mod \pi \]
\[ \text{distance} = \text{acosh}(\cosh r \cosh r' - \sinh r \sinh r' \cos \Delta \theta) \]
Why Hyperbolic Routing in NDN?

• In the ideal case, no FIB is needed
  • Each node only needs to know their neighbors’ coordinates

• Low communication cost
  • Few routing updates, as coordinates rarely change

• Have been shown to have low stretch in a power-law topology.

• NDN’s smart forwarding can adapt to short-term topological changes.
Sub-Optimal Paths in HR

To forward a packet:

• Find the neighbor closest to the destination
• Forward the packet to that neighbor

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>{B, cost=20}, {A, cost=10}</td>
</tr>
</tbody>
</table>
HR with Best-Route Strategy

- Best Route Strategy simply uses the next hop ranked highest by the routing protocol
- Delay stretch – Packet delay ratio of RTT in HR over RTT in shortest path routing
Adaptive SRTT-Based Forwarding

• Consider Round Trip Time (RTT) when choosing next hop in HR
  • Use Smoothed RTT (SRTT) to allow variation

• Best SRTT-Based Forwarding
  • Choose next hop for each FIB entry based on SRTT

• Probabilistic SRTT-Based Probing
  • Periodically probe unused next hops to learn RTT
  • Next hops that performed well previously have higher probability
HR with ASF Strategy

Deployed on the NDN testbed since March 2017.
HR is not good enough.

• HR has less than ideal stretch for testbed topology

• Geographic routing (GEO) works well for small topologies, but degrades quickly for larger topologies.

• Geohyperbolic routing: hybrid of HR and GEO
Geohyperbolic (GH) Routing and Addressing

• 3-dimensional coordinates
  • Use latitude and longitude as angular coordinates
  • Add a radial coordinate to a node’s address that captures how “central” a node is in the network

• Use geohyperbolic distance to establish network links.

\[ r_i \sim \log(\text{centrality}_i) + \text{Hyperbolic Distance} \]
GH is still not good enough.

- Success ratio of GH is almost 1 for any network size and under severe connectivity failures.
- However, suboptimal delay-wise performance is observed.

Bad delay performance example: packet forwarding from Berlin to New York via “super central” node in Shanghai.
Regionalized Geohyperbolic Routing (RGH)

- Small tweak of GH: place multiple “local hubs” within large geographic regions to “attract” packets from peripheral local nodes.
Validation – Success Ratio

(a) Success Ratio (Full Net.)

Number of Nodes ($\times 10^3$)

- Geographic
- Geohyperbolic
- Reg. Geohyperbolic
- Randomized RGH

(b) Success Ratio (Link Removal)

Number of Nodes ($\times 10^3$)
Validation – Path Stretch

![Graphs showing validation results for Path Stretch with ODS 50th and 95th percentiles for full network and link removal cases. The graphs display trends for geographic, geohyperbolic, regular geohyperbolic, and randomized RGH models.](image)
References

• NDN Routing
  • NLSR: intra-domain routing (ACM ICN 2013 and NDN TR-0037)
  • Hyperbolic Routing: inter-domain routing (IEEE IWQoS 2016 and NDN TR-0042)
  • Geohyperbolic Routing: combining geolocation and hyperbolic information in routing (SIGCOMM CCR July 2017)

• Forwarding strategy
  • Adaptive SRTT-based Forwarding (ASF): supporting hyperbolic routing (NDN TR-0042)

• Mini-NDN: lightweight NDN emulator
  • github.com/named-data/mini-ndn