Bloom Filter based Inter-domain Name Resolution: A Feasibility Study

Konstantinos V. Katsaros, Wei Koong Chai and George Pavlou
University College London, UK
Outline

- Inter-domain name resolution in ICN
- Scalability concerns
- Bloom filter based name resolution
- Evaluation framework
- Results
- Conclusions and Future Work
Inter-domain name resolution in ICN

• Name resolution: taking forwarding decisions based on names

• Inter-domain level ➞ Enormous size of the namespace
  – More than a trillion \(10^{12}\) unique web pages (Google)
  – More than 50 billion \(10^9\) IoT devices expected (Cisco)
  – Other estimations for \(10^{16}\) Information Objects (IOs)
  – Exact size subject to naming granularity i.e., hierarchical vs. flat

• Concerns about scalability
  – Memory: maintain state in RAM for low latency
  – Processing: lookup overheads
  – Bandwidth: propagate state
Inter-domain name resolution in ICN

Lookup-by-name approaches

- Distributed directory service
  - Looking up forwarding / location information

- Usually based on Distributed Hash Tables (DHTs)
  ✓ Perfect load balancing
  ✗ Stretched name resolution paths
  ✗ Routing policy violations
  ✗ Limited control over state placement

Inter-domain name resolution in ICN

*Route-by-name* approaches

- Name resolution state leads to content
- State replicated across the inter-domain topology following BGP routing
- Resolution paths follow the structure of the inter-domain topology

---

**Donna (2007)**

**Curling (2011)**
Inter-domain name resolution in ICN

Route-by-name approaches

✗ State heavily replicated (DONA: x1702.64, CURLING: x27.34)
✗ 420 TB of state for $10^{13}$ IOs at Tier-1 in DONA
✗ Highly skewed distribution of load across tiers

USING BLOOM FILTERS

• Hong et al. Bloom Filter-based Flat Name Resolution System for ICN. Internet-Draft draft-hong-icnrg-bloomfilterbased-name-resolution-03.txt, IETF Secretariat, Mar. 2015.

Bloom Filters (BF)

- Array of $m$ bits
- $k$ hash functions hash an element to one of the $m$ positions
- **ADD**: hash element $\Rightarrow$ get $k$ positions $\Rightarrow$ set to 1
- **QUERY**: hash element $\Rightarrow$ get $k$ positions $\Rightarrow$ check if all set to 1
- **UNION**: bitwise OR
- False positive ratio ($R$): $R \approx (1 - e^{-kn/m})^k$
- Optimal number of hash functions: $k = \frac{m}{n} \ln 2$

*For $R$ upper limit ($R_{\text{max}}$) and optimal $k$:*

\[
\frac{m}{n} = -\frac{\ln R_{\text{max}}}{(\ln 2)^2}
\]

*For a given $R_{\text{max}}$ and $m$ we can calculate the capacity a BF ($C_{BF}$)*
Using Bloom Filters for Name Resolution

CURLING-BF

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>CP4.1</td>
</tr>
<tr>
<td>y</td>
<td>CP4.2</td>
</tr>
<tr>
<td>z</td>
<td>CP4.3</td>
</tr>
</tbody>
</table>

Registered: x, y, z

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>CP5.1</td>
</tr>
<tr>
<td>b</td>
<td>CP5.2</td>
</tr>
</tbody>
</table>

Registered: a, b

<table>
<thead>
<tr>
<th>Name</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>CP6.1</td>
</tr>
<tr>
<td>d</td>
<td>CP6.2</td>
</tr>
<tr>
<td>e</td>
<td>CP6.3</td>
</tr>
</tbody>
</table>

Registered: c, d, e
Using Bloom Filters for Name Resolution

CURLING-BF

Globally fixed BF configuration

Bin-packing

Registered: x, y, z
Registered: a, b
Registered: c, d, e
Using Bloom Filters for Name Resolution

CURLING-BF

<table>
<thead>
<tr>
<th>Customer</th>
<th>BF</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>BF:3</td>
</tr>
<tr>
<td></td>
<td>01100101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Customer</th>
<th>BF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BF:2</td>
</tr>
<tr>
<td></td>
<td>01100101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Customer</th>
<th>BF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>BF:2</td>
</tr>
<tr>
<td></td>
<td>01100101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Customer</th>
<th>BF</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>BF:5</td>
</tr>
<tr>
<td></td>
<td>01100101</td>
</tr>
</tbody>
</table>

Name | Location
--- | ---
x   | CP4.1
y   | CP4.2
z   | CP4.3

Name | Location
--- | ---
a   | CP5.1
b   | CP5.2

Name | Location
--- | ---
c   | CP6.1
d   | CP6.2
e   | CP6.3

Registered: x, y, z
Registered: a, b
Registered: c, d, e
Configuring Bloom Filters for Name Resolution

• How to select $m$ and $C_{BF}$?
• Primary objective: limit false positives
• $F$: number of BFs at a node, $s$: number of registrations

$$F = \left\lfloor \frac{s}{C_{BF}} \right\rfloor$$

*Lower bound: overlooks BF table structure & assumes perfect bin-packing

• Setting an upper limit for $R$ at any node in the network ($R_{max}^{Node}$)

$$R_{max}^{Node} = 1 - (1 - R_{max})^F$$

★ Fixing $R_{max}^{Node}$ for worst case i.e., tier-1 domains…

• Multiple conforming BF configurations i.e., $<m, C_{BF}>$

$$m = -C_{BF} \frac{ln(1 - (1 - R_{max}^{Node})^F)}{(ln2)^2}$$
BF Configuration Tradeoff: metrics

- **Memory Requirements**
  
  \[ M^x = b_{NonBF} \cdot s_x + b_{BF} \cdot F \], \( x \in \{CURLING-BF, DONA-BF\} \)
  
  \[ M^x = b_{NonBF} \cdot s \], \( x \in \{CURLING, DONA\} \)
  
  \( b_{NonBF} = 42 \), \( b_{BF} = (m + \log_2 C_{BF})/8 \)

- **Processing Overheads**
  
  \[ LO^x = \alpha \cdot \log(s) + c \], \( x \in \{CURLING, DONA\} \)
  
  \[ LO^x = F \cdot P_N^F + \frac{P_P}{P_N} \cdot \sum_{i=0}^{F} iP_N^i \], \( x \in \{CURLING-BF, DONA-BF\} \)
  
  \[ p = \frac{s}{S}, P_{TP} = \frac{p}{F}, P_P = (1 - P_{TP})R_{max}^{FP} + P_{TP}, P_N = 1 - P_P \]

Resource requirements depend on the number of BFs
BF Configuration Tradeoff

\[ C_{BF} \gg s \]
- Sparse BFs \( \Rightarrow \) resource waste: memory and bandwidth

\[ C_{BF} \ll s \]
- Multitude of BFs \( \Rightarrow \) increased processing overheads
  - Increased number of bits-per-elements to support \( R_{Node}^{Node} \)

\[ C_{BF} = s \]
- Ideal, but state distribution is heavily skewed i.e., no single \( s \) value …
Empirical Observations (CAIDA trace set)

DONA/DONA-BF

- For large $C_{BF}$: e.g. $10^{13}$ IOs $\Rightarrow m = 23.96$ TB for a single BF!
  - Practical RAM limitations
  - No incentives for Stub ASes to use BFs (memory)
- For small $C_{BF}$: e.g., $10^{5}$ IOs $\Rightarrow m = 718.88$ KB $\Rightarrow 10^{8}$ BF lookups at tier-1
  - No incentives for Tier-1, Large ISPs to use BFs (processing)

(a) Tier-1 ISPs  (b) Large ISPs  (c) Small ISPs  (d) Stub ISPs
Empirical Observations (CAIDA trace set)

CURLING/CURLING-BF

- No single BF configuration can yield both lower memory and processing resource requirements for all ASes

(a) Tier-1 ISPs
(b) Large ISPs
(c) Small ISPs
(d) Stub ISPs
Empirical Observations (CAIDA trace set)

- Are there BF configurations leading to *some* incentives for all ASes?
- *Resource wastage*

\[
\begin{align*}
  w_{i,x-BF}^y &= \max\{0, 1 - \frac{y_i^x}{y_i^{x-BF}}\} \\
  W^y &= \frac{\sum_{i=1}^{N} w_{i,x-BF}^y}{N}
\end{align*}
\]

\(y \in \{M,LO\}, x \in \{DONA,CURLING\}, i \in [1, \ldots, N]\)
Empirical Observations (CAIDA trace set)

- Stub networks (vast majority of ASes)
  - Low resource wastage range: \( C_{BF} = 2^{23}, m = 50.63\text{MB} \) to \( C_{BF} = 2^{32}, m = 19\text{GB} \)
  - But substantial processing overheads for Tier-1/Large ISPs at this range

- No single BF configuration can achieve a good compromise
Simulation results (Scaled down topologies)

State size

![Graph showing state size vs. CBF percentage]
Simulation results

Lookup overheads

- Increased overheads for CURLING
  - Effect of not using peering links
- Low sensitivity to $C_{BF}$
  - Impact of topology structure (see next)
Simulation results
BF merging

- Larger overheads for top tier domains
  - Large number of Stub domains direct customers of top tier domains
  - Non-optimal merging
- Larger overheads for larger $C_{BF}$ values
  - $F$: lower bound estimation
Simulation results
Global False Positive Ratio

- Extremely high False Positive Ratio
  - Zipf-like workload i.e., multiple requests for popular items
    - Considerably lower ratio for unique requests, but still unacceptable
  - No BF update mechanism…
Simulation results
Global False Positive Ratio

- Vast majority of False Positives at tier-1 domains
  - Large concentration of Stub domains at level 2
  - BF maintained per customer, not merged
Conclusions

• No single BF configuration can lower both memory and processing requirements for all ASes
  – Reducing memory resource requirements for the majority of ASes inflates processing requirements at Tier-1

• Direct connectivity of large content provider ASes to tier-1 inflates False Positives

Future Work

• Uniform Recursive Tree (UTR) model
• Scalable BFs, Dynamic BFs, (d-left) Counting BFs, Cuckoo filters
Thank you.

Questions?

Konstantinos V. Katsaros
k.katsaros@ucl.ac.uk
BACKUP SLIDES
Evaluation Framework

• **Metrics**
  - State size
  - Processing overheads
  - Resolution delay
  - Registration overhead

• **Network topology**
  - Full-scale: CAIDA trace set
    • > 45K ASes, >150K links
    • Cone size (c): number of downstream customers
    • Tiers
      - Tier-1
      - Large ISPs (50 < c)
      - Small ISPs (5 < c < 50)
      - Stub networks (c ≤ 5)
  - Scaled down topologies

• **Workload**
  - IO names uniformly distributed across Content Providers at leaf ASes
  - Full-scale: $10^{13}$ IOs
  - Scaled down: ~$10^6$ IOs

<table>
<thead>
<tr>
<th>Traffic type</th>
<th>Data plane fraction</th>
<th>Control plane fraction</th>
<th>Object size median</th>
<th>Object size distribution</th>
<th>Object popularity distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web</td>
<td>35.10%</td>
<td>36.40%</td>
<td>10.386 KB</td>
<td>Lognormal-Pareto</td>
<td>Zipf</td>
</tr>
<tr>
<td>P2P</td>
<td>15.85%</td>
<td>2.56x10^{-4}%</td>
<td>650.11 MB</td>
<td>Sampling</td>
<td>Mandelbrot-Zipf</td>
</tr>
<tr>
<td>Video</td>
<td>19.54%</td>
<td>37.04x10^{-3}%</td>
<td>7.6 MB</td>
<td>Concatenated Normal</td>
<td>Weibull</td>
</tr>
<tr>
<td>Other</td>
<td>29.51%</td>
<td>63.57%</td>
<td>5 KB</td>
<td>Normal</td>
<td>Zipf</td>
</tr>
</tbody>
</table>

GlobeTraff traffic mix
Results
State distribution, full-scale, per tier, DONA vs. CURLING

- 420 TB of state for Tier-1 ASes in DONA, for $10^{13}$ IOs
- < 10 TB for 90% of Large ISPs (CURLING), 30% (DONA)
- Substantially less state for CURLING …

<table>
<thead>
<tr>
<th>Type</th>
<th>DONA Average</th>
<th>Median</th>
<th>CURLING Average</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier-1</td>
<td>26,250</td>
<td>26,250</td>
<td>15,723</td>
<td>16,215</td>
</tr>
<tr>
<td>Large ISP</td>
<td>9,635</td>
<td>11,206</td>
<td>724</td>
<td>79</td>
</tr>
<tr>
<td>Small ISP</td>
<td>4,095</td>
<td>26</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>Stub</td>
<td>536</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Number of 16 GB RAM servers required to hold state in RAM

Cumulative fraction

- For Tier-1: DONA > CURLING
- For Large ISPs: DONA > CURLING
- For Small ISPs: DONA > CURLING
- For Stub: DONA > CURLING

Cumulative fraction vs. State Size (GB)

Cumulative fraction vs. State Size (GB) for Tier-1, Large ISPs, Small ISPs, and Stub.
DONA/CURLING Scalability

Large impact of peering links

- Not exchanging state across **peering links** (CURLING):
  - Reduces state overheads: 62-fold on average, up to 679-fold for Stub domains
  - Reduces registration traffic: 684% on average
  - Increases processing overhead (2.78-fold on average), especially at top-most domains
  - Increases name resolution paths: 16% on average
Results

Processing overheads, scaled down, DONA vs. CURLING

- Increased overhead for CURLING (2.78-fold on average)
  - Especially for top-most ASes (3.9-fold on average)
- Not searching for contents in peering domains propagates requests further up
Results
Resolution delays, Registration overheads, DONA vs. CURLING

- Shorter name resolution paths for DONA (16% on average)
  - More resolution requests reach the higher tiers
- 684% increase of registration traffic for DONA
  - Impact of peering links
Further/detailed results

<table>
<thead>
<tr>
<th>Type</th>
<th>DONA Average</th>
<th>DONA Median</th>
<th>CURLING Average</th>
<th>CURLING Median</th>
<th>Avg. gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier-1</td>
<td>26,250</td>
<td>26,250</td>
<td>15,723</td>
<td>16,215</td>
<td></td>
</tr>
<tr>
<td>Large ISP</td>
<td>9,635</td>
<td>11,206</td>
<td>724</td>
<td>79</td>
<td></td>
</tr>
<tr>
<td>Small ISP</td>
<td>4,095</td>
<td>26</td>
<td>8</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Stub</td>
<td>536</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Number of 16 GB RAM servers required to hold state in RAM

<table>
<thead>
<tr>
<th>Type</th>
<th>DONA Average</th>
<th>DONA Median</th>
<th>CURLING Average</th>
<th>CURLING Median</th>
<th>Avg. gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Tiers</td>
<td>3.778%</td>
<td>0.003%</td>
<td>0.060%</td>
<td>0.003%</td>
<td>62.97%</td>
</tr>
<tr>
<td>Tier-1</td>
<td>100.00%</td>
<td>100.00%</td>
<td>59.895%</td>
<td>61.769%</td>
<td>1.67%</td>
</tr>
<tr>
<td>Large ISP</td>
<td>36.701%</td>
<td>42.687%</td>
<td>2.758%</td>
<td>0.298%</td>
<td>13.31%</td>
</tr>
<tr>
<td>Small ISP</td>
<td>15.599%</td>
<td>0.097%</td>
<td>0.029%</td>
<td>0.018%</td>
<td>537.90%</td>
</tr>
<tr>
<td>Stub</td>
<td>2.039%</td>
<td>0.003%</td>
<td>0.003%</td>
<td>0.003%</td>
<td>679.67%</td>
</tr>
</tbody>
</table>

State size per AS expressed as a percentage of the total state size throughout the inter-network (%)
Further/detailed results (Cont.)

Registration traffic (average/median)

<table>
<thead>
<tr>
<th></th>
<th>DONA</th>
<th>CURLING</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average</td>
<td>Median</td>
</tr>
<tr>
<td>Tier-1</td>
<td>66.13 Gbps</td>
<td>66.13 Gbps</td>
</tr>
<tr>
<td></td>
<td>39.61 Gbps</td>
<td>40.85 Gbps</td>
</tr>
<tr>
<td>Large ISPs</td>
<td>24.28 Gbps</td>
<td>28.23 Gbps</td>
</tr>
<tr>
<td></td>
<td>1.82 Gbps</td>
<td>197.08 Mbps</td>
</tr>
<tr>
<td>Small ISPs</td>
<td>10.32 Gbps</td>
<td>/ 64.15 Mbps</td>
</tr>
<tr>
<td></td>
<td>19.18 Mbps</td>
<td>11.90 Mbps</td>
</tr>
<tr>
<td>Stub networks</td>
<td>1.35 Gbps</td>
<td>1.98 Mbps</td>
</tr>
<tr>
<td></td>
<td>1.98 Mbps</td>
<td>1.98 Mbps</td>
</tr>
</tbody>
</table>

10^{13} IOs, two weeks average IO lifetime, REGISTRATION size = 1 KB.