A Fine-Grained Distance Metric for Analyzing Internet Topology

Mark Crovella
joint with
Gonca Gursun, Natali Ruchansky, and Evimaria Terzi
Distance in Graphs with Low Diameter

- Do ‘neighborhoods’ still exist?
- Under what distance metric?
A Motivating Problem

- Internet routing between domains
  - Autonomous Systems (ASes)
- Simple question: what paths pass through my network?
- Important for network planning, traffic management, security
  - If someone at BU were to send an email to UM, would it go through my network?
Surprisingly hard to answer!

- Routing consists of an AS choosing a next hop for each destination
  - Destinations are ‘prefixes’
- Decisions are only partially communicated to neighbors
  - In general, decisions made by a remote AS are not known
Observing Traffic

- An AS can observe the traffic passing through it
  - If BU sends traffic to UM through Sprint, Sprint knows it
- Traffic only provides positive information
  - Absence of traffic is ambiguous

- If the observer does not see traffic from i to j, it is either
  - A true zero: the path from i to j does not go through the observer; or
  - A false zero: the path goes through, but i is not sending anything to j
The Visibility-Inference Problem

- For each observer there is a ground truth matrix $T$
  - $T(i,j) = 1 \implies$ path from $i$ to $j$ passes through observer

- Traffic summarized in observable matrix $M$
  - $M(i,j) = 1 \implies$ traffic was seen flowing from $i$ to $j$
  - $M(i,j) = 1 \implies T(i,j) = 1$

- Problem: label the zeros in $M$ as either true or false

\[
T = \begin{bmatrix}
0 & 1 & 0 \\
1 & 1 & 0 \\
1 & 0 & 1
\end{bmatrix} \\
M = \begin{bmatrix}
0 & 0 & 0 \\
1 & 0 & 0 \\
1 & 0 & 1
\end{bmatrix}
\]

- Success metric: Detection Rate, False Alarm Rate
  - Of true zeros
Intuition

- Amplify knowledge obtained from traffic observation
- Empirically we observe that there are groups of sources, destinations exhibiting ‘similar routing’
- Observed traffic provides positive knowledge for entire group

\[
M = \begin{bmatrix}
s_1 & d_1 & d_2 \\
0 & 0 & 0 \\
1 & 0 & 0 \\
1 & 0 & 1
\end{bmatrix}
\]

- Requires a metric that captures the notion that \(d_1\) is ‘close’ to \(d_2\) while \(s_1\) is ‘far’ from \(s_2\)
Seeking a metric for ‘neighborhoods’

- Typical distance used in graphs is hop count
- Not suitable in small worlds

- 90% of prefix pairs have hop distance < 5
  - Clearly, typical distance metric is inappropriate
- Need a metric that expresses ‘routed similarly in the Internet’
  - or other graph
Capturing global routing state

- Conceptually, imagine capturing the entire routing state of the Internet in a matrix $N$
- $N(i,j) =$ next hop on path from $i$ to $j$
- Each row is actually the routing table of a single AS
- Now consider the columns

$$N = \begin{bmatrix} \text{green} & \text{blue} \\ \text{blue} & \text{green} \end{bmatrix}$$

[Diagram of network]
Routing State Distance

- \( \text{rsd}(a,b) = \# \text{ of entries that differ in columns } a \text{ and } b \text{ of } N \)
- If \( \text{rsd} \) is small, most ASes think the two prefixes are ‘in the same direction’
- A metric (obeys triangle inequality)
RSD in Practice

- Key observation: we don’t need all of $H$ to obtain a useful metric
  - That’s good – the problem would be solved anyway if we had that
- Many (most?) nodes contribute little information to RSD
  - Nodes at edges of network have nearly-constant rows in $H$
- Sufficient to work with a small set of well-chosen rows of $H$
  - Empirically we observe this to be the case
- Such a set is obtainable from publicly available BGP measurements
  - Note that public BGP measurements require some careful handling to use properly for computing RSD
Using RSD to amplify observed traffic

- For each zero \((i,j)\) in \(M\)
  - Find \(S_i = \{i' | rsd(i, i') < \tau\}\), \(D_i = \{j' | rsd(j, j') < \tau\}\)
  - Compute \(\pi(i, j) = \sum M(S_i, D_j)\)
  - If \(\pi(i, j) > \beta\) then \((i,j)\) is a false zero; o/w true zero
- Thresholds \(\tau\) and \(\beta\) are easy to set in an automatic way
- Applied to three sets of ASes:
  - Core-100 and Core-1000: centrally located ASes
  - Edge-1000: randomly chosen ASes, mostly near edge of net
Experimental setup

• **Ground-truth matrices from BGP data**
  – Collected all active paths from 38 sources to 135,000 destinations
  – For every AS, construct $38 \times 135,000$ ground truth matrix $T$
  – Data hygiene: discussed at end

• **Simulate traffic absence by setting some 1s to zeros**
  – Flipped at random from 1 to 0
  – 10%, 30%, 50%, 95%
Quick Look
Performance

- Accuracy on Edge-1000 is poorer
  - At edge of network, problem is actually easier
  - RSD not the right idea to use there
### Mean TPR and FPR: Core-1000

<table>
<thead>
<tr>
<th>Flip Rate</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>0.98 (0.03)</td>
<td>0.03 (0.04)</td>
</tr>
<tr>
<td>30%</td>
<td>0.98 (0.03)</td>
<td>0.03 (0.04)</td>
</tr>
<tr>
<td>50%</td>
<td>0.98 (0.03)</td>
<td>0.03 (0.05)</td>
</tr>
<tr>
<td>95%</td>
<td>0.98 (0.03)</td>
<td>0.21 (0.18)</td>
</tr>
</tbody>
</table>
Properties of RSD

On graphs of size $n$ with shortest-path routing

• Tree: $\text{rsd} = 1 + \text{length of path in tree}$
  - So $\text{rsd}$ in a tree is low, generally $O(\log n)$

• Clique: $\text{rsd} = n$

• Star: $\text{rsd} = 3$
Comparison to Classical Distance

<table>
<thead>
<tr>
<th></th>
<th>rsd</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>Star</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Balanced tree</td>
<td>log $n$</td>
<td>log $n$</td>
</tr>
<tr>
<td>Clique</td>
<td>$n$</td>
<td>1</td>
</tr>
</tbody>
</table>
RSD on small worlds

Clustering Coefficient
Hop Distance
RSD

RSD remains useful as hop distance breaks down

\[ d \]

\[ rsd \]

\[ p = 0 \]

\[ p = 0.01 \]

\[ p = 0.1 \]
RSD as a tool for data analysis and discovery

- RSD applied to Internet prefixes shows low effective rank
- Implication: RSD is good for visualization of prefixes

20 largest singular values of 1000 x 1000 RSD matrix for Internet prefixes
RSD for clustering – fine scale
Visualizing all prefixes

Smaller cluster is about 25% of all prefixes

Consistent clustering in any sample of prefixes
Understanding Clustering under RSD

In terms of the next-hop matrix $N$:
- A cluster $C$ corresponds to a set of columns of $N$, ie, $N(:,C)$
- The columns are close in RSD
- So they must be similar in some positions $S$

In terms of BGP routing:
- For any row in $N(S,C)$ the entries must be nearly the same
- So $S$ is a set of ASes making similar routing decisions w.r.t. $C$

We call such a pair $(S,C)$ a local atom
Why are Local Atoms Interesting?

- Routing decisions by ASes are made independently.
- There will often be many next hops for any given prefix.
- Different ASes will choose different next hops for a prefix.
Exposing Unexpected Coordination

A significant fraction of ASes are all following the same rule.

There is a special AS X in the system.

The rule is: “Always use AS X to get to any customer in its cone.”

The ASes making this coordinated decision are not related in any obvious way.

The smaller cluster is a local atom.
Attractivity of AS6939
What is so special about Hurricane Electric?

“What Hurricane Electric is willing to peer with networks which are connected to one or more exchange points which we have in common.”

he.net/adm/peering.html

- A peering policy based *only on colocation*
  - Settlement-free peering
  - No commercial consideration or traffic ratios

- Immensely attractive to small/medium NSPs
  - “free” access to any customer network of HE
  - HE backbone has global scope
Systematic Discovery of Local Atoms

- Manual inspection uncovers a large-scale local atom
  - The Hurricane Electric cluster
  - Includes about 25% of prefixes in the Internet
- Do smaller local atoms exist? How can we detect them?
- Need an effective clustering strategy for RSD

- A natural approach: seek clusters such that
  - Within-cluster RSD is minimized
  - Between-cluster RSD is maximized

\[
P\text{-Cost}(\mathcal{P}) = \sum_{x, x': \mathcal{P}(x) = \mathcal{P}(x')} D(x, x') + \sum_{x, x': \mathcal{P}(x) \neq \mathcal{P}(x')} (m - D(x, x'))
\]
Pivot Clustering

- Minimizing P-Cost is NP-hard, but there is an expected 3-approximation algorithm for it: *Pivot clustering*

---

**Algorithm 1 The Pivot algorithm.**

A set of prefixes $X = \{x_1, \ldots, x_n\}$ and a threshold $\tau \in [0, m]$.

A partition $\mathcal{P}$ of the prefixes

1. pick a random prefix $x \in X$
2. create a cluster $C_x = \{x' \mid D(x, x') \leq \tau\}$
3. $X = X \setminus C_x$
4. Pivot($X, \tau$)
Largest 5 clusters found using Pivot Clustering

- Clusters show clear separation
- Each cluster corresponds to a local atom
## Interpreting Clusters

<table>
<thead>
<tr>
<th></th>
<th>Number of Prefixes (C)</th>
<th>Number of Source ASes (S)</th>
<th>Destinations</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>150</td>
<td>16</td>
<td>Ukraine 83%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Czech. Rep 10%</td>
</tr>
<tr>
<td>C2</td>
<td>170</td>
<td>9</td>
<td>Romania 33%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Poland 33%</td>
</tr>
<tr>
<td>C3</td>
<td>126</td>
<td>7</td>
<td>India 93%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>US 2%</td>
</tr>
<tr>
<td>C4</td>
<td>484</td>
<td>8</td>
<td>Russia 73%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Czech rep. 10%</td>
</tr>
<tr>
<td>C5</td>
<td>375</td>
<td>15</td>
<td>US 74%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Australia 16%</td>
</tr>
</tbody>
</table>
Conclusions

• RSD is a metric that captures when two prefixes are “routed similarly in the Internet”

• It has many useful properties
  – a fine-grained distance metric even in small-world graphs
  – captures a different notion of distance than shortest-path
  – summarizes the routing state of the entire network
  – effective for visualizing Internet prefixes
  – allows in-depth analysis of BGP
  – uncovers surprising and interesting patterns in routing
    • Local atoms
Thank you!

Gonca Gursun  Natali Ruchansky  Evimaria Terzi
Related Work

- Reported that BGP tables provide an incomplete view of the AS graph.
  [Roughan et. al. ’11]

- Visualization based on AS degree and geo-location.
  [Huffaker and k. claffy ’10]

- Small scale visualization through BGPlay and bgpviz

- Clustering on the inferred AS graph.
  [Gkantsidis et. al. ’03]

- Clustering prefixes that share the same BGP paths into policy atoms.
  [Broido and k. claffy ’01]

- Methods for calculating policy atoms and characteristics.
  [Afek et. al. ’02]
Data Hygiene Implications

- BGP data is known to favor customer-provider links and miss peer-peer links
- Our restriction to 38 x 135000 known paths means that we are not missing any links in the scope of our experiments
- Hence accuracy for the chosen subsets of M is not affected by missing links
- However, the accuracy of our methods may be different on the full M
  - Whether better or worse, it’s not clear
  - There is some reason to believe it would be better...