Routing State Distance: A Path-based Metric for Network Analysis

Gonca Gürsun

joint work with

Natali Ruchansky, Evimaria Terzi, Mark Crovella

BOSTON UNIVERSITY

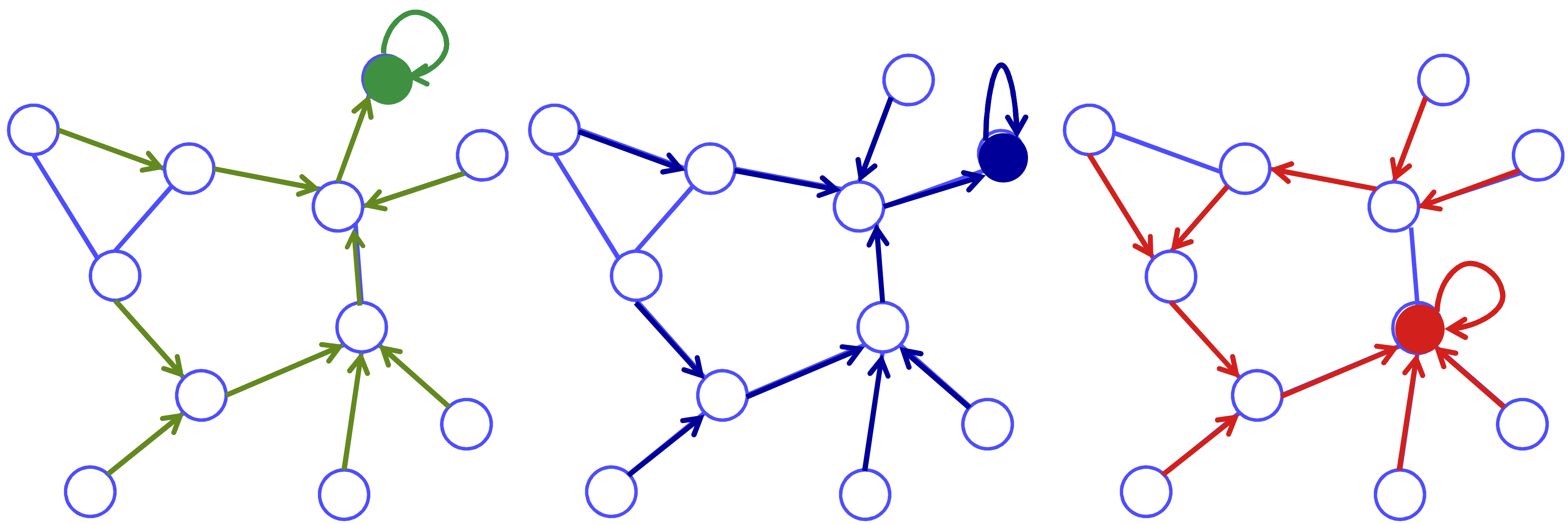
Google

NSF

Microsoft Research

YAHOO!

Distance Metrics for Analyzing Routing



~~Shortest~~ Path

Similar Routing

A New Metric

A **new** metric **path- based** metric that can use used for:

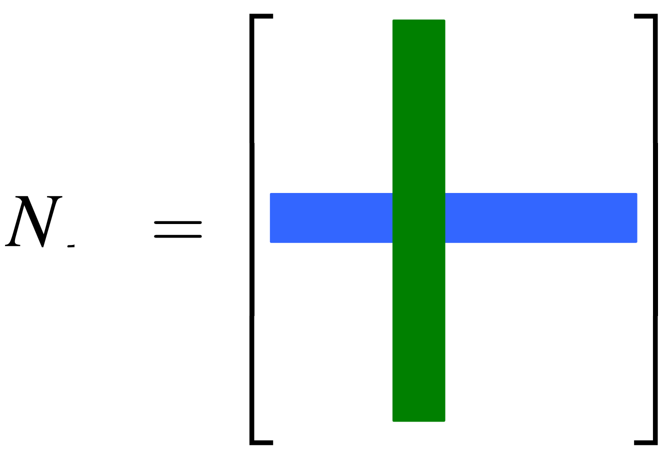
- **Visualization** of networks and routes
- **Characterizing** routes
- Detecting significant **patterns**
- Gaining **insight** about routing

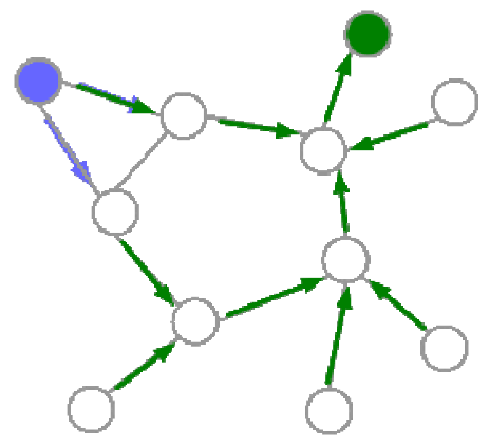
We call this **path-based** distance metric:

Routing **S**tate **D**istance

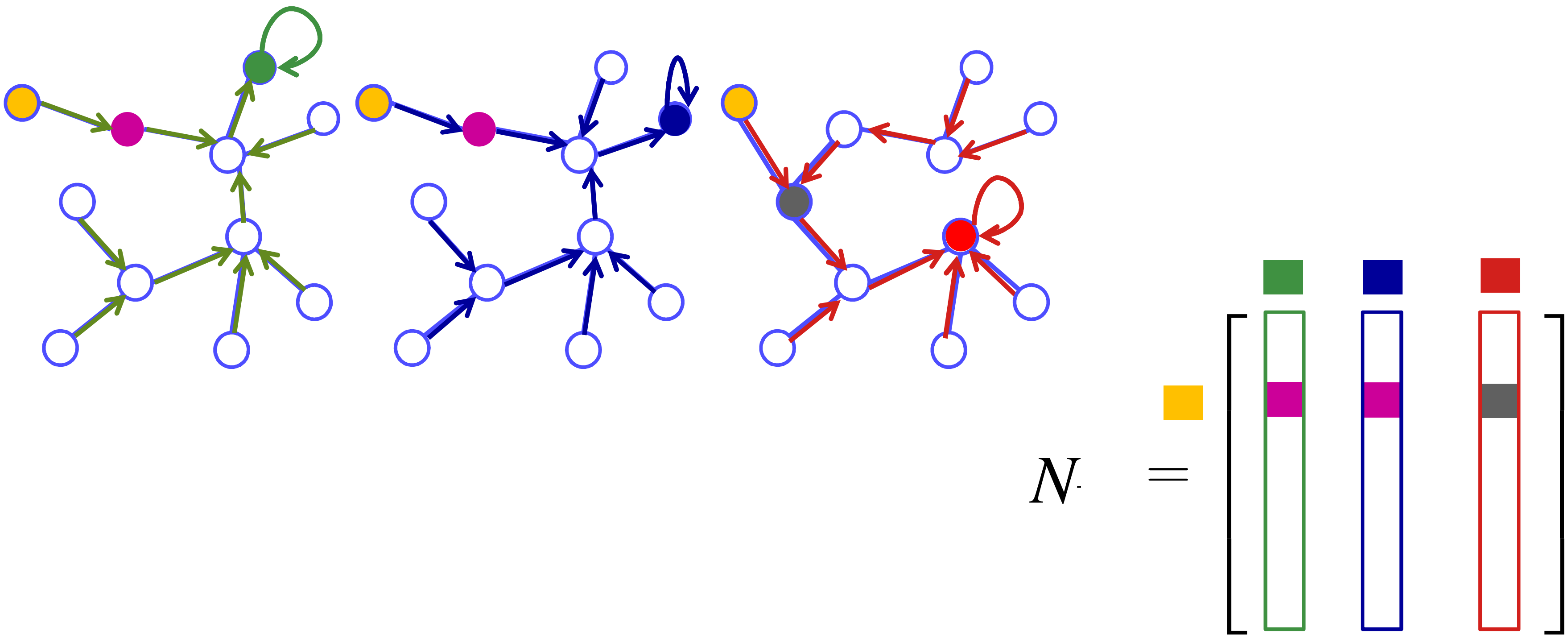
Measuring "Routing Similarity"

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





Routing State Distance (RSD)



rsd(a,b) = # of entries that differ in columns a and b of N

If rsd(a,b) is small, most nodes think a and b are **‘in the same direction’**

Formal Definition

Given a set X of destinations and a next-hop matrix N s.t.

$N(x_i, x_1) = x_j$ is the next hop on the path from x_i to x_1 ,

$$RSD(x_1, x_2) = |\{\ x_i \mid N(x_i, x_1) \neq N(x_i, x_2)\ \}|$$

RSD is a metric (obeys triangle inequality)

RSD to BGP

In order to apply RSD to measured BGP paths we define N to have all ASes on rows and prefixes on columns.

$N(a, p) =$ the next-hop from AS a to prefix p

A few issues: missing and multiple next-hops.

Dataset

- 48 million routing paths collected from
  - Routeviews and Ripe projects (publicly available)
  - Collected from 359 monitors

- Some preprocessing (details omitted)
  - 243 source ASes, 135K destinations.

$$N = \begin{bmatrix} \text{243 x 135K} \\ \\ \end{bmatrix}$$

- From N compute D, our RSD distance matrix where:

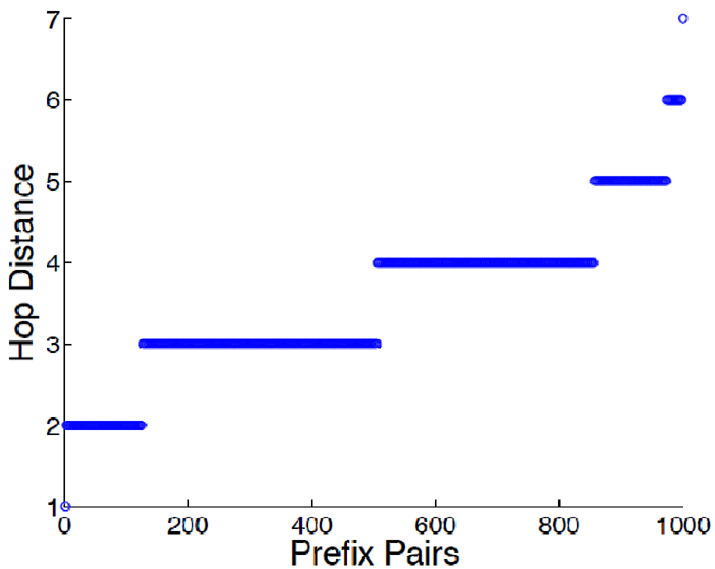
$$D(x_1, x_2) = RSD(x_1, x_2)$$

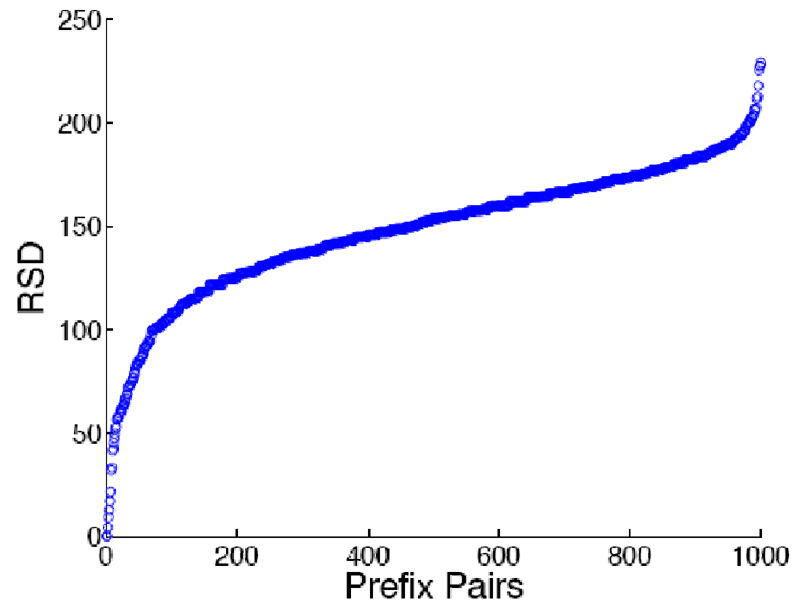
$$D = \begin{bmatrix} \text{135K x 135K} \\ \\ \end{bmatrix}$$

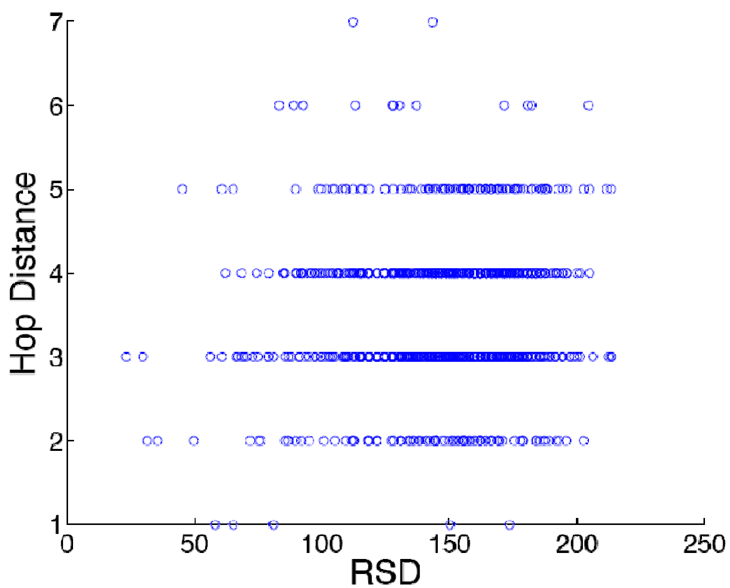
Why is RSD appealing ?

Let's look at its properties…

RSD vs. Hop Distance





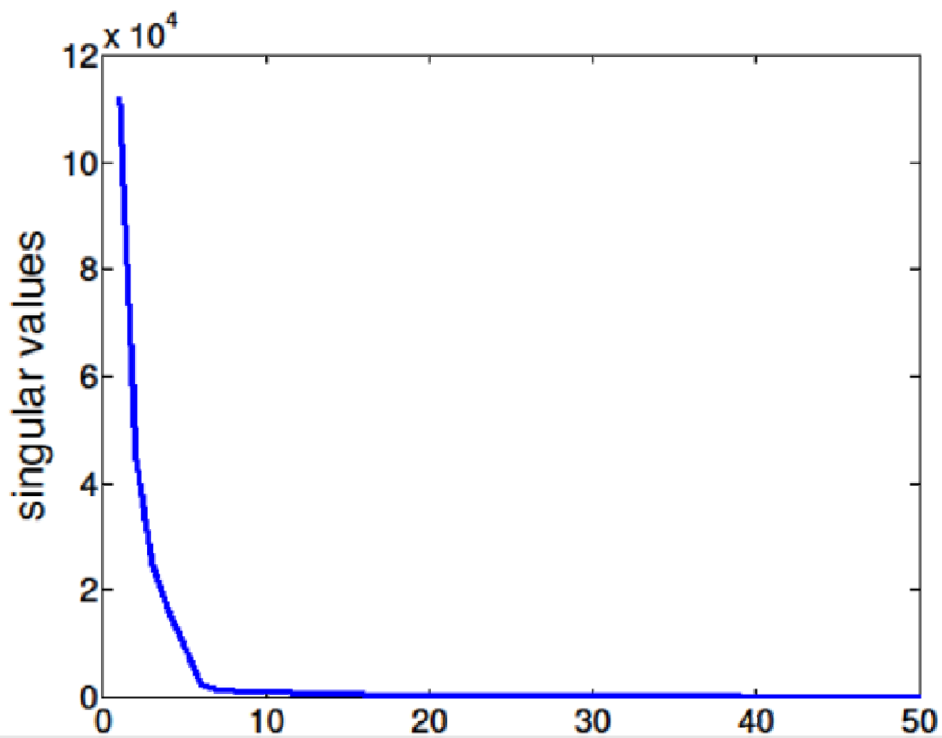


- ✓ Varies smoothly, has a gradual slope.
- ✓ Allows fine granularity.
- ✓ Defines neighborhoods.
- ✓ No relation between RSD and hop distance.

RSD for Visualization

From N compute D, our RSD distance matrix where:

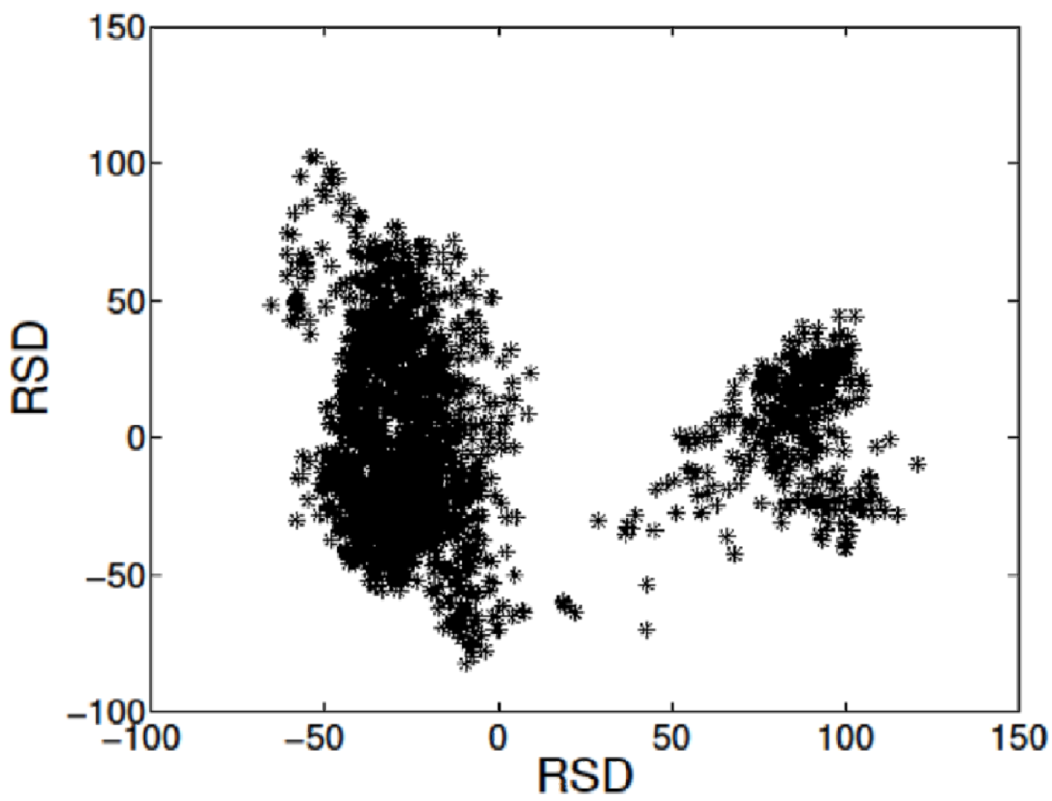
$$D(x_1, x_2) = RSD(x_1, x_2)$$



Highly **structured :**

allows 2D **visualization !**

RSD for Visualization

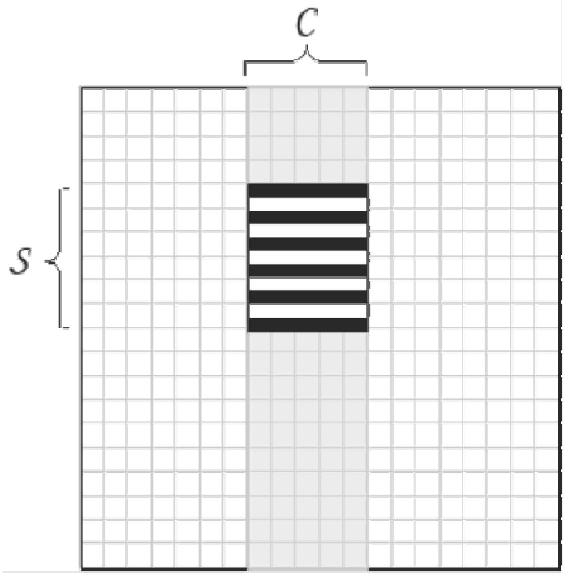


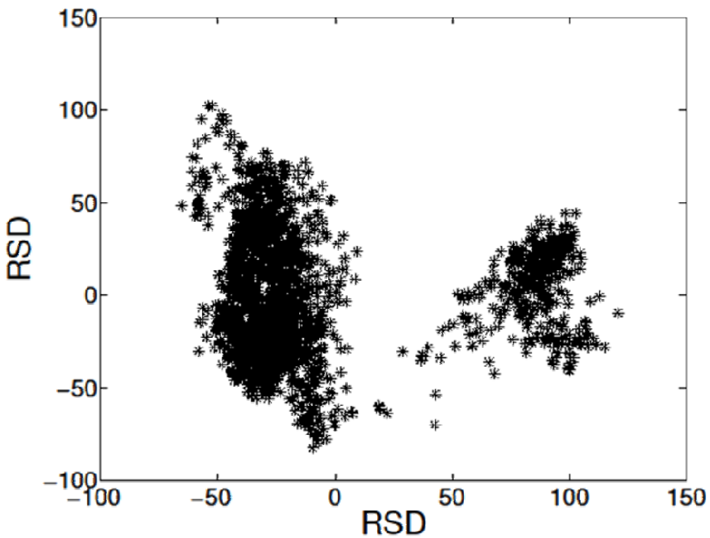
Clear Separation!

This happens with **any** random sample:

Internet-wide phenomena!

What Causes Clusters in RSD?



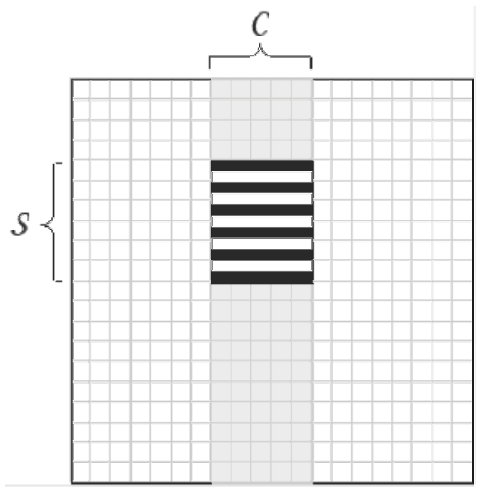


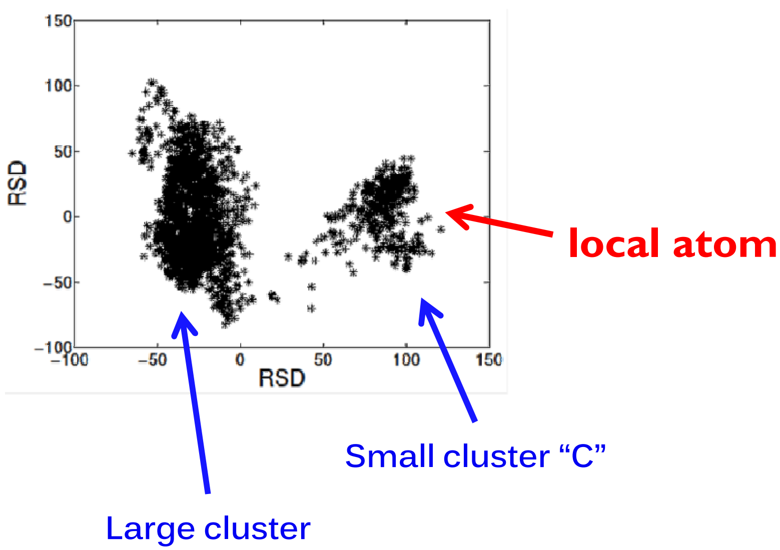
First think matrix-wise (N):

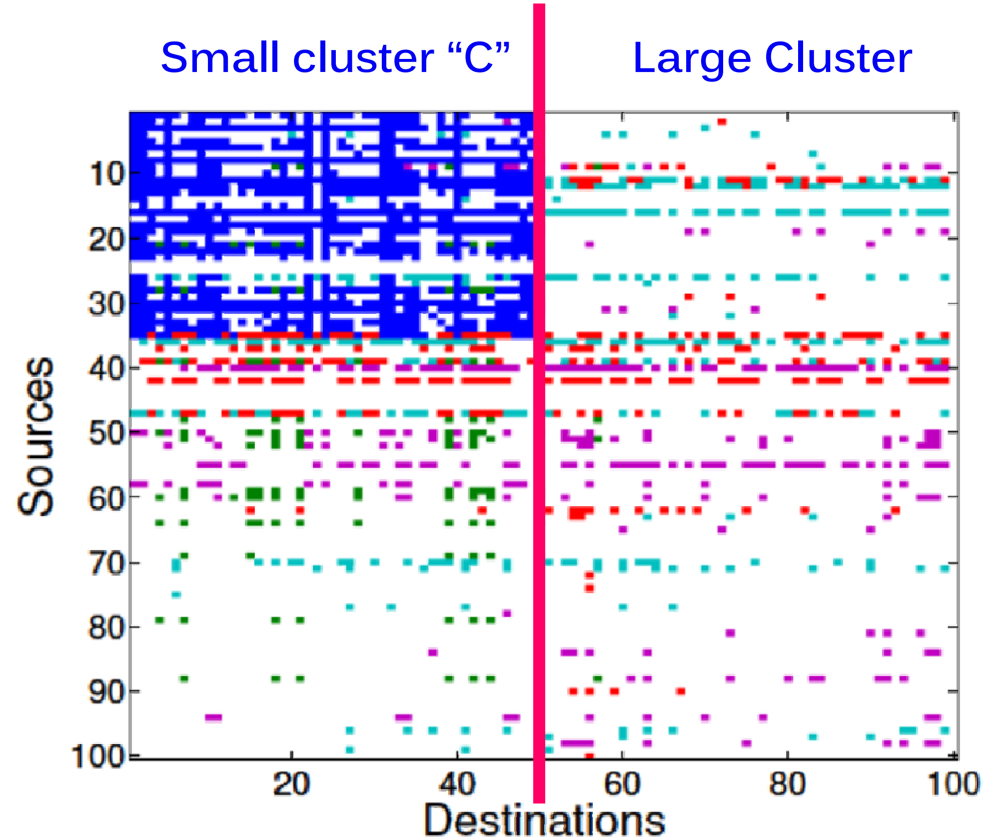
- A cluster C corresponds to set of columns
- Columns C being close in RSD means they are similar in some positions S
- N(S,C) is highly coherent

Now in routing terms:

- Any row in N(S,C) must have the same next hop in nearly each cell
- The set of ASes S make similar routing decisions w.r.t destinations C







A **local atom** is a set of destinations that are routed similarly in by a set of sources.

Why these specific destinations?

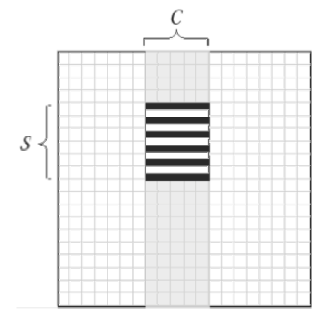
For this investigate S …

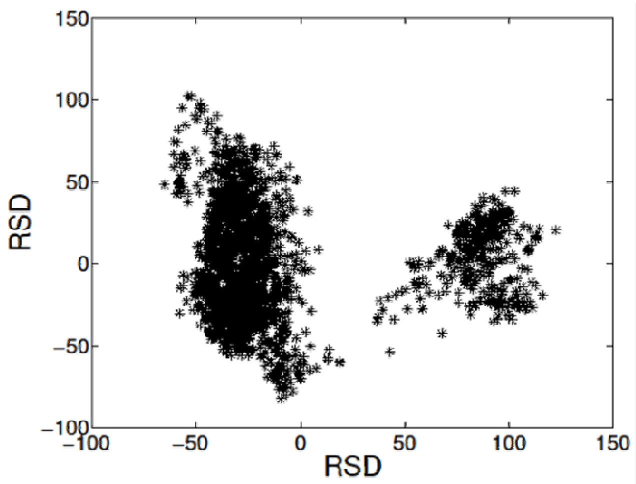
- Prefer a specific AS for transit to these destinations :

  Hurricane Electric (HE)

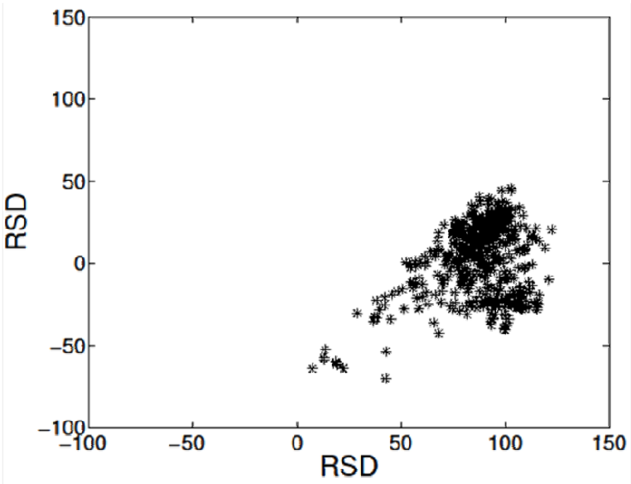
- If any path passes through HE

  1. Source ASes prefer that path
  2. Destination appears in the smaller cluster

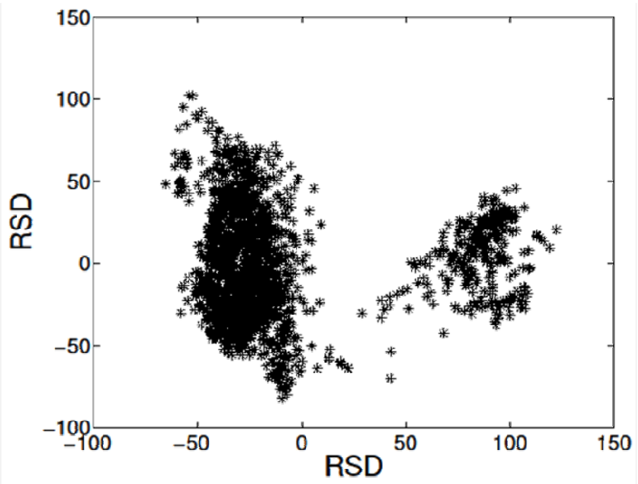




Level3



Hurricane Electric



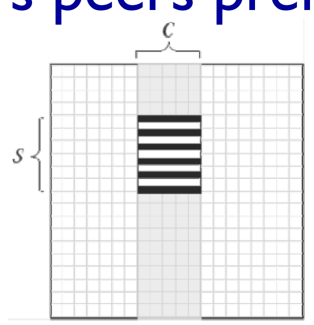
Sprint

But why do sources always route through Hurricane Electric (HE) if the option exists?

HE has a relatively unique peering policy.

It offers peering to ANY AS with presence in the same exchange point.

HE's peers prefer using HE for ANY customer of HE.



S = networks that peer with HE

C = HE's customers

Can we find more clusters ?

Analysis with RSD uncovered a macroscopic atom.

Can we formulate a systematic study to uncover other small atoms?

Intuitively we would like a partitioning of the destinations such that RSD :

- ✓ In the same group is minimized
- ✓ Between different groups is maximized

RS-Clustering Problem

Intuition: A partitioning of the destinations s.t. RSD :

✓ In the same group is minimized

✓ Between different groups is maximized

For a partition P :

$$P-Cost(P)=\sum_{\substack{x,x':\\P(x)=P(x')}} D(x,x')+\sum_{\substack{x,x':\\P(x)=P(x')}} m-D(x,x')$$

Key Advantage: Parameter-free!!

RS-Clustering is a hard problem …

Finding the optimal solution is NP-hard.

We propose two solutions:

1. Pivot Clustering
2. Overlap Clustering

Pivot Clustering Algorithm

Given a set of destinations X, their RSD values, and
a threshold parameter τ :

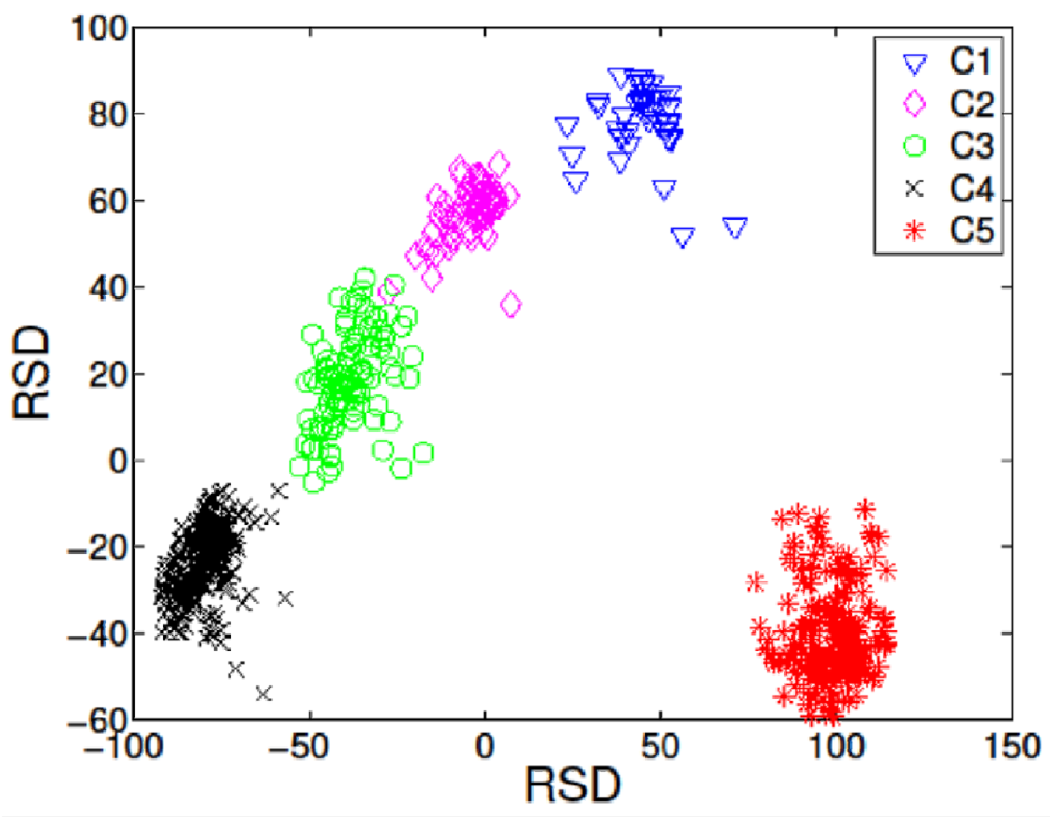
1. Start from a random destination x_i (the pivot)
2. Find all x_j that fall within τ to x_i and form a cluster
3. Remove cluster from X and repeat

Advantages:

✓ The algorithm is fast : O(|E|)

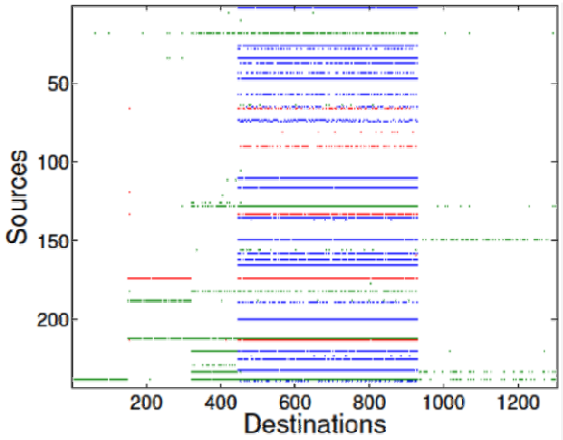
✓ Provable approximation guarantee

5 largest clusters



✓Clusters show a clear separation

✓Each cluster corresponds to a local atom



Interpreting Clusters

	Size of C	Size of S	Destinations
C1	150	16	Ukraine 83% Czech. Rep 10%
C2	170	9	Romania 33% Poland 33%
C3	126	7	India 93% US 2%
C4	484	8	Russia 73% Czech rep. 10%
C5	375	15	US 74% Australia 16%

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]
- Grouping 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]

Future Directions

1. Routing Instability Detection

   Analyzing next-hop matrices over time

2. Anomaly Detection

   Leveraging low effective rank of RSD matrix

3. BGP Root Cause Analysis

   Monitoring migration of prefixes between clusters

Take-Away

A new metric: Routing State Distance (RSD) to measure routing similarity of destinations.

- A **path-based** metric
- Capturing **closeness** useful for **visualization**
- In-depth **analysis** of AS-level routing
- Uncovering surprising **patterns**

Code, **data**, and more **information** is available on our website at:

csr.bu.edu/rsd

THANKS!

Routing State Distance: A Path-based Metric for Network Analysis

Gonca Gürsun

joint work with

Natali Ruchansky, Evimaria Terzi, Mark Crovella

BOSTON UNIVERSITY

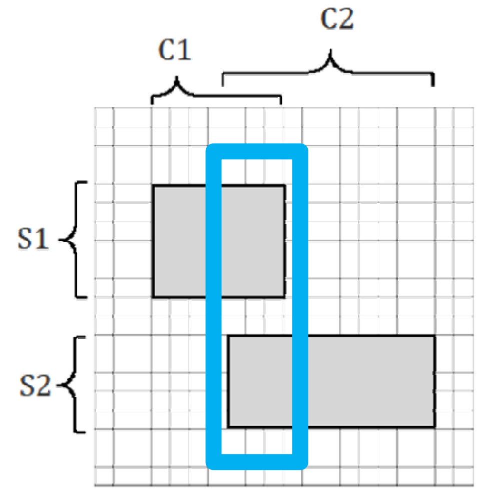
Google

NSF

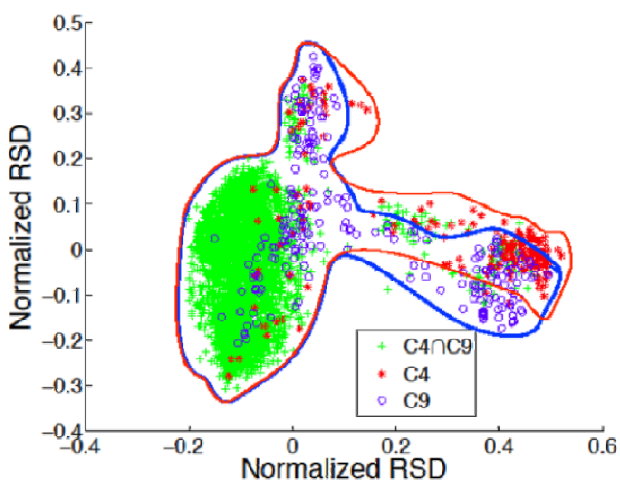
Microsoft Research

YAHOO!

We ask ourselves if a **partition** is really best?



Seek a clustering that captures **overlap**



To address this we propose a formalism called **Overlap Clustering** and show that it is capable of extracting such clusters.

Missing Values

Issue:

Measured BGP data consists of paths from a set of monitor ASes to a large collection of prefixes.

For any given (a, p) the paths may not contain information about $N(a, p)$

Solution:

1. Using only a set of high degree ASes on the rows of N
2. Rescaling $RSD(p_1, p_2)$ based on known entries both in $N(:, p_1)$ and $N(:, p_2)$

Multiple Next-Hops

Issue:

An AS may use more than one next hop for a given prefix.

Solution:

Partition that AS by its quasi-routers [**Muhlbauer** et. al. '07]

RSD Metric Proof

Proposition:

$$RSD(x_1, x_2) \leq RSD(x_1, x_3) + RSD(x_2, x_3)$$

Recall: $RSD(x_1, x_2) = |\{x_i \,|N(x_i, x_1) \neq N(x_i, x_2)\}|$

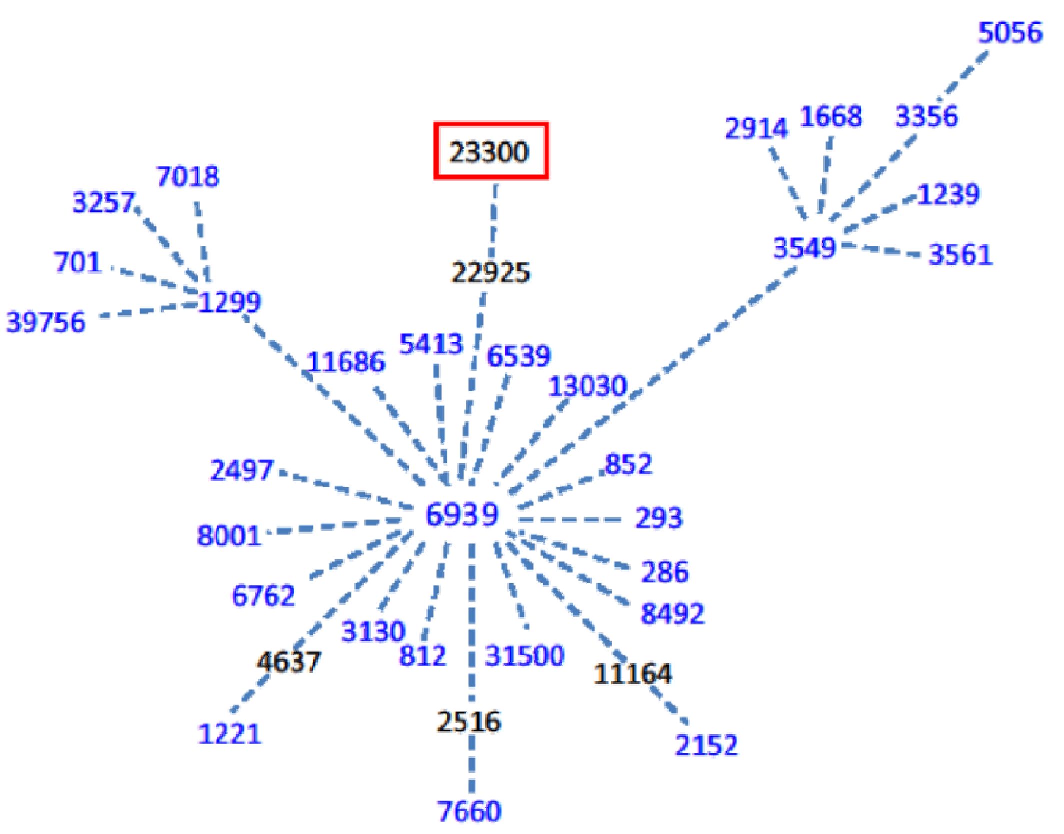
Proof:

Assume not...then there must a node x such that

$N(x, x_1) \neq N(x, x_2)$ but

$N(x, x_1) = N(x, x_3)$ and $N(x, x_2) = N(x, x_3)$

Contradiction!



BGPlay snapshot

Multi-Dimensional Scaling

Given:

a set of items $\boldsymbol{I}$ and a set of item-item distances $\{\boldsymbol{d_{ij}}\}\ \boldsymbol{i}, \boldsymbol{j} \in \boldsymbol{I}$,

Task:

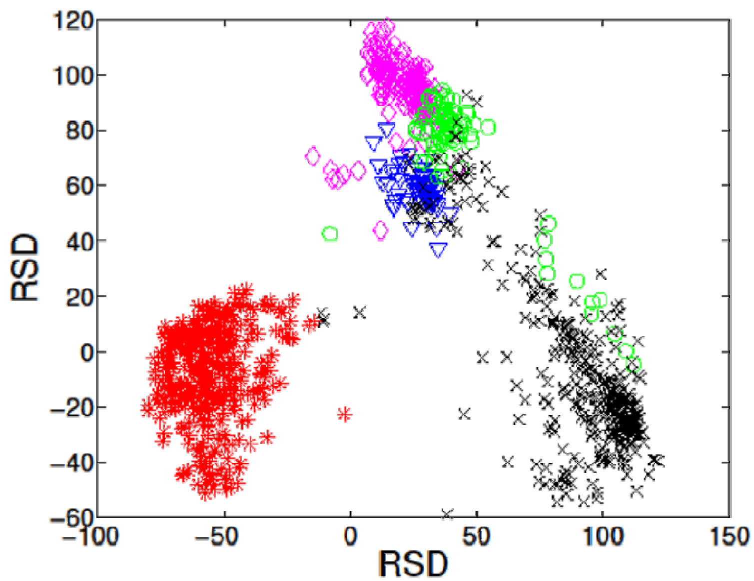
assign each item a location $\boldsymbol{x_i}$ in some r-dimensional Euclidean space.

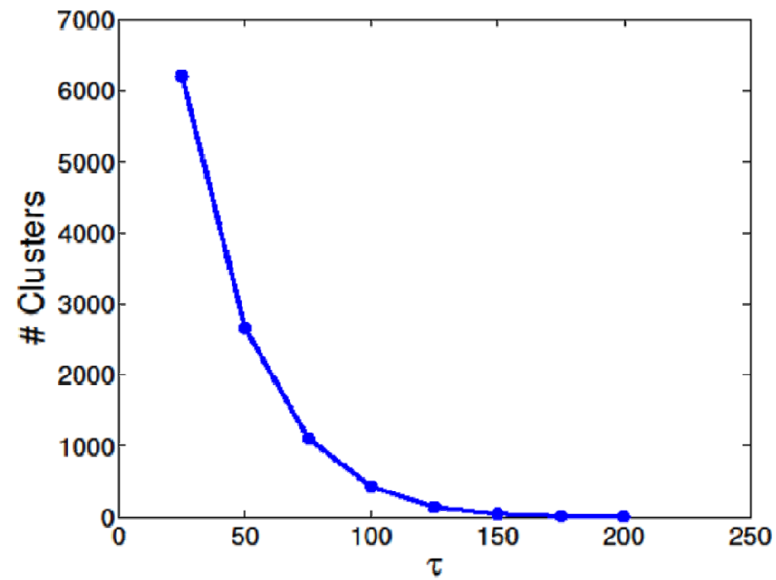
Such that:

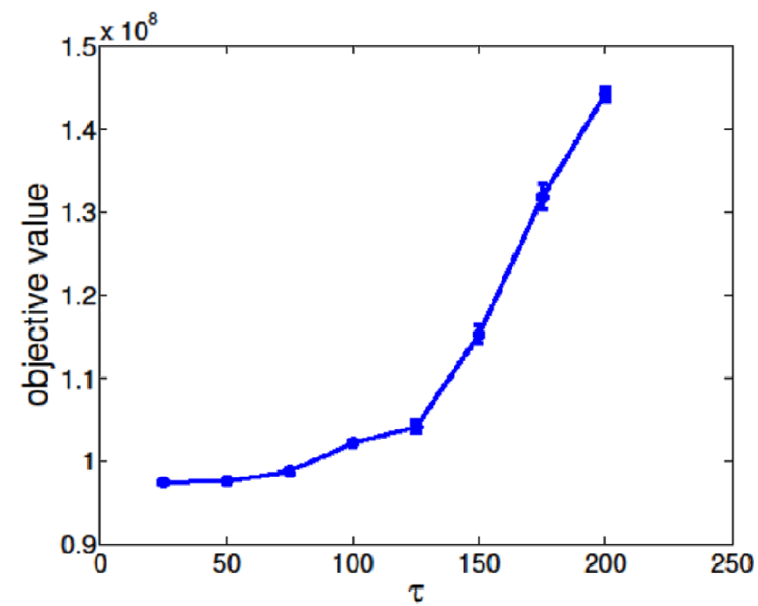
$$\min_{x_i \dots x_{|I|}} \sum_{i<j} \left(\|\boldsymbol{x_i} - \boldsymbol{x_j}\| - \boldsymbol{d_{ij}} \right)^2$$

When $\boldsymbol{r = 2}$ the results can be plotted (with distance approximately preserved)

Choosing τ







Proposition 3:

For $\frac{m}{2}$, the pivot algorithm is an expected **3-approximation** algorithm for the **RS-Clustering** problem.

$\frac{m}{2}$ translates to $\boldsymbol{\tau = 120}$.

From the plots we see that the **best choice** is $\boldsymbol{\tau = 50}$

Overlap Clustering

Given a set of prefixes $\boldsymbol{X}$, and their $\boldsymbol{RSD}$ values:

- We seek to assign to each $\boldsymbol{x_i}$ a **labeling** $\boldsymbol{L(x_i)}$ s.t. for any pair $\boldsymbol{x_i}$, $\boldsymbol{x_j}$:

  Distance between $\boldsymbol{L(x_i)}$ and $\boldsymbol{L(x_j)}$ is close to $\boldsymbol{RSD(x_i, x_j)}$
- $\boldsymbol{L(x_i)}$ corresponds to the **clusters** of $\boldsymbol{x_i}$

We use **Jaccard** distance between labels:

$$\boldsymbol{J(A, B) = \frac{A \cap B}{A \cup B}}$$

Takes a parameter $\boldsymbol{p}$:

Max number of clusters $\boldsymbol{x_i}$ can belong to.

Note that $\boldsymbol{p}$ does **not** limit the final number of clusters.

[Bonchi et al '11]

Details of Overlap Clustering

Input: $\boldsymbol{rsd}$ distance matrix $\boldsymbol{Z}$, initial clustering $\boldsymbol{S}$, and a set of prefixes $\boldsymbol{S}$

While global cost decreases

  For each $\boldsymbol{x_i} \in \boldsymbol{X}$

    Find minimum cost labeling $\boldsymbol{L_i}$

    Update $\boldsymbol{S(x_i) = L_i}$

Output $\boldsymbol{S}$

Local Search of OC

Approximated using NNLS.
Recall: $\boldsymbol{S}$ labeling, $\boldsymbol{L_i}$ vector of labels of $\boldsymbol{x_i}$, $\boldsymbol{Z}$ the $\boldsymbol{rsd}$ matrix

Key to formulation comes from rewriting Jaccard Similarity:

$$\boldsymbol{J(L_i, S(j))} = \frac{\sum_{m \in S(j)} \boldsymbol{L_i(m)}}{|\boldsymbol{S(j)}| + \sum_{m \in U} \boldsymbol{L_i(m)} + \sum_{m \in S(j)} \boldsymbol{L_i\,(m)}}$$

Since we want $\boldsymbol{J(L_i, S(j)) = 1 - Z(i, j)}$ we can write:

$$-\boldsymbol{Z L_i} + [(\boldsymbol{1} + \boldsymbol{Z}).* \boldsymbol{S}]\boldsymbol{L_i} = \boldsymbol{ZS}$$

Note that $\boldsymbol{L_i}$ is the only unknown so formulate NNLS:

$$\boldsymbol{A} = [(\boldsymbol{1} + \boldsymbol{Z}).* \boldsymbol{S}] - \boldsymbol{Z} \text{ and } \boldsymbol{b} = \boldsymbol{ZS}$$

Post Processing of OC

Drawbacks of NNLS:

1. No constraint of max $\boldsymbol{p}$ labels
2. Output $\boldsymbol{x}$ not restricted to 0-1

Instantiate a **Greedy** post-processing:

- **Sort** $\boldsymbol{x}$ in decreasing order
- Obtain $\boldsymbol{x_q}$ by setting $\boldsymbol{x(1:q) = 1}$ and rest to $\boldsymbol{0}$
- Vary $\boldsymbol{q = 1:p}$
- Select $\boldsymbol{x_q}$ with **minimum cost**

Cost Functions of OC

$\boldsymbol{X}$: the set of prefixes

$\boldsymbol{L_i}$: the labeling assigned to $\boldsymbol{x_i}$

$\boldsymbol{Z}$: the $\boldsymbol{rsd}$-distance matrix

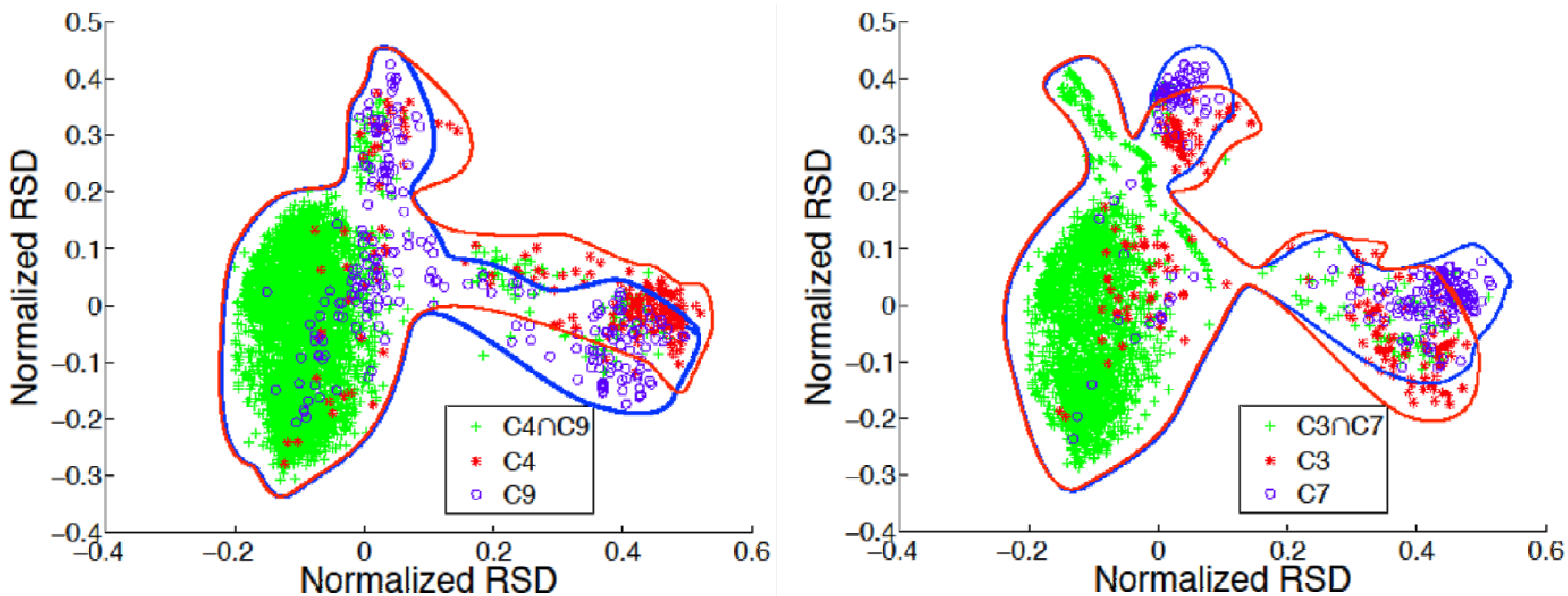
Local Cost:

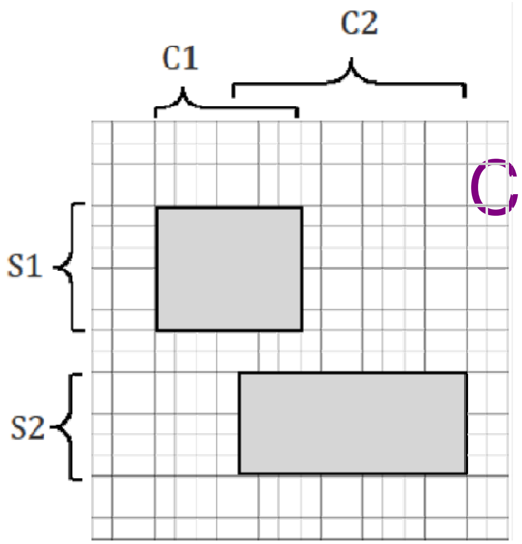
$$\boldsymbol{C_{i,p}(L_i \mid S) = \sum_{j \in X} |J(L_i, S(j)) - Z(i,j)|}$$

Global Cost:

$$\boldsymbol{C(X,S) = \frac{1}{2} \sum_{j \in X} C_{i,p}(L_i \mid S)}$$

Overlap Clustering

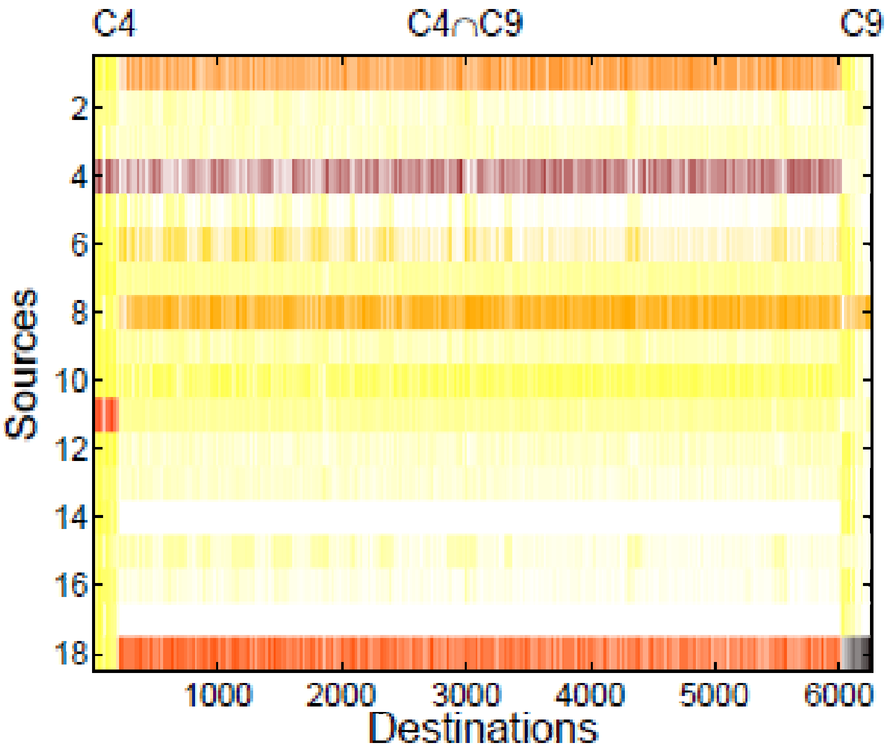




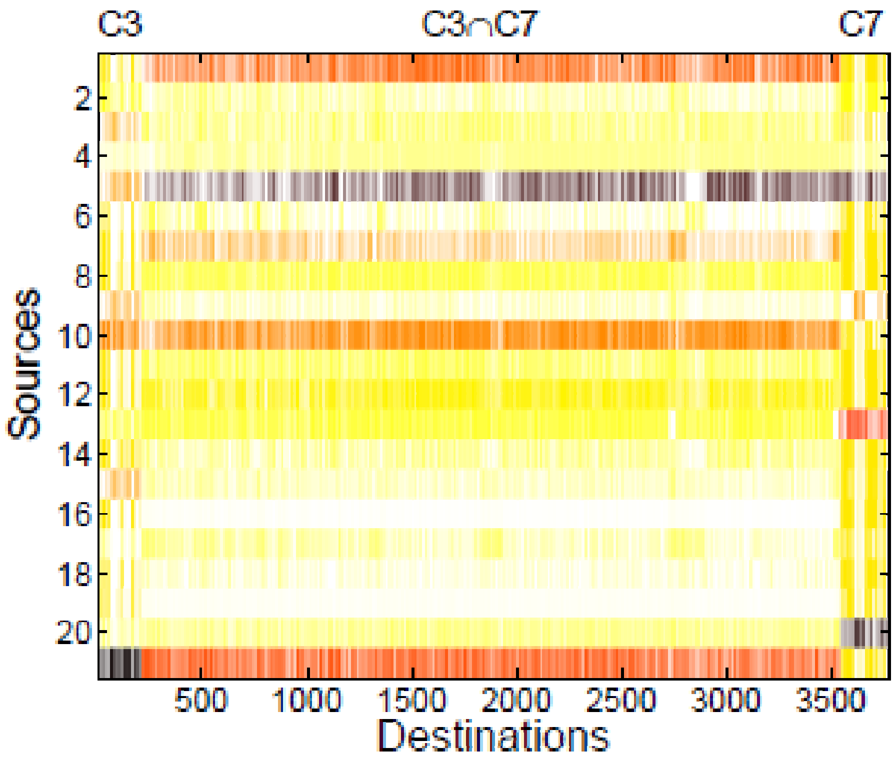
Comparison with non-overlapping

- Each of $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$ causes prefixes to cluster together (independently)
- Do not find that clusters typically map to geographic set
- Not as compact in $\boldsymbol{RSD}$ space
- Find clusters in which different AS sets have coherent routing over overlapping prefix sets

OC Visual



C4
C4∩C9
C9
Sources
Destinations



C3
C3∩C7
C7
Sources
Destinations

Clustering Algorithm Comparison

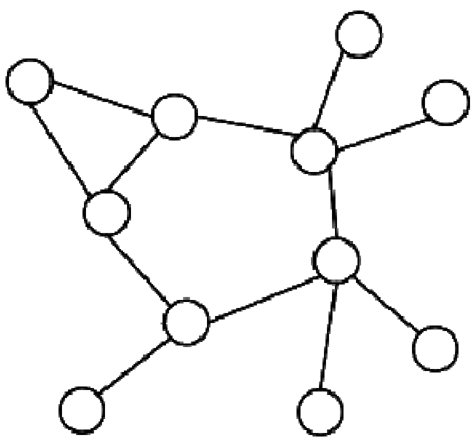
- Operate over a continuous space [Kmeans]
  - *Our data is* ***categorical*** *(next hops)*
- Require defining a '**representative**' [Kmedian]
  - *This is not clear in the* ***RSD*** *metric space*

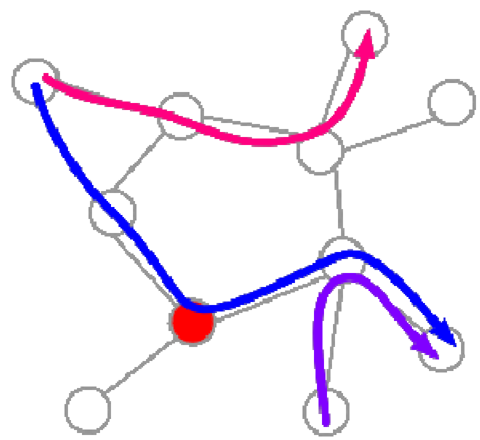
Furthermore,

- Input **number of clusters**
- Objective function guaranteed to decrease as number of clusters increase

Motivating Problem

- What paths pass through my network?
  - If someone at Boston University were to send an email to Telefonica, would it go through my network?
- Important for network planning, traffic management, security, business intelligence.





Surprisingly hard!

Inferring Visibility: Who is (not) Talking to Whom?,
<u>Gürsun</u>, Ruchansky, Terzi, Crovella, In the proc. of SIGCOMM 2012.

A New Metric

A **new** metric **path- based** metric that can use used for:

We only have an incomplete view of the AS graph [Roughan et. al. '11]

- **Visualization** of networks and routes
  - Visualization based on AS degree and geo-location [Huffaker '10]
  - Small scale visualization through BGPlay and bgpviz
- **Characterizing** routes
  - Clustering on the inferred AS graph [Gkantsidis et. al. '03]
- Detecting significant **patterns**
- Gaining **insight** about routing

RSD in Practice

- Key observation: we don't need all of N to obtain a useful metric

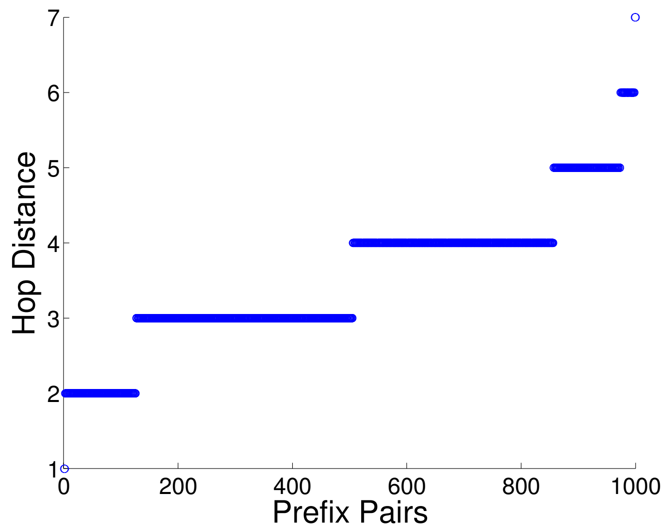
- Many (most?) nodes contribute little information to RSD
  - Nodes at edges of network have nearly-constant rows in H

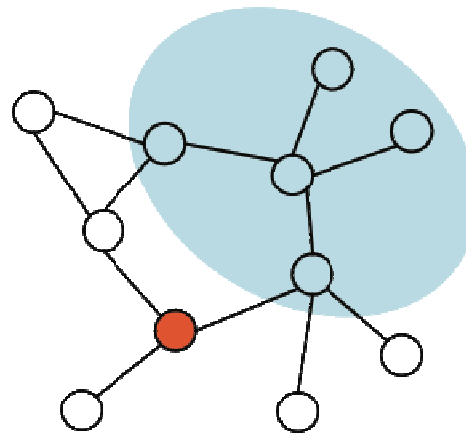
- Sufficient to work with a small set of well-chosen rows of N

- 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

Seeking a metric for 'neighborhoods'

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





- 90% of destination pairs have hop distance < 5
  - Clearly, typical distance metric is inappropriate
- Need a metric that expresses 'routed similarly in the Internet'
  - or other graph