Today’s Agenda

- AS relationship inference
  - Why is it important?
  - Why is it difficult?
  - Some algorithms proposed in the literature
  - Evaluating inference algorithms
  - Some open problems.
Applications of Accurate Global AS Graphs

- Internet service management
  - Placement of proxy servers, web-hosting servers
- Help administrators in
  - Traffic engineering
  - Network management, debugging, fixing problems
- Aid ISPs in signing contracts
- Aid inter-domain routing algorithms (avoid route divergence, e.g.)
- Verify consistency of IRR database
- Help optimize various network protocols and services (P2P, CDN, ...)

SUNY at Buffalo; Computer Science; CSE620 – Advanced Networking Concepts; Fall 2005; Instructor: Hung Q. Ngo
Challenges of Inferring AS Relationships (1)

1. AS relationships come from contracts between ISPs, which they don’t want to reveal
   - Hard to evaluate inference algorithms!

2. Multiple sources of information, which could be contradicting
   - Internet registries like ARIN
   - BGP routing table entries from different Ass

3. Information is incomplete and erroneous (configuration errors, e.g.)
   - Only have access to a subset of BGP tables
   - Some edges are impossible to see without direct BGP table access (e.g. peer-peer edge between small ASes)
Challenges of Inferring AS Relationships (2)

4. Geography might play a role
   - E.g., contract between X & Y in the U.S. may be different from that in Europe

5. AS pairs may have *back-up* relationships

6. AS pairs may peer indirectly
Some Basic Assumptions (1)

- Four types of AS-AS relationships
  - Provider-Customer, Customer-Provider: $$$
  - Peer-Peer: exchange traffic between customers
  - Sibling-Sibling: two AS’s belong to the same administrative domains, or have a mutual transit agreement

- Rules for exporting routes:
  - To a provider or a sibling: its own, customers’, siblings’, not peers’ nor other providers’
  - To a customer or a peer: everything
Some Basic Assumptions (2)

Traffic allowed

Traffic NOT allowed

Sibling-sibling

Peer-Peer

Prov. – Cust.
Input Data

- Looking Glass Servers: BGP routing table snapshots
  - Route Viewer Server (Oregon)
  - NANOG’s looking glass links
- Routing Arbiter Database (RADB)
  - Archives of routing policies
- Internet Routing Registries
  - > 50 routing policy databases conforming to RPSL
- WHOIS Services
  - Name/addresses of routing domain owners
- Traceroute
  - Gives much more specific data
How to Evaluate Inference Algorithms

- Small number of invalid paths
- Compare outputs of different algorithms
- Compare outputs with data from IRR, RADB and other sources
- Compare outputs with proprietary data
Some Current Solutions

- Lixin Gao, ToN 2000
- Subramanian et al., SIGCOMM 2002
- Battista et al., INFOCOM 2003; Erlebach et al., CCN 2002
- Xia & Gao, GLOBECOM 2005
- Some others
Uphill path: sequence of C-P and S-S edges
Downhill path: sequence of P-C and S-S edges
AS Paths are “valley free”
  - Uphill path
  - Downhill path
  - Uphill, then downhill
  - Uphill, then P-P edge
  - P-P edge, then downhill
  - Uphill, P-P edge, then downhill

In general, AS paths have the form
  - [Uphill] ° [P-P edge] ° [Downhill]
Basic Algorithm

1. Compute degrees of ASes
2. For each path in routing tables
   ■ Highest degree AS assumed to be top provider
   ■ Pairs on the left are C-P
   ■ Pairs on the right are P-C
3. For each AS pairs (which are connected)
   ■ If labeled both C-P and P-C, then it’s S-S
Refined Algorithm

- Idea: allows 1 error in C-P or P-C classification

1. For every edge e
   - $x = \#$ of paths classifying e as C-P
   - $y = \#$ of paths classifying e as P-C
2. If $(x > y = 0 \text{ or } 1)$ then e is C-P
3. If $(y > x = 0 \text{ or } 1)$ then e is P-C
4. Otherwise, e is S-S
Final Algorithm

1. Coarsely classify edges into C-P, P-C, S-S using either basic or refined algorithms
2. Identify pairs that cannot be P-P
   - P-P can only involve top provider in a path
   - Plus another heuristic (top provider is likelier to peer with higher degree neighbor)
3. For the rest of the edges, classify as P-P if the difference in degrees is at most R (=60)
Findings

- 90.5% AS pairs are C-P or P-C
- 1.5% AS pairs are S-S
- < 8% AS pairs are P-P

- 99.8% inferred relationships are confirmed by AT&T internal data
- 50% of S-S relationships are confirmed by WHOIS
  - Not confirmed doesn’t mean they’re wrong (yet)

- Can also identify erroneous routing table entries
  - Mis-configuration
  - Weird AS relationships
[Subramanian et al., SIGCOMM 2002]

- Only assume P-P, C-P, and P-C
- Exporting rules:
  - To a provider: its own, customers’, \textbf{not} other providers and peers
  - To a peer: its own, customers’, \textbf{not} providers and peers
  - To a customer: everything
- Data gathered from multiple \textit{vantage} points
- Valid paths: $+...+-...-, \text{ or } +...+0-...-$
- \textbf{ToR Problem}
  - Given a graph $G=(V,E)$, and a set of paths $P$
  - Find an edge-labeling with $+, -, 0$ to minimize the number of invalid paths
Heuristic: Observation and Key Ideas

- **Keys**
  - Find breaking point between downhill & uphill
  - Reconcile differences viewed from vantage points
- **Observations**
  - Provider-Customer relationship is acyclic
  - Take view source into account: views from tier-1 ASes tend to be acyclic
Heuristic: AS Ranking

- Rank each AS for each vantage point
  - Leaves are ranked 1
  - Recursively remove leaves and increase ranks
  - For the final connected component, assign last rank

- Map $AS_i$ to a vector $(r_{i1}, \ldots, r_{in})$
  - $r_{ij}$ = rank of $AS_i$ viewed from vantage point $j$
  - $l_{ij} = |\{k : r_{ik} > r_{jk}\}|$
    number of vantage points that rank $i$ higher than $j$
  - $e_{ij} = |\{k : r_{ik} = r_{jk}\}|$
    number of vantage points that rank $i = j$
Heuristic: Inferring Peer-Peer Relationship

- **Equivalence**
  - \( \text{AS}_i \) and \( \text{AS}_j \) are *equivalent* if \( e_{ij} > n/2 \)
  - If they are connected, then they are peers
  - This deals with peers visible from lots of vantage points

- **Probabilistic equivalence**
  - \( \text{AS}_i \) and \( \text{AS}_j \) are prob. equiv. If \( 1/\delta_1 \leq l_{ij}/l_{ji} \leq \delta_1 \)
    (\( \delta_1 \) is close to 1, chosen to be 2 in the paper)
  - If they are connected, then they are peers
  - This deals with peers partially visible
Heuristic: Inferring Provider-Customer Relationship

- **Dominance**
  - $AS_i$ dominates $AS_j$ if $l_{ij} \geq n/2$ and $l_{ji} = 0$
  - If they share an edge, $AS_i$ is a provider of $AS_j$

- **Probabilistic dominance**
  - If $l_{ij}/l_{ji} > \delta_0$ for large $\delta_0 > \delta_1$
    ($\delta_0 = 3$ in the paper)

- **In applying the rules**
  - Deterministic dominance and equivalence applied first
  - Probabilistic ones after
Findings

- Validation based on # of path anomalies
- Percentage of anomalies between 0.6% and 3.0%
  - Many anomalies due to sibling relationship between ASes of the same administration
  - Merging, splitting of ISPs
- Result helps classify Internet ASes into 5 hierarchical levels
  - Dense core
  - Transit core
  - Outer core
  - Small regional ISPs
  - Customers
Works on the ToR Problem

- Determining if there is a good labeling
  - [Battista et al., INFOCOM 2003], [Erlebach et al., CCN 2002]
  - Can be done in linear time with a reduction to 2SAT
  - If there is a good labeling, it’s linear-time computable
  - Solve 2SAT by computing strongly connected components and topological ordering (standard technique!)

- Max-ToR Problem [Erlebach et al., CCN 2002]
  - NP-Hard
  - Cannot be approximated to within $n^{1-\varepsilon}$ for any $\varepsilon > 0$ unless $\text{NP} = \text{coRP}$
    (Reduction from Independent Set)

- Max-ToR with bounded path lengths [Erlebach et al., CCN 2002]
  - APX-complete in general
  - Approximable to within $2^k/(k+1)$ where $k$ is path-length upper bound
[Xiao et al. GLOBECOM 2005]

- Obtained partially real AS relationships
  - Usages of BGP community attributes
  - Instances of AS-SET objects in IRR databases
  - Routing policies in IRR databases
  - Partial AS relationships from RADB and IRR databases
- Evaluated algorithms by [Gao, 2000], [Subramanian et al., 2002]
  - Did not touch algorithms by [Battista et al 2003]
  - The other two don’t predict peer-peer edges well (<50%)
- Proposed a new heuristic
  - Out-perform the other two on peer-peer, now obtain about 90% accuracy