Characterizing the Internet Hierarchy from Multiple Vantage Points

Lakshminarayanan Subramanian, Sharad Agarwal, Jennifer Rexford, Randy H. Katz

***Abstract*— The delivery of IP traffic through the Internet depends on the complex interactions between thousands of autonomous systems (ASes) that exchange routing information using the Border Gateway Protocol (BGP). This paper investigates the topological structure of the Internet in terms of customer-provider and peer-peer relationships between ASes, as manifested in BGP routing policies. We describe a technique for inferring AS relationships by exploiting partial views of the AS graph available from different vantage points. Next we apply the technique to a collection of ten BGP routing tables to infer the relationships between neighboring ASes. Based on these results, we analyze the hierarchical structure of the Internet and propose a five-level classification of ASes. Our characterization differs from previous studies by focusing on the commercial relationships between ASes rather than simply the connectivity between the nodes.**

I. Introduction

TODAY'S Internet is divided into more than 10,000 Autonomous Systems (ASes) that interact to coordinate the delivery of IP traffic. An AS typically falls under the administrative control of a single institution, such as a university, company, or Internet Service Provider (ISP). Neighboring ASes use the Border Gateway Protocol (BGP) [1], [2] to exchange information about how to reach individual blocks of destination IP addresses (prefixes). An AS applies local policies to select the best route for each prefix and to decide whether to propagate this route to neighboring ASes, without divulging these policies or the AS's internal topology to others. In practice, BGP policies reflect the commercial relationships between neighboring ASes. AS pairs typically have a customer-provider or peer-peer relationship [3], [4]. A provider sells connectivity to the Internet as a service to its customers, whereas peers provide connectivity between their respective customers.

AS relationships and the associated routing policies have a profound influence on how traffic flows through the Internet. An understanding of the structure of the Internet in terms of these relationships facilitates a wide range of important applications. For example, consider a content distribution network (CDN) that can place replicas of a Web site in data centers hosted by different ASes. The company can identify the IP prefixes and ASes responsible for a large portion of the traffic from the site [5]. With an accurate view of the connectivity and relationship between ASes, the company can identify the best locations for its replicas. As another example, consider a new regional ISP that wants to connect to a small number of upstream providers. An accurate view of the AS topology and relationships can help the ISP determine which ASes would provide the best connectivity to and from the rest of the Internet.

L. Subramanian, S. Agarwal and R. H. Katz are with the Computer Science Division, University of California, Berkeley, CA, USA. E-mail: {lakme, sagarwal, randy}@eecs.berkeley.edu. J. Rexford is with the Internet and Networking Systems Center at AT&T Labs - Research in Florham Park, NJ, USA. E-mail: jrex@research.att.com. S. Agarwal is supported by DARPA (Defense Advanced Research Projects Agency) Contract No. N00014-99-C-0322.

The Internet topology alone does not provide enough information to answer these questions. For example, suppose that AS B connects to two providers, AS A and AS C. An AS graph would show connectivity from A to B and from B to C; however, AS B's routing policies would not permit transit traffic between A and C. In addition to determining the relationship between AS pairs, it is useful to identify the position of individual ASes in the Internet hierarchy. For example, a growing ISP or CDN company may need to identify the set of so-called *tier-1* providers to aid in selecting potential private peering relationships. This requires performing further analysis of the AS graph in terms of the commercial relationships between the various AS pairs. A classification of ASes based only on the topological structure (say, node degree) is not sufficient. For example, some ASes near the edge of the Internet may have a relatively large number of upstream providers, whereas some large ISPs may have modest node degree consisting mainly of private peering relationships. Knowing the relationships between the ASes is important for making these distinctions.

In the absence of a global registry, the AS-level structure of the Internet is typically inferred from analysis of routing data. Previous work has focused on constructing a view of the AS graph from traceroute experiments or individual BGP table dumps. Traceroute provides a view of the path from a source to a destination host at the IP level. The traceroute data must be analyzed to infer which interfaces belong to the same router and which routers belong to the same AS [6]. Running experiments between multiple source-destination pairs provides a larger collection of paths over time [6], [7], [8]. Other studies have extracted AS paths directly from BGP routing tables or BGP update messages [9], [10]. The routing table dump from the University of Oregon RouteViews server [11], [12] has been the basis of several studies of basic topological properties, such as the distribution of node degrees [13], [14]. With the exception of the work in [10], [15], these studies have focused on the topological structure without considering the relationship between neighboring ASes. [10] presents a heuristic for inferring the relationships from a collection of AS paths and evaluates the technique on the RouteViews data.

In this paper, we propose a technique for combining data from multiple vantage points in the Internet to construct a more complete view of the topology and the AS relationships. Each vantage point offers a partial view of the Internet topology as viewed from one source node. Due to the presence of complex routing policies, these partial views are not necessarily shortest-path trees and may, in fact, include cycles. We generate a directed AS-level graph from each vantage point and assign a *rank* to each AS based on its position. Then, each AS is represented by the vector that contains its rank from each of the routing table dumps. We infer the relationship between two ASes by comparing their vectors. Based on these relationships, we construct a new directed AS graph and examine the AS level hierarchy of the Internet. We present a five-level classification of ASes with a top-most level that consists of a rich set of peer-peer relationships between 20 so-called *tier-1* providers.

The work we describe in this paper is novel in three ways. First, we analyze AS paths seen from multiple locations to form a more complete view of the graph. Second, rather than simply combining the data from the various vantage points, we propose a methodology for exploiting the uniqueness of each view to infer the relationships between AS pairs. Third, we characterize the hierarchy of ASes based on the commercial relationships, rather than simply the connectivity of the graph. We evaluate our technique on a collection of ten BGP routing tables and summarize the characteristics of the AS relationships. To validate the inferences, we check for paths that are not consistent with the routing policy assumptions underlying customer-provider and peer-peer relationships. We show that these cases account for a small proportion of the paths and that the most common inconsistencies may stem from misconfiguration or more complex AS relationships.

II. Problem Formulation

In this section, we present a brief overview of how AS relationships affect BGP export policies and formally define the Type of Relationship (ToR) problem. Then we discuss the practical challenges that arise in solving this problem and validating a potential solution.

A. Type-of-Relationship Problem

The relationships between ASes arise from contracts that define the pricing model and the exchange of traffic between domains. ASes typically have a *provider-customer* or *peer-peer* relationship [3], [4]. In a provider-customer relationship, the customer is typically a smaller AS that pays a larger AS for access to the rest of the Internet. The provider may, in turn, be a customer of an even larger AS. In a peer-to-peer relationship, the two peers are typically of comparable size and find it mutually advantageous to exchange traffic between their respective customers. These relationships translate directly into policies for exporting route advertisements via BGP sessions with neighboring ASes:

- *Exporting to a provider:* In exchanging routing information with a provider, an AS can export its routes and routes of its customers, but cannot export routes learned from other providers or peers.
- *Exporting to a peer:* In exchanging routing information with a peer, an AS can export its routes and the routes of its customers, but cannot export routes learned from other providers or peers.

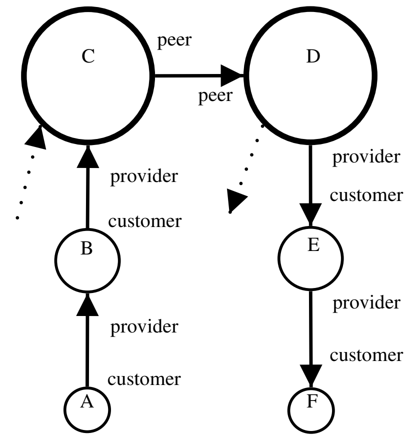

Fig. 1. Type-2 path following export rules

- *Exporting to a customer:* An AS can export its routes, routes of its customers, and routes learned from other providers and peers to its customer.

Each BGP session defines a relationship between the two ASes it connects. Although two ASes may have multiple BGP sessions, the relationship between the two ASes should be uniquely defined.

BGP export policies have a direct influence on the AS paths seen from a particular vantage point in the Internet. If every AS adheres to the customer and provider export rules, then no path should traverse a customer-provider edge after traversing a provider-customer or peer-peer edge, and no path would ever traverse more than one peer-peer edge [10], [16]. To formulate these properties in mathematical terms, we denote an edge from a customer to a provider with a -1, an edge from one peer to another with a 0, and edge from a provider to a customer with a $+1$. Restating a result from [10] in these terms, we have:

Theorem 1: If every AS obeys the customer, peer, and provider export policies, then every advertised path belongs to one of these two types for some $M, N \geq 0$:

1) *Type-1*: $-1, \ldots$ (N times), $+1, \ldots$ (M times).
2) *Type-2*: $-1, \ldots$ (N times), 0, $+1, \ldots$ (M times).

The first stage of a Type-1 path contains only customer-provider links (*uphill* portion) and the second stage contains only provider-customer links (*downhill* portion). The second type captures all paths which traverse exactly one peering link. The single peering link must appear in between the uphill and the downhill portions of any path. This is shown in Figure 1, which is a Type-2 path where M and N are both 2. The dotted lines indicate where other paths would likely intersect this path. Note that if all the directions are reversed, another valid path will be shown.

The type-of-relationship (ToR) problem can be formulated as a graph theory optimization problem for labeling the edges of the graph with a -1, 0, or 1 such that the observed paths obey the export policies implied by the relationships. Given a graph G with each edge labeled as -1, 0 or $+1$, a path p is said to be *valid* if it is either of Type-1 or Type-2:

ToR Problem: Given an undirected graph G with vertex set V and edge set E and a set of paths P, label the edges in E as either -1, 0 or $+1$ to maximize the number of *valid* paths in P. G represents the entire Internet topology where each node is an AS and each edge represents a relationship between the incident pair of ASes. P consists of all paths seen from the various vantage points. Although we believe that the ToR problem is NP-complete, we have not been able to prove this claim. We are not aware of any polynomial-time solution to the problem.

B. Practical Challenges

Identifying the commercial relationships between ASes is challenging in practice. First, peer-peer relationships are difficult to classify. Consider the path in Figure 1. Mistakenly labeling the C to D edge as a customer-provider or provider-customer edge would not result in an invalid path; rather, the Type-2 path would appear as a Type-1 path. As such, identifying peer-peer edges requires the use of heuristics. Second, some AS pairs do not obey the BGP policy guidelines outlined in Section II-A. For example, two ASes operated by the same institution may have a *sibling* relationship, where each AS exports all of its routes to the other AS [10]. Other AS pairs may have *backup* relationships to provide connectivity in the event of a failure [16]. Alternatively, two ASes may peer indirectly through an intermediate AS [17]. Also, an AS pair may have different relationships for certain blocks of IP addresses; for example, an AS in Europe may be a customer of an AS in the United States for some destinations and a peer for others. Router misconfiguration may also cause violations in the export rules. For example, a customer may mistakenly export advertisements learned from one provider to another.

Previous work in [10] proposed and evaluated an algorithm for inferring AS relationships from a collection of AS paths. For each AS path, the algorithm assumes that the node with the highest edge degree marks the boundary between the uphill and downhill portions of the path. In the uphill portion, each node provides transit service for the previous node; in the downhill portion, each node provides transit service for the subsequent node. The inferences from multiple paths are later combined to infer the relationship between the ASes. If ASes u and v provide transit service to each other, the two ASes are siblings; if u provides transit service to v but v does not provide transit service to u, u is a provider of v. Peer-peer edges are detected using a heuristic that considers the degree of the nodes adjacent to the top provider in the path; nodes with similar degree can be classified as peers. In the next section, we propose an alternate approach to the ToR problem based on the paths seen from different vantage points. Our inference technique does not depend on node degree and can tolerate occasional exceptions to the export rules in Section II-A. We classify edges as provider-customer or peer-peer and use the (small number of) invalid paths to identify AS pairs that have unusual relationships (e.g., sibling, backup, or misconfiguration).

Evaluating any solution for the ToR problem requires access to a large set of paths P. In the past few years, a number of service providers have made their BGP routing tables available to the public. Coordinated efforts such as the RouteViews project and the RIPE Routing Information Service provide BGP data from multiple locations in the Internet. Still, the publicly-available tables do not provide a complete view of the AS graph or the set of AS paths. Most of the public tables come from large ISPs near the top of the Internet hierarchy. Certain edges are difficult to see from these vantage points [18]. For example, a peer-peer edge between two universities (say, in the same city) would appear only in the BGP tables of these two ASes. A peer-peer edge between two large ISPs would not be visible from a third ISP that peers with both of them. Collecting BGP data from a large number of vantage points provides a more complete view of the AS graph and the set of AS paths, but it is difficult (if not impossible) to quantify the extent of coverage.

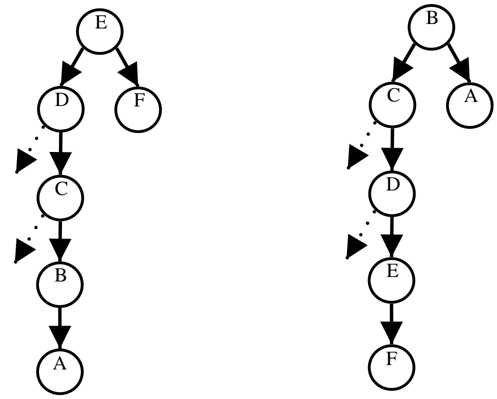

Fig. 2. (a) Partial view from AS E; (b) Partial view from AS B

Verifying the results of the inference algorithm is difficult without a complete and accurate repository of the relationships between ASes. Four main approaches to verification are possible, each with its own limitations. First, the number of invalid paths provides a rough measure of the success of the algorithm. The invalid paths correspond to cases where the inference algorithm has misclassified certain edges, or certain ASes have unusual relationships. Second, the output of the algorithm can be compared with the results of other algorithms on the same input data. Third, the inferences can be compared with the routing policies archived in the Routing Arbiter Database (RADB) [19]; however, the information in the RADB can be incomplete or out-of-date. Fourth, the inferences can be compared with proprietary data about the routing policies and commercial relationships of individual ASes. We follow the first approach to analyze the results of our algorithm in Section IV-C and plan to consider the other approaches in our future work.

III. Inferring AS Relationships

Our algorithm for inferring AS relationships exploits the structure of partial views of the AS graph as seen from different locations in the Internet. First this section describes the properties of the graph seen from a single vantage point. Next we present a reverse-pruning algorithm that assigns a rank to each AS for each of the partial views. Then, we infer the relationship between two ASes by comparing their vectors of ranks. Rather than simply combining the data from different vantage points into a single graph, our algorithm exploits each unique perspective to help us infer the AS relationships.

```
G = G_X;
r = 1;
while (leaves(G) ≠ φ) {
      for all u ∈ leaves(G)
            rank(u) = r;
      v' = v(G) − leaves(G);
      r = r + 1;
      G = G_{v'};
}
for all u ∈ v(G)
      set rank(u) = r;
```

Fig. 3. Reverse pruning algorithm on graph G_X

A. A Partial View of the AS Graph

Consider routing data from a single vantage point E from the example in Figure 1. The routing data provides a set of paths that can be used to construct a directed graph rooted at E. For example, the paths (E, D, C, B, A), (E, F), and (E, D, C) would result in the graph in Figure 2(a). The key to inferring the AS relationships is to identify the boundary point between the uphill and downhill portions of the three paths.

In practice, the Internet consists of a relatively small number of large Internet Service Providers (ISPs) and a large number of smaller ASes. The routes seen by a small AS near the edge of the Internet should have roughly equal uphill and and downhill portions of non-zero lengths. A small AS must traverse one or more upstream providers to reach most of the many other small ASes, so the uphill portions of the routes should have a less diverse collection of edges than the downhill portions. For large ISPs in the core of the Internet, the routes consist mainly of downhill portions. In either case, we expect a large portion of the edges in the partial view of the AS graph to fall in a large, acyclic portion consisting of provider-customer edges. The remaining edges (if any) should fall into a connected component near the source node.

As such, a leaf node in the graph is likely to be a customer of its parent node(s). For example, in Figure 2(a), F is likely to be a customer of E and A is likely to be a customer of B. We exploit this property of partial views by successively pruning the leaf nodes and assigning ranks to ASes.

B. AS Ranking

Our algorithm assigns a rank to each AS for each vantage point. Let X denote the source AS of a particular view of the AS graph and let $P(X)$ denote the set of AS paths seen from X. Since each path $p \in P(X)$ consists of a sequence of nodes starting with X, we construct a directed graph G_X rooted at X from $P(X)$. We let $v(G_X)$ denote the set of all vertices in G_X and let $leaves(G_X) \subset v(G_X)$ denote the leaves of the graph. We assign a ranking $rank(u)$ to each vertex $u \in v(G_X)$ by applying the reverse pruning algorithm in Figure 3. At each stage, the algorithm identifies the leaf nodes, assigns them a rank, and removes these nodes (and their incident edges) from the graph. In the end, the remaining nodes (if any) form the connected component of the original graph G_X; these nodes are all assigned the same (highest) rank.

In the example in Figure 2(a), F (rank 1) is a customer of E (rank 5), A (rank 1) is a customer of B (rank 2) and so on. B and F have no direct relationship regardless of their rank because they do not share an edge. When considering a partial view from a tier-1 AS that does not have any upstream providers, every path consists of zero or one peer-peer edges followed by a downhill portion consisting of provider-customer edges. In practice, we expect the provider-customer relationship to be acyclic [17]. For example, if in Figure 1 there is an additional edge between A and C, then since A is a customer of B and B is a customer of C, C cannot be a customer of A but is rather an additional provider to A. Hence, the partial view from a tier-1 AS would tend to be acyclic. In this case, successive pruning would identify provider-customer relationships. However, in other scenarios, the graph may have cycles. For example, consider the plausible scenario of Figure 1 but where E is a customer of both D and C. A partial view from E can contain paths (E, D, C, B, A), (E, F), and (E, C, D). The resulting graph has a cycle between nodes C and D. As such, it is difficult to infer the relationships between C, D, and E. We exploit this observation in our algorithm by assigning the same rank to all of the ASes in the remaining graph with no leaves. Information from other vantage points is necessary to construct an inference for these ASes.

C. Multiple Vantage Points

If we continue pruning in Figure 2(a), the eventual leaf C will be inferred as a customer of D, even though the two ASes have a peer-peer relationship. Identifying the boundary point between the uphill and downhill portions of a path is tricky. The structure of the partial view of the AS graph depends on the position of the AS in the Internet hierarchy. In Figure 1, the boundary is between C and D (the peer-peer relationship), not at E as suggested by this partial view in Figure 2(a). Now consider the view from AS B in Figure 2(b). This view confirms that A is a customer of B and F is a customer of E. However, the graph contradicts the previous view in that D is a customer of C. Clearly D and C cannot be customers of each other. This contradiction suggests that the two ASes may have a peer-peer relationship.

Another reason for having multiple views arises when two ASes share a link in some but not all the views. In this scenario, our algorithm imposes a relative rank for these two ASes in a partial view even though they may not share an edge from this source's perspective. Consider the topology in Figure 1 with the addition of a peering link between B and E. Consider a partial view from C that has paths (C, B, A) and (C, D, E, F) but no paths that use the (B, E) edge. Our algorithm assigns F and A a rank of 1, B and E a rank of 2. Despite the fact that the edge (B, E) does not appear in G_C, we may be able to exploit the presence of both nodes in $v(G_C)$ in conjunction with the ranking from other vantage points that do include the edge to draw inferences about the relationship between B and E. If B and E have the same rank in the views without the (B, E) edge, they are more likely to be inferred as peers.

D. Inference Rules for the ToR problem

From our ranking algorithm, we make the following observation: If ASes A and B share a relationship and the rank of A

is more than B from a given vantage point X, then A *appears* to be the provider of B from X's perspective. However, if A and B have the same rank, then they are indistinguishable from X's perspective. We exploit this observation in designing our inference rules described below.

Given data from N vantage points, we map each AS into an N-dimensional vector $(r_{i1}, \ldots, r_{iN})$, where r_{ij} is the rank of AS i from vantage point j. Let $l(i,j)$ be the number of coordinates k where $r_{ik} > r_{jk}$ and $e(i,j)$ be the number of coordinates k where $r_{ik} = r_{jk}$, for all $k = 1, 2, \ldots, N$.

1) Inferring Peer-Peer Relationships: We use the Equivalence rule below to identify peer-peer edges that are visible from many views. An AS relationship may not be visible from some partial views because some ASes may assign a low preference to paths that traverse this edge. We use the Probabilistic Equivalence rule to find peering edges where the relationship between two ASes is not visible from many partial views.

- **Equivalence** Two ASes i and j are said to be equivalent if $e(i,j) > N/2$. This rule considers two ASes that have the same rank in more than half the vantage points. If these ASes share an edge, they are likely to be peers.
- **Probabilistic Equivalence** Two ASes are probably equivalent if $\frac{1}{\delta_1} \leq \frac{l(i,j)}{l(j,i)} \leq \delta_1$ for a δ_1 close to 1. We use this rule to infer peering relationships between ASes when visibility is poor across the partial views. For our experiments, N is 10 and δ_1 is 2.

2) Inferring Provider-Customer Relationships: We use the Dominance rule to determine if an edge between two ASes is a provider-customer relationship because one AS tends to have a higher rank than the other in many of the partial views. Typically, in graphs from the vantage point of j or its customers, it is probable that $rank(j) > rank(i)$ even if i is a provider of j. To avoid an incorrect inference in such cases, we use the Probabilistic Dominance rule.

- **Dominance** An AS i is said to dominate AS j if $l(i,j) \geq N/2$ and $l(j,i) = 0$. If i dominates j, then we can infer that i is the provider of j, if the two ASes share an edge.
- **Probabilistic Dominance** If $\frac{l(i,j)}{l(j,i)} > \delta_0$ for a high value of δ_0 then i probably dominates j, and thus i is a provider of j. δ_0 should be greater than δ_1. We use a value of 3 for δ_0 in our experiments.

The orthogonal Equivalence and Dominance rules infer peer-peer and provider-customer relationships with a high degree of confidence. We apply those rules first in our inference algorithm, followed by the two probabilistic rules. Those AS relationships which are not inferred using these rules have the value of $max(\frac{l(i,j)}{l(j,i)}, \frac{l(j,i)}{l(i,j)})$ between δ_1 and δ_0.

IV. Experimental Results

This section evaluates our inference techniques on a collection of ten publicly-available BGP routing tables. We classify the relationships between ASes and identify a small number of AS paths that are inconsistent with the relationship assignment. The most common anomalies seem to stem from recent acquisitions and mergers, suggesting that some AS pairs may have a sibling relationship.

TABLE I

Telnet Looking Glass Servers

AS #	Name	# Edges
1	Genuity	13419
1740	CERFnet	14287
3549	Globalcrossing	13542
3582	University of Oregon	23136
3967	Exodus Comm.	19005
4197	Global Online Japan	13474
5388	Energis Squared	13534
7018	AT&T	14160
8220	COLT Internet	11282
8709	Exodus, Europe	15519

A. BGP Routing Table Data

Our inference techniques have been applied to a collection of ten BGP routing tables available from Telnet Looking Glass servers. We automated the process of contacting each server, sending "`show ip bgp`" to the command-line interface, and storing the resulting table. For each destination prefix the table has one or more routes with a variety of BGP attributes, including the AS path. We extract the best and alternate paths for each prefix and construct a list of all AS paths that appear in the table. For each path, we add the AS number of the router to the beginning of each path and remove duplicate AS numbers that arise from AS prepending. Then we process the paths to construct a partial view of the AS graph. After constructing the partial views, we apply the ranking algorithm and inference rules from Section III to assign a relationship to each AS pair that shares an edge in one or more of the routing tables.

Table I provides a summary of the ten tables we downloaded on April 18, 2001. The "# Edges" column shows the number of unidirectional edges in the AS paths. The entry for AS 3582 corresponds to the University of Oregon RouteViews server, which has 52 peering sessions with 39 different ASes [11]. The RouteViews server has an especially rich view of the AS graph, with over 23,000 edges compared to 11,000–15,000 edges for most of the other routing tables. In total, the AS paths in the ten routing tables have 24,752 unidirectional edges between 24,059 pairs of ASes.

We use the partial views from these ten routing tables to generate our inferences of the AS relationships. In Section IV-C, we validate our inferences using the AS paths from another collection of routing tables. We manually downloaded routing data from Ebone (AS #1755), MAE-West (AS #2548), KDDI Japan (AS #2516), and Cable and Wireless (AS #6893) on April 9, 2001. These four tables are available from Web Looking Glass servers that have a slightly different interface than the Telnet servers. The Web interface typically does not permit users to invoke the "`show ip bgp`" command. Instead, we rely on the "`bgp paths`" command that produces a list of AS paths, without the destination prefix or an indication of the best path. As with the "`show ip bgp`" data, we extract the AS paths, add the AS number of the source AS, and remove duplicate ASes. Next we use the results of our inference algorithm to assign a relationship to each unidirectional edge. Then, we look for paths that violate the two patterns identified in Section II.

TABLE II

INFERRED RELATIONSHIPS FOR 23,935 AS PAIRS

Relationship	# AS pairs	Percentage
Provider-customer	22,621	94.51%
Peer-peer	1,136	4.75%
Unknown	178	0.74%

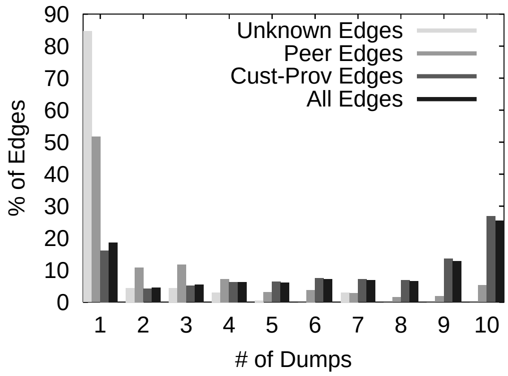

Fig. 4. Percentage of edges that appear in x of the ten tables

B. Relationship Inferences

Table II summarizes the results from applying our inference algorithm to the ten BGP tables from Table I. Our algorithm produces an inference for over 99.2% of the edges in our AS graph ($23,757$ of the $23,935$ AS pairs). The vast majority of AS pairs appear to have a provider-customer relationship. Approximately 5% of the AS pairs have a peer-peer relationship. Table III highlights the role of the various inference rules in drawing conclusions about AS relationships. A large percentage of the provider-customer relationships are inferred from the Dominance rule. Dominance in N dimensions is a good indication of a provider-customer relationship and we can be reasonably certain of 95.57% of our provider-customer inferences. Similarly, close to 75% of the peering links are inferred from the Equivalence condition. The probabilistic rules account for 4.43% of the provider-customer inferences and 25.26% of the peer-peer inferences.

In Figure 4, we show the frequency of occurrence of edges in the BGP routing tables. We see that about 25% of all edges appear in all ten tables while about 15% appear in only one table. A similar distribution exists for the customer-provider subset. Almost all of the "unknown" edges appear in only one table. These edges may correspond to AS pairs with sibling or backup relationships. Similarly, many of the edges that we determined to be peer-peer edges appear in only one table. This is likely due to the peer export policy we described in Section II-A. If routes learned from one peer are not distributed to other peers, and if most peer-peer edges exist in the "dense core" as we show later, and if the dense core is almost a clique, it thus implies that most peer-peer edges will be visible from only one view.

The percentages of provider-customer and peer-peer relationships in Table II are consistent with the conclusions of Lixin Gao in [10]. Our inference that 4.75% of the AS pairs (1,136 pairs) are peers is close to Gao's values between 5.3% and

TABLE III

DISTRIBUTION OF THE 23,757 INFERENCES

Rule	Number	Percentage
Complete dominance	21,618	95.57%
Probabilistic dominance	1,003	4.43%
Equivalence	849	74.74%
Probabilistic equivalence	287	25.26%

7.8%. The percentage of provider-customer relationships we infer is within 1–1.5% of the figure reported in [10]. Her study drew on RouteViews data from September 1999, January 2000, and March 2000. The number of edges in the RouteViews dump has grown by over 70% over the last 13 months. With the larger RouteViews table and the nine other tables, our collection of edges is twice as large as the graph used in her earlier study. Using traceroutes from 16 sources to $400,000$ destinations [8] in October 2000, CAIDA constructed an AS graph that is slightly larger than ours. Their final graph consists of $7,563$ ASes and $25,005$ edges. Ours contains $10,698$ ASes and $23,935$ AS pairs. However, they do not explore this graph in terms of AS relationships.

C. Validation of Inferences

Since the peering and customer information of an ISP are proprietary information, we cannot validate our inferences against an official list of AS relationships. Instead, we determine what percentage of the AS paths actually adhere to the export rules suggested by our inferences. There are two scenarios where we may label an AS path as an anomaly: some AS in the path actually violates the export rules, or our inference of one of the edges in the path is wrong. The percentage error that is reported in this section is the sum total of these two scenarios. For our validation, we draw on the list of AS paths from two of the ten of the Telnet servers (AS numbers 1 and 7018) used to construct our original inferences, as well as the four Web servers (AS numbers 1755, 2516, 2548, and 6893).

If every AS pair has a customer-provider or peer-peer relationships, then every AS path should have one of the two patterns identified in Theorem 1. A path is an anomaly if it has any two adjacent edges having one of the following patterns:

1) $(+1\ -1)$: An AS permits transit traffic between two of its providers.
2) $(+1\ 0)$: An AS permits transit traffic from one of its peers to one of its providers.
3) $(0\ -1)$: An AS permits transit traffic from one of its providers to one of its peers.
4) $(0\ 0)$: An AS permits transit traffic between two of its peers.

Case 1 represents a serious violation of the export rules. This anomaly may arise from a misconfigured customer or due to a misclassified relationship where the customer AS is actually a sibling of one or both of the providers. Backup and sibling relationships can cause case 2 and case 3 anomalies. Case 4 suggests that the path traverses two consecutive peering links, which may be permissible if the two peers have a sibling relationship for some destination prefixes. Detecting these anomaly

TABLE IV

QUANTIFICATION & DISTRIBUTION OF PATH ANOMALIES

AS #	AS Name	# of Paths	Anomaly Paths	Anomaly %	Unique Anomalies	Case 1	Case 2	Case 3	Case 4	Popular Anomaly
1	Genuity	65,383	421	0.65%	83	25	58	0	155	(1 7176 8938)
7018	AT&T	141,283	889	0.63%	82	17	64	1	185	(8297 1290 174)
6893	CW	70,253	2,050	2.92%	135	27	98	10	212	(3561 5400 5727)
2548	MaeWest	115,199	1718	1.49%	209	59	145	5	245	(1239 8043 6395)
1755	Ebone	23,469	678	2.89%	116	23	86	7	207	(3300 8933 2200)
2516	KDDI	126,414	10,927	8.67%	555	262	260	33	1051	(209 1800 1239)

paths provides a way to identify AS pairs that may have more complicated relationships.

As shown in Table IV, the vast majority of paths are consistent with the relationship assignments and the associated export policies. The percentage of anomalies varies between 0.6–3.0% for five of the six routing tables. These results validate our base assumption that the export rules are observed by a large percentage of the ASes. However, KDDI (AS #2516) has a relatively high percentage of anomalous paths (8.7%). For every anomalous path, we can identify an anomaly pattern consisting of three adjacent ASes (A, B, C) where the pair of edges (A, B) and (B, C) falls into one of four cases. The results in Table IV show that case 3 anomalies are very uncommon and case 1 arises less frequently than case 2 and case 4. KDDI exhibits an unusually high number of all four cases (especially case 4); further investigation is necessary to explain this fact. A small number of AS triples (A, B, C) are responsible for the vast majority of the anomalies. For most of the routing tables, ten different AS triples were responsible for more than 90% of the anomalous paths.

D. *Common Anomaly Patterns*

The last column in Table IV lists one popular triple for each routing table dump. We analyzed the popular anomaly patterns using the RADB *whois* data [19] which identifies ASes by name and sometimes includes a list of import and export policies. We observed that many popular anomalies occur due to sibling relationships between two ASes under the same administration. For example, in the anomaly pattern (1 7176 8938), Genuity Europe (7176) and Genuity (1) exhibit a sibling behavior. This is also true with the anomaly pattern (8297 1290 174) with Teleglobe Europe (7018), PSINet UK (1290), and PSINet (174). The anomaly patterns (1239 1740 7018) and (3561 5400 5727) seem to have similar explanations; Cerfnet (1740) was acquired by AT&T (7018), and AS 5400 and AS 5727 are both part of the Concert IP backbone. For the anomaly pattern (1239 8043 6395), further investigation showed that IXC Communications acquired SmartNAP (8043) [20] and IXC was later renamed as Broadwing (6395). Similar anecdotes apply to many of the other popular anomaly patterns.

Identifying anomaly patterns may be a useful way to detect sibling relationships. In the absence of misconfigurations, we can label all case 1 anomalies as caused by sibling relationships. That is, if the AS path (A, B, C) is a case 1 anomaly, either A and B or C and B are siblings. We do not extend this to case 2 or case 3 anomaly patterns since these anomaly patterns may represent backup relationships or other complex transit agreements. Ignoring the KDDI dump, we observed 110 unique case 1 anomaly patterns. In these 110 patterns, we found 196 unique AS pairs with possible sibling relationships; 18 of these possible sibling relationships appeared in multiple paths. As an example, AS 2685 (AT&T Global Network Services) appears in the middle as a customer in many case 1 anomalies. This AS may have a sibling relationship with AS 7018 (AT&T). Our sibling inferences account for roughly 0.8% of the edges in the AS graph. The work by Gao [10] identifies 1.5% of the edges to be siblings; her validation of a subset of these inferences on a private data set found that 20% of these inferences were valid. We plan to explore our approach for detecting sibling relationships in more detail as part of future work.

V. INTERNET HIERARCHY

In addition to inferring the relationship between AS pairs, it is useful to identify the position of each AS in the Internet hierarchy. Previous work has classified ASes based on node degree [9]; ASes with a large number of neighbors are placed above ASes with small node degree. However, a simple degree-based approach may not capture the essence of the *tiers* in the hierarchy. Instead, we classify ASes based on the commercial relationships derived in the previous section. Typically, a customer should be at a lower level in the hierarchy than its provider(s). We represent the AS topology as a directed graph where the direction of an edge indicates the type of relationship between the two ASes. In our graph, a provider-customer relationship between A and B is represented by a directed edge from A to B and a peering relationship between A and B is represented by two directed edges, one from A to B and the other from B to A. Such a graph representation has also been independently proposed in [15]. An important difference between our approaches is the procedure used for determining the Internet hierarchy. The work in [15] maps the Internet topology into a strict hierarchy based on provider-customer edges while our classification also uses the distribution of peering links to identify the top levels of the hierarchy.

A. *Customers and Small Regional ISPs*

Customers are the easiest class of ASes that can be classified from this directed graph structure of the AS topology. Customers are those stub networks which are origins and sinks of traffic and which do not carry any transit traffic. From the very definition of the direction of edges in our graph, we can infer the customer ASes to be the leaves of this directed graph. In a directed graph, a leaf is a node with out-degree 0. Since an undirected graph makes no distinction between out-degree and in-degree, customers with multiple providers would have a degree more than 1 and hence would not appear as leaves of the graph. Modeling the topology as a directed graph provides a more precise characterization of the bottom-most level in the AS hierarchy. In the directed graph constructed from the ten BGP dumps, $8,898$ of the $10,915$ ASes are leaf nodes. The rest of the graph contains just 18.5% of the ASes.

Once we identify the customers and remove these nodes, the resulting graph has a new set of leaves. These leaves represent small regional ISPs that have one or more customers. We can continue the process of pruning the leaves of the graph until we reach a point where the graph has no leaves. This involves applying a reverse pruning algorithm similar to Figure 3 in Section III-B. We define the set of nodes removed by this process as *small regional ISPs*. Since every peering relationship is represented as a loop of two edges in the graph, no ASes with peering relationships are included in this level. Applying the reverse pruning algorithm to our graph reveals 971 small regional ISPs. We define the remainder of the graph as the *core*, consisting of a connected component with just 1046 ASes and 6249 unidirectional edges. This represents approximately 25% of the total number of edges in the graph. The nodes in the core have an average degree of 6.

B. Dense Core

The set of ASes that remain after the pruning process represent the *core* of the Internet. Given the nature of the reverse pruning process, we can infer that for every AS present in the core, all of its peers and its provider should also be present in the core. The core of the graph should include the small number of so-called *tier-1* providers. In practice, the term "tier-1 provider" is loosely defined as a "large" AS or as an AS that does not have any upstream provider. We could identify these ASes by looking for all provider-free nodes. However, this approach would be sensitive to a small number of missing edges or misclassified relationships in our AS graph. From our BGP tables and relationship inferences, there are 98 ASes with no provider. Such a large number of ASes unlikely form today's Internet dense core. This list includes small ISPs such as CCP Online and HutchCITY. Instead, we could exploit the observation that every provider-free AS would peer with every other provider-free AS to ensure reachability to all destinations. That is, the set of tier-1 ASes should form a clique where every AS has an edge to and from each of the other ASes. Other provider-free ASes, if they exist in our graph, would be excluded from the set of tier-1 providers.

In practice, some ASes may have complex transit or backup relationships to provide connectivity. We define a weaker notion of the *dense core* as the largest subset of ASes whose induced subgraph is "almost a clique." We define a directed graph of N nodes to be *dense* if every node in the graph has an in-degree and out-degree of at least $N/2$. $N/2$ is the smallest value of a degree of every node for which we can guarantee that the shortest path between any two nodes in the graph is at most 2 (irrespective of the graph's structure). This can ensure that two ASes in the dense core which do not share a peering

compute $z \in v(G)$ with maximum out-degree;
$X = \{z\}$;
$pos(z) = 1; r = 1$;
while ($X \neq v(G)$) {
    compute $y \in v(G) - X$ with max $d(y, X)$
        (selecting the y with the max out-degree)
    $X = X \cup \{y\}$;
    $maxindegree(r) = d(y, X)$;
    $r = r + 1$;
    $pos(y) = r$;
}

Fig. 5. Greedy heuristic to order the nodes

relationship can potentially peer *indirectly* through some other intermediary AS in the dense core (the intermediary AS in the path of length 2). The problem of determining the largest clique in a graph is NP-hard. Given that a clique is just one example of a dense graph, the problem of finding the largest dense subgraph of a graph becomes much harder. We have developed a greedy heuristic for identifying the ASes in the dense core.

1) Identifying the Dense Core: First, we order the vertices based on a "greedy" notion of connectivity, following the heuristic in Figure 5. Let G represent the directed graph representation of the core. Let $v(G)$ and $E(G)$ represent the vertices and edges of the graph G. Let $d(x, Y)$ for $x \in v(G)$ and $Y \subset v(G)$ denote the number of edges of the form (x, z) where $z \in Y$. Connectivity from a node to a given set of nodes refer to the number of directed edges from that node to any of the nodes in the set. Assume that k of the N nodes are already ordered. For each of the remaining $N - k$ nodes, we determine the connectivity to the k nodes and pick the node with the maximum connectivity as the $(k+1)^{th}$. When multiple nodes have the same connectivity, we choose the node with a higher out-degree. In Figure 5, $pos(x)$ denotes the position of a node x in the final ordering.

Let x_i denote the i^{th} AS in the ordering and X_i be the set of the top i ASes. Let $conn(i)$ represent the connectivity of x_i which is equal to $d(x_i, X_{i-1})$. We define the dense core as the set X_k for the smallest value of k such that $conn(k+1) < (k+1)/2$ and X_k is dense. Once the value of $conn(k+1)$ falls below the value $(k+1)/2$, the $(k+1)^{th}$ node will violate the *dense* property. Therefore if $conn(k+1) < (k+1)/2$, the induced subgraph of X_{k+1} will not be dense since the out-degree of x_{k+1} will be less than $(k+1)/2$. However this does not mean that if $conn(k+1) > (k+1)/2$, then X_{k+1} is dense. Consider the scenario where a node x_j for some $j < k$ is linked to more than $j/2$ elements in X_j and not linked to any node in $X_k - X_j$. This is an example where $conn(k) > k/2$ but X_k is not dense. In this regard, our heuristic is greedy. For the AS topology that we obtained, the point where $conn(k)$ dropped below $k/2$ was the first value of k for which X_k was not dense. This indicates that the ordering output by the heuristic was indeed a good ordering for choosing the vertices of the dense core. In other words, it validates the rationale behind our greedy approach that if y appeared before z in the ordering then y had a better chance of being present in the dense core than z.

TABLE V

ASES IN THE DENSE CORE

AS#	AS Name	AS#	AS Name	AS#	AS Name
1	Genuity	174	PSINet	1239	SprintLink
1755	Ebone	4200	Telia	1833	TeliaNet
209	Qwest	2548	Digex	6453	Teleglobe
2914	Verio	3356	Level 3	3549	GlobalCrossing
3967	Exodus	4006	Cogent	3561	Cable&Wireless
2828	XO	701	UUNet	5511	FranceTelecom
8918	Carrier 1			7018	AT&T WorldNet

2) Properties of the Dense Core: Applying this heuristic to the core of our graph, we identify a dense core consisting of 20 ASes. These ASes include the large ISPs such as Genuity, Sprint, AT&T, and UUNet, as shown in Table V. The top 20 ASes have a very dense connectivity of 312 peering links. The top 15 of the 20 ASes almost form a clique with only three edges missing from the clique. The largest clique we observed in this innermost core consisted of 13 ASes. The 20 ASes have $6,732$ provider-customer edges to customer ASes and 958 provider-customer edges to the small regional providers. After removing the dense core, the remainder of the core consists of 1026 ASes.

We also obtained routing table dumps on 22 September 2001[1]. The new dense core adds KDDI, another AT&T AS, Reach and Korea Telecom, while removing Ebone, both Telia ASes, Cogent, UUNet and Carrier1. The dense core for 20 December 2001 adds TeliaNet, AOL, and Tiscali, while removing Digex, France Telecom, one AT&T AS, and Korea Telecom. Most of the additions and removals came from or went to the transit core.

C. Transit Core

After removing the dense core, we noticed the presence of other large national providers and hosting companies that have peering relationships with many of the ASes in the dense core. To identify these ASes, we define the notion of a *transit core*. Nodes in the transit core peer with each other and with ASes in the dense core, but they do not tend to peer with many other ASes. In our directed graph representation, these peering links are essentially the incoming directed edges from vertices outside this set to vertices within the set. We define such a set of edges to be the *in-way cut* of the graph induced by the given set of vertices. Using this property, we define the transit core as the smallest set of ASes containing the dense core which induces a weak in-way cut, that is, one having a small number of edges compared to the total number of ASes in the transit core.

1) Identifying the Transit Core: Given $X \subset v(G)$, let $cut_{in}(X)$ denote the set of all edges of the form (y, z) where $y \in v(G) - X$ and $z \in X$. We define a cut X of the vertex set $v(G)$ to be a weak cut if $|cut_{in}(X)| < |X|/2$. The problem of finding weak cuts in a graph is NP-complete and no good approximation algorithms are known for that problem. Given that the transit core is a superset of the dense core and that the dense core is derived by the greedy ordering, we apply the same ordering to find the transit core as was used to find the dense core. A natural way of using this ordering to find the transit core is to find the smallest value of k such that $|cut_{in}(X_k)| < k/2$. Surprisingly we found that the value of k at which $|cut_{in}(X_k)| < k/2$ also satisfied the property that $conn(k+1) = 1$. This means that no two edges in $cut_{in}(X_k)$ have the same source. A weak cut also means that more than 50% of the ASes in X_k do not have any peering relationship with any of the ASes in $v(G) - X_k$. Hence by this definition, X_k should indeed contain all the transit providers.

2) Properties of the Transit Core: Applying the in-way cut algorithm to our graph, we discover a transit core consisting of 129 ASes. These 129 ASes have 183 peering links with the ASes in the dense core. Concert, Singapore Telecommunications, UUNet European division, Teleglobe European division and KDDI Japan are some example ISPs in our transit core. We found many of the top providers in Europe and Asia to be present in our transit core.

D. Outer Core

We classify all of the remaining ASes in the core as the *outer core*. The members of the outer core typically represent regional ISPs which have a few customer ASes and a few peering relationships with other such regional ISPs. The outer core consists of 897 ASes that have 29 peering sessions with ASes in the dense core and 145 peering sessions with ASes in the transit core. We observed that many members of our outer core are regional ISPs. Some examples include Turkish Telecomm, Williams Communications Group, CAIS Internet, Southwestern Bell Internet Services and Minnesota Regional Network. It is interesting to note that while Exodus Communications (AS 4197) is present in our outer core, Exodus.net (AS 3967) is present in the dense core.

E. Summary

TABLE VI

DISTRIBUTION OF ASES IN THE HIERARCHY

Level	# of ASes
Dense core (0)	20
Transit core (1)	129
Outer core (2)	897
Small regional ISPs (3)	971
Customers (4)	8898

Table VI summarizes the number of ASes at each level in the hierarchy—dense core (level 0), transit core (level 1), outer core (level 2), small regional ISPs (level 3), and customers (level 4). Table VII summarizes the connectivity between various levels in the AS hierarchy. Each number in the table is the total number of edges from one level to another. For example, 626 is the total number of edges from level 0 to level 1. The tables shows several key properties of the Internet topology:

- The ASes in dense core are very well connected.
- As we move from the dense core toward customers, the inter-level and intra-level connectivity drops significantly.

[1] The same set of ASes in Table I were used, except for the CERFnet looking glass server which has been offline.

TABLE VII

INTER-CONNECTIVITY ACROSS LEVELS

Level	0	1	2	3	4
0	312	626	1091	958	6732
1	183	850	1413	665	3373
2	29	145	1600	543	3752
3	0	0	0	212	2409

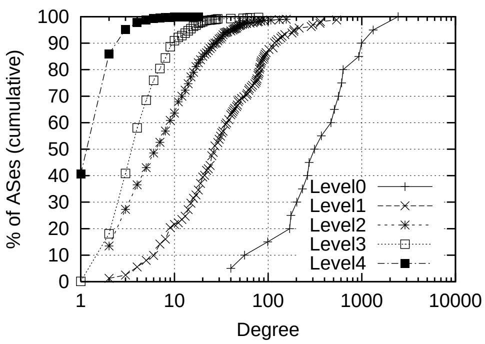

Fig. 6. Cumulative distribution of AS degree by level

- The large number of customer ASes have their providers distributed across all the levels. The ASes in level 0 support a large number of customer ASes. This indicates that the connectivity across levels is not strictly hierarchical, as also observed in [9].
- The number of edges within the outer core is less than the total number of vertices in the outer core. This indicates the presence of multiple disconnected groups of ASes in the outer core; ASes in different groups communicate via ASes in the dense core and the transit core.

The graph in Figure 6 explores the relationship between node degree and the levels in the hierarchy. We define node degree as the number of neighboring ASes without regard to the relationship. The graph plots the cumulative distribution of node degree on a logarithmic scale. In general, level 0 and 1 ASes have high degree, and level 3 and 4 ASes tend to have low degree. However, this is not universally true. Some customers at level 4 have a large number of upstream providers, and some ASes in the dense core at level 0 have a relatively small number of neighbors. For example, our results suggest that AS 1833 (TeliaNet USA) has a degree of only 40. Yet, we classify TeliaNet as part of the dense core due to its rich collection of peering relationships. A hierarchy based solely on degree distribution would not be able to make this distinction.

VI. CONCLUSIONS

The relationships between ASes has a significant impact on the flow of traffic through the Internet. Our work makes two important contributions toward understanding the structure of the Internet in terms of these relationships:

- An algorithm for inferring AS relationships from partial views of the AS graph from different vantage points
- A mechanism for dividing the Internet hierarchy into levels based on AS relationships and node connectivity

The complete structure of the Internet is unknown and difficult, if not impossible, to obtain. Our approach is comprised of many heuristics, with certain limitations:

- We draw our inferences based on only ten vantage points available from Telnet Looking Glass servers. Ideally, we would have a larger collection of routing tables from more diverse vantage points, including smaller customer ASes.
- We treat the RouteViews routing table as a view from a single AS. In future work, we plan to extract a separate view for each AS participating in the RouteViews project.
- Multiple ASes may fall under the administrative control of a single institution, due to historical artifacts and market forces. We plan to extend our methodology to incorporate more complex routing policies that are not captured by the traditional customer-provider and peer-peer relationship.

Despite these limitations, we have shown that our approach provides a detailed view of the Internet topology in terms of the relationships between ASes.

ACKNOWLEDGMENTS

We are grateful to the anonymous reviewers for helping us improve the paper. We would also like to thank everyone who has made their BGP routing tables available to the research community.

REFERENCES

[1] John W. Stewart, *BGP4: Inter-Domain Routing in the Internet*, Addison-Wesley, 1998.

[2] Sam Halabi and Danny McPherson, *Internet Routing Architectures*, Cisco Press, second edition, 2001.

[3] Geoff Huston, "Interconnection, peering, and settlements," in *Proc. INET*, June 1999.

[4] C. Alaettinoglu, "Scalable router configuration for the Internet," in *Proc. IEEE IC3N*, October 1996.

[5] B. Krishnamurthy and J. Wang, "On network-aware clustering of Web clients," in *Proc. ACM SIGCOMM*, August/September 2000.

[6] R. Govindan and H. Tangmunarunkit, "Heuristics for Internet map discovery," in *Proc. IEEE INFOCOM*, March 2000.

[7] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "Network topologies, power laws, and hierarchy," Tech. Rep. 01-746, Computer Science, University of Southern California, June 2001.

[8] "CAIDA Web Site," `http://www.caida.org`.

[9] R. Govindan and A. Reddy, "An analysis of inter-domain topology and route stability," *Proc. IEEE INFOCOM*, April 1997.

[10] L. Gao, "On inferring automonous system relationships in the Internet," *IEEE/ACM Trans. Networking*, vol. 9, no. 6, December 2001.

[11] "University of Oregon RouteViews project," `http://www.routeviews.org/`.

[12] "BGP tables from the University of Oregon RouteViews Project," `http://moat.nlanr.net/AS/Data/`.

[13] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos, "On power-law relationships of the Internet topology," in *Proc. ACM SIGCOMM*, August/September 1999, pp. 251–262.

[14] C. Jin, Q. Chen, and S. Jamin, "Inet: Internet topology generator," Tech. Rep. CSE-TR-433-00, U. Michigan, September 2000.

[15] Z. Ge, D. Figueiredo, S. Jaiwal, and L. Gao, "On the hierarchical structure of the logical Internet graph," in *Proc. SPIE ITCOM*, August 2001.

[16] Lixin Gao, Timothy G. Griffin, and Jennifer Rexford, "Inherently safe backup routing with BGP," in *Proc. IEEE INFOCOM*, April 2001.

[17] L. Gao and J. Rexford, "Stable Internet routing without global coordination," *IEEE/ACM Trans. Networking*, vol. 9, no. 6, pp. 681–692, December 2001.

[18] H. Chang, R. Govindan, S. Jamin, S. Shenker, and W. Willinger, "On inferring AS-level connectivity from BGP routing tables," 2001.

[19] "RADB Whois Server," `whois.radb.net`.

[20] "IXC Buys Three ISP's," `http://www.thedigest.com/93/93-38.html`.