The Internet Dark Matter – on the Missing Links in the AS Connectivity Map

Rami Cohen
Computer Science Department
Technion, IIT
Email: ramic@cs.technion.ac.il

Danny Raz
Computer Science Department
Technion, IIT
Email: danny@cs.technion.ac.il

Abstract— The topological structure of the Internet infrastructure is an important and interesting subject that attracted significant research attention in the last few years. Apart from the pure intellectual challenge of understanding a very big, complex, and ever evolving system, knowing the structure of the Internet topology is very important for developing and studying new protocols and algorithms. Starting with the fundamental work of Falostous et. al, a considerable amount of work was done recently in this field, improving our knowledge and understanding of the Internet structure. However, one basic problem is still unanswered: how big is the Internet. In the AS level this means: how many peering relations exist between ASs. Finding this number is hard since there is no direct way to retrieve information from all nodes regarding their direct neighbors, and all our knowledge is based on sampling processes. Thus, it is very difficult to characterize the Internet since it may well be the case that this characterization is a result of the sampling process, and it does not hold for the "real" Internet.

In this paper we attack this problem by suggesting a novel usage of the measurements themselves in order to infer information regarding the whole system. In other words, in addition to looking at the overall graph that is generated from the union of the data obtained by performing many measurements, we consider the actual different measurements and the amount of new data obtained in each of them with respect to the previous collected data. Using the second moment allows us to reach conclusions regarding the structure of the system we are measuring, and in particular to estimate its total size. We present strong evidence to the fact that a considerable amount (at least 35%) of the links in the AS level are still to be unveiled. Our findings indicate that almost all these missing links are of type *peer-peer*, and we provide novel insight regarding the structure of the AS connectivity map with respect to the peering type.

I. INTRODUCTION

The topological structure of the Internet infrastructure is an important and interesting subject that attracted significant research attention in the last few years. Apart from the pure intellectual challenge of understanding a very big, complex, and ever evolving system, built by people and used by many more, knowing the structure of the Internet topology is very important for developing and studying new protocols and algorithms.

The work in this area composed of collecting information regarding the current (and possibly past) state of the Internet [1], [2], [3], inferring from the collected data the actual topological structure [4], [5], and analyzing this structure in order to understand the inherently important characteristic and evolution of the system [6], [7], [8], [9]. One problem that arises in this context is that it is impossible to measure the full structure of the Internet directly and to obtain the full connectivity graph. This is not only due to the fact that the Internet is too big and ever evolving, but largely because there is no direct way to retrieve information from each node regarding its direct neighbors. Thus, it is very difficult to characterize the Internet since it may well be the case that this characterization is a result of the sampling process, and it does not hold for the "real" Internet. In other words, it is extremely difficult to know if the picture we have is a good approximation of the "real" Internet connectivity, or if it is biased to a large extend by the measurements, and thus does not reflect the "true" picture.

In this paper we attack this problem by suggesting a novel usage of the measurements themselves in order to infer information regarding the whole system. In other words, rather than looking at the overall graph that is generated from the union of the data obtained by performing many measurements, we consider the actual different measurements and the amount of new data obtained in each of them with respect to the previous collected data. This technique allows us to reach conclusions regarding the structure of the system we are measuring. In our case we apply the methods to the Autonomous System (AS) level connectivity graph.

In Section II we start with a rigorous study of the different data collecting techniques that are used in order to collect AS connectivity information. We then use the characterization of the data collection to draw conclusions regarding the actual structure of the full AS map. We consider a new measure for graphs, the number of (policy-based) shortest path spanning trees needed to cover the edges of a graph and show that the AS map is unique in the sense that a considerable amount of links are not revealed in this covering process. We also show that these unrevealed peers are mostly of the type *peer-peer* while this process unveiled many of the *customer-provider* peers. We also analyze the routing policy of available database using concepts that have been presented in [7] and [6], and we settle a fundamental difference between the results that were presented in these works.

While the size of the full AS connectivity map is unknown, in Section III we show several methods to estimate this size. In particular, we use a data from several databases to approximate

the actual number of missing links and present a strong evidence to the fact that at least 35% of the links in the AS level are still to be unveiled.

In Section IV we examine the vertex degree distribution of the AS connectivity map. Using our inferences from Section II, we explain the difference between earlier results that show that the AS connectivity map follows the power law [4] and other results that question this observation [5], by showing that while the *customer-provider* subgraph follows the power law, the *peer-peer* subgraph behaves differently.

II. UNDERSTANDING THE AS DATA GATHERING PROCESS

The AS level connectivity map is modelled by a graph $G = \{V, E\}$. Each node in the graph represents an autonomous system, and an undirected edge represents a peering relation between two ASes. In order to formalize the methods used to gather information about the available peering relations in the Internet we use several simplifications and assumptions. This helps us create a rigorous view of the discovery process, and hopefully maintains the most important and relevant aspects of the discovery process while eliminating less important issues.

There are several methods to gather and sample information of the AS connectivity map. The first one is a BGP based database consisting of a set of BGP routing tables from a set of ASes. Each routing table contains the paths (in terms of ASes) to each of the relevant subnetworks. For simplicity, we assume that this data, the path vector of an AS, contains paths to all other ASes and not to specific subnetworks. Since most AS level routing do not distinguish between different networks within the same AS, this should not add a significant inaccuracy. Thus, the collection of all the path vectors from a given AS to all other ASes is a DAG (Directed Acyclic Graph). Retrieving the peering connectivity as reflected by the data of a specific AS is the most basic peering retrieval process. This can be done by a direct access to the BGP data, or via the *Looking Glass* tool[1].

One of the well appreciated advantages of BGP is its ability to use policy based routing where each AS defines its own local policy. In practice, the policy of an AS reflects the commercial relationship with other ASes. Thus, the AS connectivity graph has a hierarchical structure in which connected ASes have a *customer-provider* relationship if a small AS is connected to a larger AS, and they have a *peer-peer* relationship if they have comparable size [10], [11]. Moreover, permitted paths do not include so called *valleys* nor *steps* [8], [12]. Although ASes may use other policies and BGP routing table may reflect more than one route for a destination AS, we assume that under the above policy, routing is done along shortest paths and the information retrieved form a single AS is a tree. This assumption makes the discussion regarding the retrieval process formal and rigorous. Thus, one can now model the process of retrieving peering information by creating a policy based shortest path tree, namely a shortest path tree

that follows the policy guidelines presented above, from a given node to all the nodes in the graph.

The question of discovering peering relations translates now to the amount of edges covered by a union of such trees rooted at a given set of nodes. In other words, the amount of peering relations covered by a collection of BGP path vectors from a set of ASes, corresponds to the amount of edges covered by a set of the corresponding trees. In this work we use the Route-View project [1] as a source for our BGP database. This project collects a snapshot of the Internet AS level topology on a daily basis from about 40 ASes.

The second method to gather information is the Internet Routing Registry (IRR) [2]. This is a union of world-wide routing policy databases that use the Routing Policy Specification Language (RPSL) [13], [14]. These databases contain, among other things, the local connectivity information for the registered ASes. In terms of the AS connectivity graph, this corresponds to discovering all the edges connected to a given node and therefore these objects are referred to as stars. Obviously, if we had such a complete and an updated database we could easily derive the AS connectivity map. However, not all the ASes are willing to publish their peering relationship. Moreover, in some cases the entries in the database are out of date, thus they may fail to contain existing peers while in some other cases they may contain peering relationships that are no longer valid.

The most updated and complete IRR database is maintained by RIPE [15]. This is one of four Regional Internet Registries and during June 2005 it consisted of about 6800 ASes that have registered in Europe[2]. In this database almost all the registered ASes (over 98%) share their peering relationship. In contrast, only 400 ASes out of 11000 ASes that have registered in ARIN (American Registry for Internet Numbers) share this information.

As mentioned above, entries in the IRR may be invalid. Thus, one should use a filtering mechanism in order to remove these entries. In this work we use a filter that is based on the sanity checks that have been presented in [5]. These filters are based on the fact that a valid peer should appear in the entries of both ASes while a peer that appears in only one entry may be out of date and should be removed from the IRR. In case that one of the ASes does not have an entry in the IRR it is not clear whether this peer should be filtered out or not. Thus, we consider two filtered database, the first one contains these peers while in the second one the peers are removed.

DIMES [3] is a new project that samples the Internet using *traceroute*. In particular, distributed agents located in thousands hosts around the world, perform periodic *traceroute* to a set of IP addresses. In contrast to the other methods (i.e. BGP routing tables and IRR) this technique obtains information regarding the Internet connectivity in the router level. Nevertheless, one can correlate between an IP address

---

[1]A list of ASes that provide access to the *Looking Glass* tool can be found in www.traceroute.com.

[2]Actually there are almost 10000 registered ASes but over 3000 out of them seem to be inactive due to the fact that they do not appear in the Route-View database and their entries look invalid (see discussion regarding IRR filtering mechanism in the next paragraph).

and its corresponding AS (i.e. the AS that allocated this IP address) and therefore connectivity in the router level induces connectivity in the AS level. Currently (June 2005), DIMES consists of 3781 distributed agents[3] located in 77 countries around the world and it spans 39000 links from the AS map.

A. Covering graph by shortest path trees

In [5] the authors found that the available databases consisting of BGP routing data alone, are not complete enough and when adding the IRR database, a significant number of links are revealed. They explain this result by the fact that there are several paths to each AS while only one path is published, and by the private nature of *peer-peer* links. In [16] the authors pointed out that the majority of these new links, connect between so called *rich nodes* - nodes with large numbers of links. In this section we reconstruct the gathering process that is used to build the Route-View database, using information from June 2005. We show that most of the links in this database are already disclosed by less than 10 BGP routing tables (from different sources) while the amount of peers that are discovered by the rest of the sources is very small. Simulating this process over several graph models we find that policy has a significant role in the revealing process and that most of the hidden peers are of type *peer-peer*, while almost all the *customer-provider* peers are unveiled.

During June 2005 the Route-View database consisted of about 20000 ASes and 43200 links and it was a superposition of 40 different BGP routing tables[4]. With respect to the discussion above, the process of composing these BGP routing tables into a complete database is similar to the process of spanning a graph by a set of policy-based shortest path trees. In order to examine the process in which these routing tables are composed, we took from the Route-View database the 40 BGP routing tables (each representing a policy-based tree). Then, using 50 random permutations, we calculated the average number of new links found in each step. Figure 1 depicts the average number of new peers that are unveiled by the i'th BGP tree. Clearly, the first routing table discover over 20000 links[5] (the average size of a BGP routing table). However, the amount of new discovered links decreases very fast. In fact, starting from the tenth routing table, each new table unveiled less than 300 new peers (compare to more than 1000 peers in the first 5 trees), a very small number comparing to the size of the graph. In other word, a small number of BGP trees revealed a significant amount of peers while the rest of the routing tables reveal very little. Although many links remain hidden, incrementing the number of BGP routing tables will not help much in increasing the number of unveiled links.

As mentioned before, by disregarding the policy, a BGP routing table can be approximately simulated by a shortest
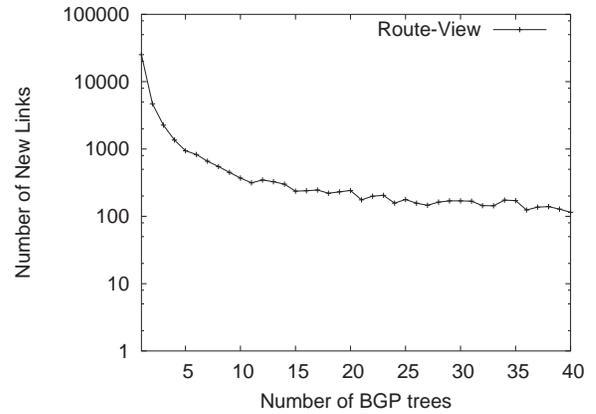

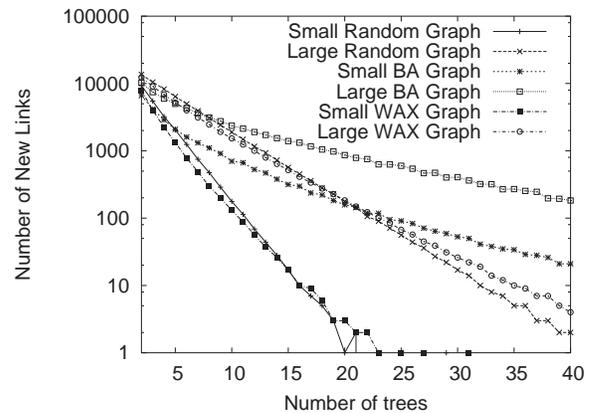Fig. 1. Route View Covering Process

Fig. 2. Graph Simulation Covering Process

path tree. Thus, we can simulate this covering process by generating graphs and cover their edges by a set of shortest path trees. In Figure 2 we show this covering process for several random graphs[6]. One can see that similar to the Route-View covering process, the contribution of new trees by means of new edges decreases very fast. Obviously, the degression is more moderate in graphs that have more links due to the fact that each tree contains exactly $N-1$ edges, thus exposing large graph requires more trees.

There are two main differences between the covering process of the AS database and our simulated graphs. The first one refers to the quantity of the cover. From the fact that the IRR database contains thousands of links that do not appear in the route-view database we know that the route-view database covers at most 60% of the links in the AS connectivity map. On the other hand, in our simulation the set of 40 trees cover almost all the edges in the graph (see Figure 3). The second

---

[3]Note that several agents can be located in a single AS.

[4]Actually, there are 45 routing tables but 5 of them contains less than 1500 links compare to more than 20000 links in the others. Since their total contribution is less than 200 links, we ignore these small tables to avoid unnecessary deviations.

[5]In practice, a complete BGP routing table may contain several paths to each AS. Thus, each table contains more links compared to a tree.

[6]In this experience, and throughout this paper we consider the following random graphs. $G_N(p)$ graphs in which an edge between two nodes exists with probability p, a Barabasi-Albert graph [17], [18] and Waxman graph [19], all have about 18000 nodes. For each model we generated two graphs with 40000 (termed a a small graph), and one with 80000 edges (termed a large graph).
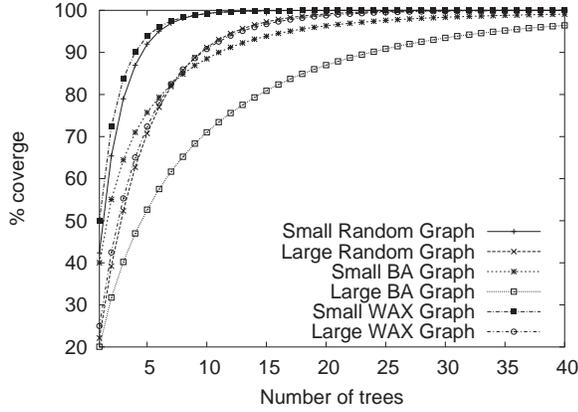
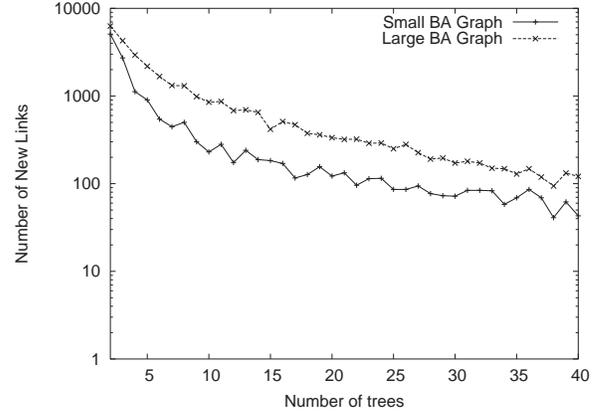Fig. 3. Percentage of Covering

Fig. 4. Policy Based Covering Process

difference can be observed by considering figures 1 and 2. While the degression of the number of new link becomes moderated in the AS graph (see Figure 1), it remains linear (in the log scale) in our simulation (see Figure 2).

In order to understand these differences we study the impact of the policy routing on the covering process. According to our policy paradigm, as described above, two ASes from different hierarchy level are connected by *customer-provider* link while two ASes from the same hierarchy level are connected by *peer-peep* link. In order to simulate this structure we need to classify the nodes in each one of our graphs into several hierarchy levels and give orientation to the links according to this hierarchy.

Using the guidelines from [20], [6], [21] we divide the set of nodes V of each graph into four hierarchy groups according to the vertex degree where the number of ASes in each group increases exponentially. Thus, the set of nodes of each graph is divided in the following way: 15 ASes (that have the highest degree) are in level 1, 150 ASes are in level 2, 1500 ASes are in level 3, and about 16500 ASes (that have the lowest degree) are in level 4. According to this hierarchy, ASes from the same groups are connected by a *peer-peer* relationship while ASes from different groups are connected by a *customer-provider* relationship. Note that while this approach (degree based hierarchy) may not be the best to approximate the hierarchy structure of the AS connectivity map, it provides "good enough" method to simulate the policy-based discovery process.

In general, when routing policy is considered, connectivity does not necessarily mean reachability, namely if two ASes are physically connected via one or more physical paths it does not necessarily means that there is a permitted path with respect to the adopted policy. In heavy-tailed models such as Barabasi-Albert, there is a strict correlation between the hierarchy structure of the graph and the policy. Thus, In these kind of models more edges are of type *customer-provider* and the reachability under the policy routing constraints is stronger compare to other model. In particular, in the Barabasi-Albert

graphs that have been used in our simulation, more than 54% of the edges are of type *customer-provider* while in the random graphs and the Waxman graphs only 30% and 35% of the edges are of type *customer-provider* respectively. Moreover, In the Barabasi-Albert graphs a single tree unveiled couple of thousands vertices, in the Waxman graphs a single tree unveiled couple of hundreds vertices, while in the Random graphs only a few dozens of vertices are unveiled by a single tree. One can observe that there is a correlation between the number of *customer-provider* links in a graph, its structure, and its reachability.

The intuition behind this property is that a permitted path cannot contain more than one *peer-peer* link [6], [21]. Moreover, *customer-provider* links must precede *customer-provider* links. Thus, in a heavy-tailed graph, where many vertices are connected to a heavy core by *customer-provider* links, almost all the ASes are connected. In other models, every vertex can reach only its local environment since the existence of many *peer-peer* links forbid many (long) paths. Using the fact that the AS graph is strongly connected[7] similarly to the Barabasi-Albert model, we simulate the policy-based covering process over this model alone. In Figure 4, one can see that similar to previous simulation, a small number of trees reveals most of the link and similar to the AS graph, the degression of number of new link become moderated. In addition, significant amount of peers (more than 45%) remain hidden (see Figure 5).

Another interesting point is the difference between the covering process of *peer-peer* links and *customer-provider* links. In [6], [21] the authors presented some heuristics to infer the type of relationship of the peering in the AS graph. In both works they found that in the Route-View database, less than 8% of the links are of type *peer-peer*. Using the algorithm presented in [6] we have found that 39700 out of 43200 links in the current Route-View database are of type *customer-provider* while 3650 are of type *peer-peer*. In contrast, in [7] the authors show that in the IRR database more than 56% of the links are of type *peer-peer*. They have tried to

---

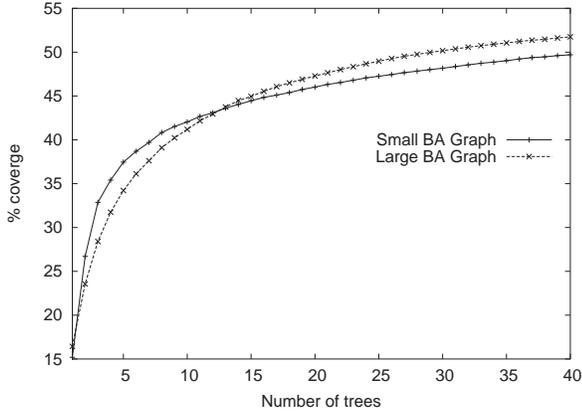[7]In the Route-View database every single tree unveiled almost all the ASes.

Fig. 5.   Policy Based Percentage of Covering

Fig. 6.   Subgraph Covering Process

explain this fundamental difference by the fact that one of the algorithms may mislead or by the fact that entries in the IRR are incorrect. We suggest a different explanation that is based on the fundamental difference between BGP based database and IRR database. As explained before, due to the locality of *peer-peer* links, it is hard to unveiled this kind of links by BGP routing tables, thus this overwhelming majority of *customer-provider* links in the Route-View database is not surprising and one should not infer that this is the ratio between the number of *peer-peer* and *customer-provider* links in the full AS graph. In contrast, the IRR database is not affected by the policy and therefore it unveils more *peer-peer* links and reflects the ratio between *peer-peer* and *customer-provider* links more accurately. Our simulations support this explanation and show that in the covering process (that simulates the BGP database) less than 3% of the links that have been unveiled are of type *peer-peer* compare to almost 45% in the full graph. Namely, a graph may contain many *peer-peer* links that will be hidden in a subgraph composed of BGP routing tables. In addition, in a subgraph that consist of a set of stars (simulating the IRR database) there are 44% *peer-peer* links, similar to the original full graph.

Recall that when we ignored policy, the covering process unveiled almost all the edges in the graph (including the *peer-peer* links). We also saw that *peer-peer* links are almost not revealed when policy is used. An interesting question is, if we consider the *customer-provider* subgraph alone (i.e the subgraph that consist of *customer-provider* links), what is the quality of the cover? namely, Does the covering process cover most of these links, or due to policy consideration many links remain hidden?

Figure 6 depicts the coverage of both the *peer-peer* and the *customer-provider* subgraphs. In the large graph (i.e. the graph that has about 80000 links) 48000 links are of type *customer-provider* and more than 90% of these links were unveiled in the covering process (recall that 97% of all the links were unveiled when the policy is ignored). In the small graph (i.e. the graph that has about 40000 links) in which
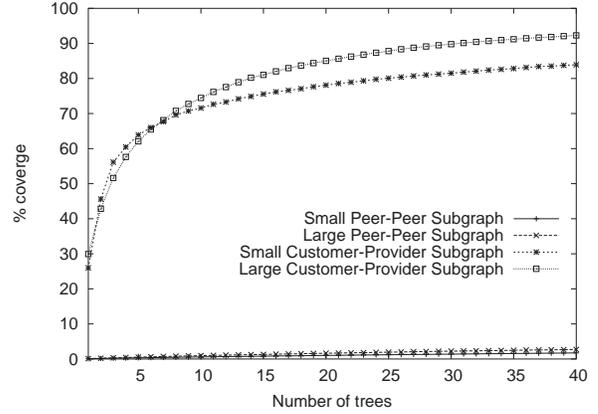
25758 link are of type *customer-provider*, 84% of these links were covered. This coverage is smaller than the one in the large graph since there are less links in this graph and reachability is more difficult. Nevertheless, in both graphs the coverage of the *customer-provider* subgraph is very large which implies that the AS connectivity graph has a similar behavior. Thus, while the covering process of the Route-View database is not complete and a significant amount of links are remain hidden, a large portion (80%-90%) of the *customer-provider* peers are unveiled by this database. Namely, the Route-View database span almost all the *customer-provider* subgraph while the *peer-peer* subgraph remains hidden.

B. IRR Policy Analysis

The information that is gathered by projects like IRR and Route-View is also incomplete in the sense that the type of relationship (i.e., customer-provider or peer-peer) is not part of the collected information. This incompleteness is due to the fact that many ASes do not expose their commercial relationship, or due to the fact that this information is gathered from BGP resources consisting on a set of path vectors. In [6] the author presented algorithms that infer the AS relationships based on the heuristic that the size of the AS is typically proportional to its degree in the AS connectivity graph. Using this heuristic (that was used in [20] to classify the ASes into four hierarchy levels) they classify the type of peering between ASes. In [21] the authors define the ToR (Type of Relationship) as an optimization problem aiming at giving an orientation to the edges of a graph such that the number of invalid paths is minimized. They propose a technique to classify the type of peering relationship by combining data from multiple vantage points. The algorithms presented in both work are based on an analysis of BGP pathes, thus they can be applied over BGP database such as Route View, and not over IRR database that gives only local view. In [7] the authors developed framework to analyze the RPSL policies of ASes in the IRR and infer the type of relationship of these ASes. In this section we use the concept that was presented in [7] and analyze the policies in the IRR database using a variant of this

method[8]. Using this analysis we infer the business relationship of the registered ASes. In addition, we consider two practical scenarios in which links may be classified in more than one way.

Routing Policy Specification Language (RPSL) allows network operator to specify routing policies at various levels in the Internet hierarchy. In particular, RPSL is used in IRR to describe BGP routing policy at the Autonomous System level. RPSL is an object oriented description language that contains many classes and attributes. The most important class from our point of view is the *aut-num* class that contains *import* and *export* attributes that describe the routing policy of an AS. Parsing the RIPE IRR we have obtained the export policy of about 8,200 registered ASes. To infer the business relationship between connected ASes (i.e. the type of peers) we use the guidelines presented in [11] and [10] in which an AS can export its routes and its customer routes to its providers and peers, but usually does not export its provider or peers routes. In contrast, AS can export its routes and its customer routes, as well as its provider or peer routes to its customers and sibling.

Consider a simple topology of six ASes described in Figure 7. The export policy of each AS is also described in RPSL format using *aut-num* class and *export* attribute[9]. For example, AS5 does not export AS4 and AS2 to AS3. Thus, according to the guidelines described above, the peer *AS5-AS3* cannot be a *customer-provider* link (where AS5 is the provider), neither *sibling-sibling* link. In order to determine the type of this link we should explore the export policy of the other edge (i.e. AS3). Since AS3 exports all its neighbors to AS5 we now can infer that *AS5-AS3* is a *customer-provider* (where AS3 is the provider). One can observe that in order to determine the type of a single peer, the export policy of both edges are required. Moreover, the import policy of an AS does not reflect the export policy of the other edge. Thus, only peers in which both edges describe their export routing policy in the IRR can be analyzed. As described above our first IRR filter mechanism that is used to remove invalid peers meets this requirement and therefore it is used in our analysis. After analyzing the IRR database using this method, we have found that 26700 out of 36237 links in the IRR database are of type *peer-peer*, 8990 links are of type *customer-provider* and small amount of 490 links are of type *sibling-sibling*.

Analyzing peering type of relationship as described above may lead to several problems in which the type of links may be interpreted in different ways. The first problem refers to the case in which an AS has no more than one provider (e.g. if an AS is a stub or if an AS is in top hierarchy). In this case (and according to our export guidelines) an AS exports all its routes
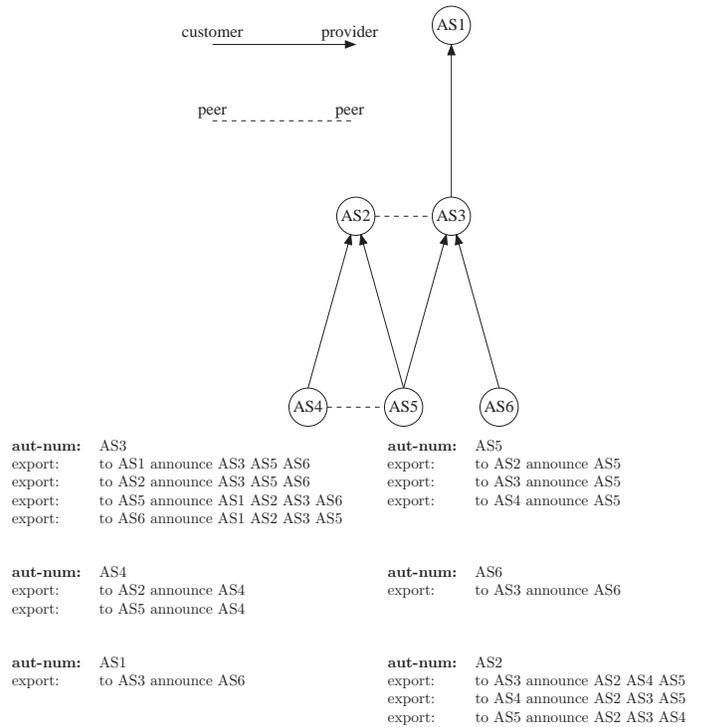

Fig. 7.   Routing Policy Example

to its neighbors, regardless the type of relationship. Thus, the export policy in these cases is not enough to determine the type of relationship of a link. For instance, using the same export policy described in Figure 7, the link *AS1-AS3* can be identified as a *peer-peer* instead of a *customer-provider* and the link *AS3-AS6* can be identified as a *sibling-sibling* instead of a *customer-provider*. In the RIPE database only 950 links (out of 36237) are in this category and may be interpreted in more than one way. In particular, 300 *sibling-sibling* links can be classified as *customer-provider* links and 650 *customer-provider* links can be classified as *peer-peer*.

While this problem refers to the way in which the export policy is interpreted, the second problem questions the classic paradigm that consider three, well defined, types of relationship (i.e. *customer-provider*, *peer-peer*, *sibling-sibling*). In contrast to the theoretical guidelines in which an AS either exports all its providers (and peers) or none, in practice there may be case in which an AS export only a subset of its provider. For instance, consider that AS5 exports AS3 to AS4 but does not export AS2. According to our analysis, in this case the link *AS4-AS5* is interpreted as a *peer-peer* link (since AS4 does not export AS2 and AS5 does not export AS2). Nevertheless, this analysis may be wrong since although AS5 does not export all its providers it exports part of them. Thus, these link cannot be interpreted by the classic paradigm presented above. In particular, one may infer that the type of the link *AS4-AS5* should be *customer-provider* (where AS5 is the provider) and not *peer-peer*. This kind of ambiguity is very common and 4800 peers in the IRR meet
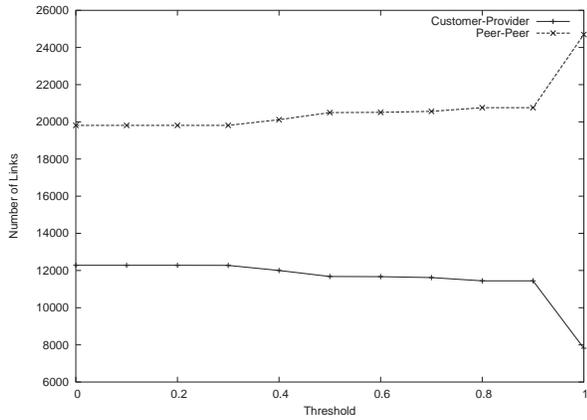
---

[8]In particular, we do not rely on the import policy of an AS, since in many cases this policy does not reflet the actual export policy of the neighbor AS. Thus, in contrast to [7] we infer the business relationship of a link only if the export policy of both edges are available.

[9]In practice, the *aut-num* class contains more attribute such as *import* to describe the import policy and other administrative attributes for maintenance. In addition, the *export* may be more complex and contain regular expressions, routes, and aggregation of routes and ASes (using *as-set* and *route-set* RPSL classes).

Fig. 8. $x - customer$-$provider$ Threshold

this category. These links may be interpreted as *customer-provider* or *sibling-sibling* instead of *peer-peer* or *customer-provider* respectively, since at least one edge exports subset of its providers.

One way to deal with this problem is to determine the type of each link by a threshold as follow. We say that a *AS-i* is *x-provider* of *AS-j* if exactly $100x\%$ of *AS-i* providers are exported to *AS-j*. According to this definition if *AS-j* is *0-provider* of *AS-i* and *AS-i* is *0-provider* of *AS-j*, thus the link *AS-j, AS-i* is of type *peer-peer*. If *AS-j* is *1-provider* of *AS-i* and *AS-i* is *1-provider* of *AS-j*, thus the link *AS-j, AS-i* is of type *sibling-sibling*. And finally, If *AS-j* is *1-provider* of *AS-i* and *AS-i* is *0-provider* of *AS-j*, thus the link *AS-j, AS-i* is of type *customer-provider* (where *AS-j* is the provider). For the rest of the cases (i.e. where $0 < x < 1$) we define a threshold t. If $x > t$ then x is considered to be 1. If $x \leq t$ then x is considered to be 0. Figure 8 depicts the number of *peer-peer* and *customer-provider* links in the IRR for different threshold values. Clearly, for small threshold values the amount of *customer-provider* links increases at the expense of *peer-peer* links (in absolute number, the total amount of *sibling-sibling* links almost does not change and thus it is not considered in our discussion). Another interesting inference from Figure 8 is that almost all the links that moved from *peer-peer* to *customer-provider* type have a big threshold value (i.e. $t = 0.9$). Namely in almost all the cases, a specific AS exports all its providers (to another AS) except one.

Another way to deal with this problem is to consider new types of relationship. Recall, that the types of relationship intend to describe the business relationship between ASes. For instance, in a *customer-provider* link the providers gives to the customer full access to its routes. Nevertheless, in some cases a provider gives only partial access to its provider (i.e. some of the routes are blocked), thus the provider is not a *full-provider* but a *semi-provider*.

III. APPROXIMATING THE AS CONNECTIVITY GRAPH SIZE

Using data, collected during June 2005 from Route-Views, IRR, and DIMES projects we unveiled 83782 AS peers[10]. Nevertheless, as discussed earlier, this data is not complete enough and despite the increasing effort to reveal the full map, some peers may remain hidden. In this section, we address the following question: What is the overall size of the AS connectivity graph. We want to be able to answer this question without assuming anything about the full (partly unrevealed) graph. It is important to note that having a good approximation of the size of the AS connectivity map is not just a theoretical question. The overall number of active ASes is known (about 21000 during June 2005), and thus the overall number of edges translates directly to the average node degree - which is an important parameter regardless of the model we use. We show in this section that the AS connectivity map contains at least 128000 links and most likely the size of the graph is higher.

First, we try to estimate the size of the connectivity graph using the degree of the stars in the IRR database assuming that the database is representative. This database consist of 6583 stars and 72474 links, using the first filtering or 7129 stars and 82339 links, using the second filtering. Thus, the average degree is $\frac{72474}{6583} = 11$ and $\frac{82339}{7129} = 11.54$ respectively. Using the fact that the AS graph consists of 21000 active ASes this implies that the AS graph consists of $21000 \cdot \frac{11}{2} = 115000$ or $21000 \cdot \frac{11.54}{2} = 121000$ peers. Clearly, the estimation using this method depends on the filtering used. An aggressive filtering, may remove legal peers and therefore it induces a smaller IRR graph compare to the real one. Thus, the estimation in this case deviates down. On the other hand, using a moderate filtering the induced graph may contain peers that are no longer valid. In this case the estimation deviates up. To emphasize this issue let us estimate the size of AS connectivity graph using the unfiltered IRR data. In this case the database contains 9247 stars and 138343 reflecting an average degree of 14.9 and a size of 156000 edges for the full graph.

With respect to the assumption that the database is representative, one may refer to the fact that almost all the ASes in the IRR are located in Europe and it is possible that the average degree of ASes in Europe differs from the average degree of an ASes in other region. In order to avoid such assumption, we aim at estimating the number of AS peering relations using the data added by the IRR database to the Route-View database.

In Section II-B we analyzed the type of relationship in the IRR database and showed that 8990 out of 36237 peers (i.e. 25%) in the filtered IRR[11] are of type *customer-provider*. If we consider the last discussion in Section II-B regarding the way in which export policy is analyzed, there may be a doubt regarding 4200 *peer-peer* links. In this case, when we consider only the undoubt links, $\frac{8990}{8990+22056} = 29\%$ of the link are of type *customer-provider*. Assuming that the sampling space is

[10]This is after filtering the IRR data as discussed in Section II.

[11]Recall that this analysis requires that both edges will be in the database. Thus, we have used only the second filter mechanism.

representative, 25-29% of the peers in the full AS connectivity map are of type *customer-provider* while 71-75% are of type *peer-peer* (or *sibling-sibling*). Since the Route-view database contains 39700 *customer-provider* links it indicates that the full AS connectivity map contains at least $\frac{100}{29} * 39700 = 138000$ links. Using a more conservative analysis, considering all the doubt links (i.e. link that their type is unknown with respect to the last discussion in Section II-B) as a *customer-provider* links, 37% of the links in the IRR are of type *customer-provider*. Therefore, the same technique indicates that the AS connectivity map contains at least 107300 links. Nevertheless, in this case $0.36 * 36237 = 13045$ of the peers in the IRR are of type *customer-provider* while the IRR and the Route-View data have only 6730 common *customer-provider* links. Thus, the size of the *customer-provider* subgraph is at least $39700 + 13045 - 6730 = 46015$ links. In this case the lower bound is 128000 links[12].

Obviously, these estimations are very conservative and intent to give a lower bound on the size of the AS map. For instance, in the last estimation we assumed that the Route-View data does not discover about 6500 *customer-provider* links in Europe alone. This indicates that (under the assumption that 36% of the links are of type *customer-provider*) a large portion of 20000 *customer-provider* links remain hidden and therefore the estimation should be 165800 links.

The next estimation is based on the intersection of the new sample space (the stars coming from the IRR) with the existing coverage created by trees from the BGP routing information. If the new data contains a set of independent edges, we could measure the portion of the full graph covered by the BGP data, because it gives us the probability that an edge is covered by the BGP data. Recall that the Route-View database contains 43200 links and the unfiltered IRR contains 102106 links. The union of these two graph contains 128697 links. This means that 16618 out of the 102106 edges where already covered by the Route-View database. Therefore, the probability of an edge to be discovered by the Route-View database is $\frac{16618}{102106} = 0.16$ and the total number of edges can be approximated by $1/0.16 \times 43200 = 265400$. Obviously, removing invalid peers from the IRR database, increase the probability of an edge to be discovered by the Route-View database. Thus, similar to pervious method (using the average degree of a node), the estimation is significantly affected by the filter mechanism. Using the filtered IRR, this probability is increased to $\frac{12107}{46611} = 0.26$ (where 12107 is the number of peers exists in the IRR and have already covered by the Route-View database, and 46611 is the total number of peers in the filtered IRR database) using the first filter mechanism and $\frac{8719}{36237} = 0.24$ using the second filter mechanism. Thus, the total number of edges can be approximated by $1/0.26 \times 43200 = 166000$ and $1/0.24 \times 43200 = 180000$. Note that the accuracy of this

Graph	Size Estimation	Graph Size
BA	161700	161955
BA+Wax	141085	141135

TABLE I

GRAPH SIZE ESTIMATION

Estimation Method	400 Biggest Stars	1000 Small Stars
Average Degree (First Method)	718000	58900
Data Intersection (Second Method)	195000	121800

TABLE II

ROBUSTNESS OF ESTIMATION

method depends of the independency of the database. Thus, if there is a correlation between the data the estimation will deviate and indicate a smaller value (and vice versa).

In order to examine the method we simulated the process over several graphs. The first graph is a Barabasi-Albert graph and the second one is a superposition of a Barabasi-Albert graph (that contains 40000 edges) and a Waxman graph (that contain 100000 edges). Both graphs contain 18000 nodes. From each graph we have constructed two subgraphs. The first one consisted of 5000 random stars (that simulates the IRR database) and the second consisted of 40 policy based shortest path trees (that simulates the Route-View database). Table I summarizes the average results of the estimation process over 20 independent iteration, with respect to the actual size[13].

In contrast to the vertex degree method, where the estimation is significantly affected by the average degree of the sampling space, this method seems to be much more robust with respect to the degree of the nodes in the sampling space. However, our sampling space is indeed dependable and the estimation is still affected by the average degree. Every AS has at least one peer in the Route-view (Recall that the Route-View consists of a set of trees that span the graph), thus the probability of an edge in an AS with small degree to be covered by the Route-View is bigger than an AS with bigger degree. To demonstrate it, we divided the IRR into two subgraphs. The first subgraph contains the 400 biggest stars (i.e. the stars with the highest degree) and the second contains 400 small stars[14]. Table II summarizes the estimation using these two subgraphs (instead of using the full IRR).

Next, we use the same technique to estimate the size of the AS map using data from the DIMES project. Thus, we measure the intersection between DIMES and IRR data and between DIMES and Route-view data. During June 2005, DIMES unveiled 38928 links. 7000 links of them have already been revealed by the IRR data, while 23850 of them have been revealed by the Route-View data. Namely, the probability of

---

[12]In [7] the authors indicate that 42% of the links in the IRR are of type *customer-provider*, thus estimating the lower bound using their results may be different. However, since information regarding the common *customer-provider* links is also required, we cannot present this estimation.

[13]Recall that in the simulations, unlike in the AS case, we know the actual size of the full graphs.

[14]The IRR contains many ASes with one peer (i.e. their degree is one). These links are found by the Route-View as well. Thus is order to avoid this side effect we did not took the ASes with the smallest degree but ASes that have at least 5 peering relationship.

an edge in the DIMES data to be discovered by the IRR data or by the Route-View database is $\frac{23850}{43200} = 0.55$ and $\frac{7000}{36237} = 0.19$ respectively. Thus, the total number of edges can be approximated by $1/0.55 \times 38928 = 201606$ using IRR and $1/0.19 \times 38928 = 70522$ using Route-View. Clearly, the second estimation (using DIMES and Route-VIew data) seems to be very low. However, since DIMES is based on *traceroute* queries, it obtains only links that traverse permitted AS paths. Thus, it has strong overlapping with BGP based database such as Route-View and the probability of an edge covered by DIMES to be unveiled by Route-View is bigger compare to two independent subgraphs.

Currently DIMES consists of almost 4000 distributed agent performing *traceroute* to a set of random IP addresses. While the IRR contains peering information of ASes located in Europe, less than 25% of DIMES agents are located in Europe. Moreover, the majority of IP addresses are located outside Europe. Thus, most of the *traceroute* performed by DIMES agents are probably targeted to destination that are not covered by IRR data and their source is outside of IRR scope as well. Therefore, the correlation between IRR and DIMES data is weaker than independent random subgraphs and the estimation based on these two data set deviates up.

Using the estimations presented so far, one can bound the size of the AS connectivity map between 128000 and 200000 links. Both methods used in this section are sensitive to many parameters (e.g. the average degree, the accuracy of filtering and the type of relationship analysis, independency of the database, etc.), thus trying to approximate the accurate size of the AS connectivity map is very difficult. Nevertheless, while the lower bound seems to be too conservative, the upper bound is too loose. Therefore, the actual size of the AS connectivity map is somewhere between these boundaries.

IV. Vertex Degree Distribution

In their paper, Faloutsos et al. [4] showed that despite the apparent randomness of the Internet, simple power-laws hold for the Internet in the AS level. This novel observation was adopted by many researches and it is one of the basic building blocks for modelling the AS connectivity map. The authors use the NLANR - National Laboratory for Applied Network Research data [22] consisting of several BGP routing tables. This kind of database (i.e. a database that consisting on routing tables alone) is incomplete and may cause a significant inaccuracies. In other words, the graph that is derived from this kind of database is only a subgraph of the full AS graph, thus properties that hold in the subgraph may not be valid for the full graph. In [5] the authors questioned this observation and showed using a more complete database (yet not fully complete) that the vertex degree distribution deviates from the straight line (reflecting a power-law distribution). One can see this deviation in Figure 9 that depicts the complementary distribution function of the AS degree as it is derived from the route-view database alone, and from the route-view plus IRR data. The data for this graph has been collected during June 2005. In this section we study the vertex degree distribution of
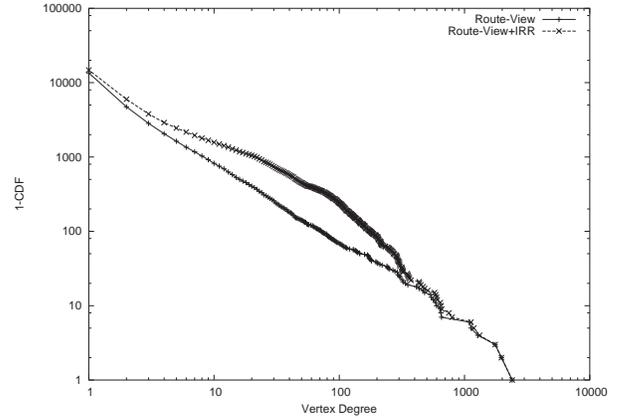

Fig. 9. AS Vertex Degree Distribution

the AS graph. We show that despite the fact that the databases are not complete, their vertex degree distribution may reflect the distribution of the full graph. In particular, we observe that the *customer-provider* subgraph follows the power-law while the *peer-peer* subgraph may behave differently. Although the most complete database contains both IRR, Route-View and DIMES data, in order to understand the vertex degree distribution of the AS graph, we use the IRR database alone since it is not affected by the policy and therefore it is much more representative in the sense that the ratio between *peer-peer* and *customer-provider* link is more accurate.

As discussed in Section II, BGP based database contains mostly *customer-provider* links and therefore it may be considered as a subgraph of the *customer-provider* graph. Moreover, using more representative database (i.e the IRR database) we showed that the ratio between *peer-peer* and *customer-provider* links is completely different and there are many more *peer-peer* links in the full AS graph. Therefore, although the vertex degree distribution of databases such as Route-View and NLANR follows the power-law, it reflects the distribution of the *customer-provider* subgraph alone and does not give information regarding the distribution of the *peer-peer* subgraph.

To support this finding, we use an independent *customer-provider* subgraph that is based on IRR data. We use the analysis described in Section II-B to divide the IRR data into *peer-peer* and *customer-provider* subgraphs. With respect to the last discussion in II-B, we consider only the links with undoubt type. Thus, we have 8990 *customer-provider* and 22050 *peer-peer* links. Figure 10 depicts the vertex degree distribution of the *customer-provider* subgraphs as they derived from the Route-View and the IRR database. The size of the *customer-provider* subgraph derived from the IRR data is much smaller than the Route-View subgraph (in particular the first contains 8990 links while the second contain 43200 links), thus it is located below Route-View subgraph. Clearly, both graphs follow the power-law.

In contrast to the *customer-provider* subgraph we suggest
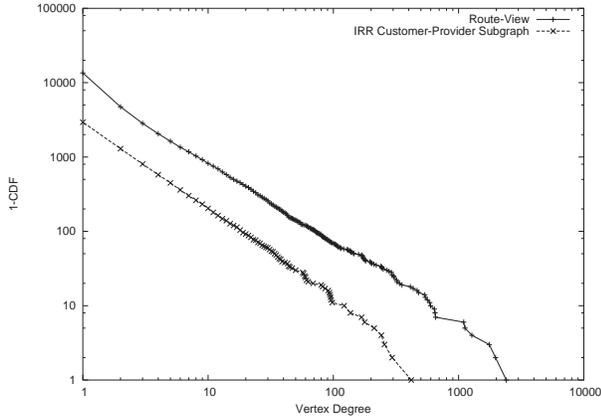
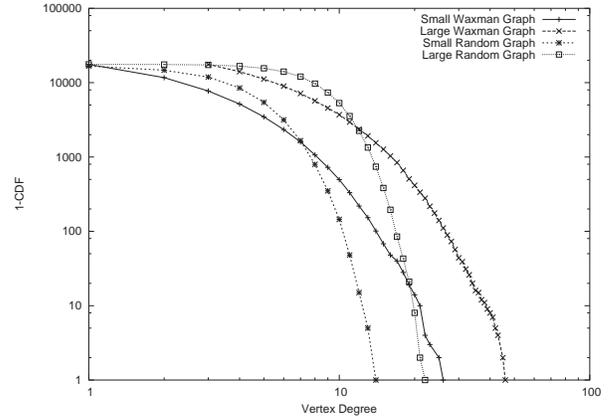Fig. 10.   Vertex Degree Distribution of *customer-provider* Subgraphs

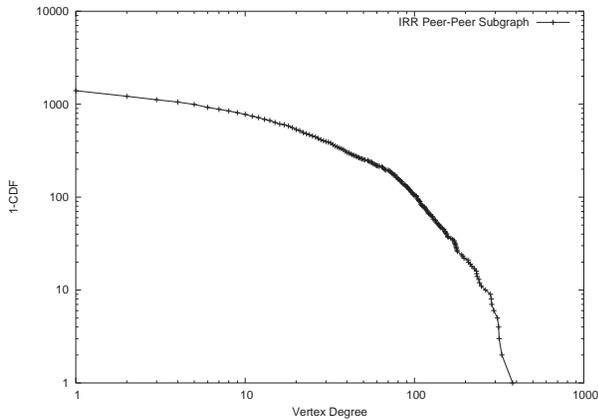Fig. 12.   Vertex Degree Distribution of Several Graphs

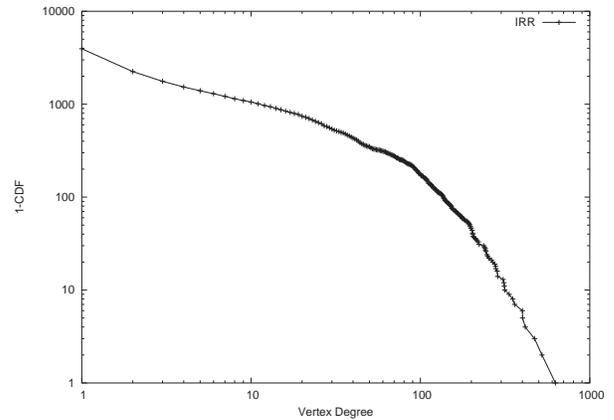Fig. 11.   Vertex Degree Distribution of *peer-peer* Subgraph

Fig. 13.   IRR Vertex Degree Distribution

that the *peer-peer* subgraph does not follow the power law. Figure 11 depicts the vertex degree distribution of the *peer-peer* subgraph derived from the IRR. The distribution is completely different. In particular, it is much more similar to the distribution of Waxman graphs (see Figure 12).

Naturally, the distribution of the full graph is a superposition of both distributions (i.e. *customer-provider* and *peer-peer* subgraph). Figure 13 depicts the vertex degree distribution of the IRR. Recall that about 75% of the links in the IRR database are of type *peer-peer* and only 25% are of type *customer-provider*. Thus, the distribution of the full graph is mostly affected from the *peer-peer* subgraph.

So far we drew conclusions regarding the vertex degree distribution using only partial data. In particular, we used subgraph consisting of a collection of policy-based shortest path tree (i.e. the Route-View database) and another subgraph consisting of a set of random "stars" (i.e. the IRR database). As pointed up in [23], the vertex degree distribution of a graph may be differ from the distribution of a subgraph that is derived from the original graph. In particular, given a random graph, the vertex degree distribution of a subgraph formed by a collection of shortest paths trees from a set of sources

to a set of destination, may be heavy-tailed. Thus, one may question our inferences by suggesting that these subgraphs do not represent the vertex degree distribution of the full graph.

However, in [23] the set of destination vertices is very small compare to the number on vertices in the graphs (only 1%) while when we consider BGP routing table, the destination set contains almost all the vertices in the graph. The significance of this difference is depicted in Figure 14. One can observe that when the size of the set of destination is small, the vertex degree distribution of the derived subgraph follow the power-law, but increasing the size of the set bring the distribution of the derived subgraph closer to the distribution of the full random graph.

While the Route-View database represents a policy-based shortest path trees subgraph, the IRR database may represent a random subgraph (generated by a set of random stars) and does not follow the last discussion. Our simulation results indicate that in this case the vertex degree distribution of the subgraph is similar to the full graph. Figures 15 and 16 depicts the vertex degree distribution of small and big subgraphs generated by a set 1500 and 5000 random stars respectively. Both subgraphs preserve the vertex degree distribution of the full graph.
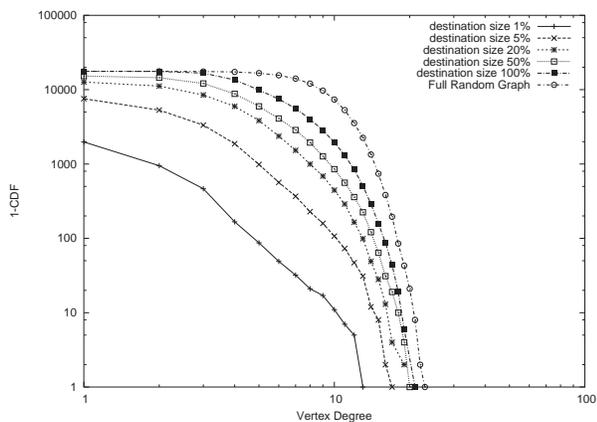
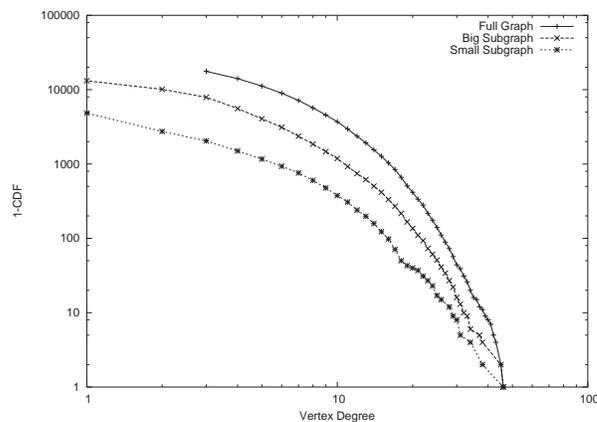Fig. 14.   Vertex Degree Distribution of a Shortest Path Subgraph

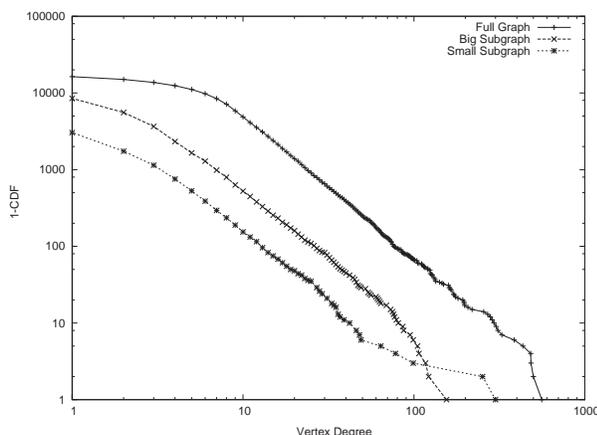Fig. 16.   Vertex Degree Distribution of a Waxman Subgraphs

Fig. 15.   Vertex Degree Distribution of a Barabasi-Albert Subgraphs

V. DISCUSSION AND FUTURE WORK

In this paper we showed that despite the increasing effort to unveil the AS connectivity map at least 35% of the links are still missing from all known databases. Less conservative estimations indicate that more than 50% of the link remain hidden. By understanding the gathering process of databases such Route-View and IRR we showed that almost all missing links are of type *peer-peer* while a considerable amount of *customer-provider* links are revealed. Thus, trying to disclose the full AS connectivity graph by an increasing set of BGP routing table or by a set of agents performing periodic *traceroute* (both discover mostly *customer-provider* links) may be insufficient in order to fully unveil the *peer-peer* subgraph. A better understanding and modelling the structure of these unveiled *peer-peer* links and their location in the hierarchical structure is a subject to future work. Note that unlike the Route-View and IRR databases, at this time the DIMES project is relatively new and a more thorough study of its information gathering is in place.

We also studied the vertex degree distribution of the AS connectivity graph and showed that the distribution of the *peer-peer* subgraph is considerately different from the one of the *customer-provider* subgraph. These inferences, may lead to new models describing the AS connectivity map that consist of two separate models. One describing the *peer-peer* subgraph and another describing the *customer-provider* subgraph. In particular, these models should take into account our finding regarding the vertex degree distribution of each subgraph. Namely, the vertex degree distribution of the *customer-provider* subgraph follows the power-law and the the vertex degree distribution of the *peer-peer* subgraph is similar to the distribution of a Waxman graph.

An interesting area for future work is studying a more complex peering relationship that does not follow the classic export paradigm. As we pointed out in Section II-B, an AS may export to some its customers only a subset of its provider's paths. In such a case, one or more providers may give (to this AS) only local services. This kind of export policy that actually determines a new type of relationship between ASes, was not studied in the past and it may lead to a different model describing and characterizing the AS hierarchy connectivity map.

Another direction is to study some classical routing related problems (e.g. minimum spanning tree, the cache location problem) that have been well studied in the past over flat graphs, over the hierarchical structure of the AS graph. In particular, in this case the AS graph connectivity does not necessarily mean reachability and the triangle inequality does not necessarily holds, and thus new approaches may useful.

REFERENCES

[1] "University of oregon route views projects," http://www.routeviews.org.
[2] "Internet routing registry," http://www.irr.net.
[3] "Dimes - distributed internet measurements and simulations," http://www.netdimes.org.
[4] C. Faloutsos, P. Faloutsos, and M. Faloutsos, "On power-law relationships of the internet topology," in *Proc. of ACM SIGCOMM*, September 1999, pp. 251–262.
[5] Hyunseok Chang, Ramesh Govindan, Sugih Jamin, Scott J. Shenker, and Walter Willinger, "Towards capturing representative as-level internet topologies," *Computer Networks Journal*, vol. 44, no. 6, pp. 737–755, April 2004.

[6] Lixin Gao, "On inferring autonomous system relationships in the internet," *ACM/IEEE Transactions on Networking*, vol. 9, no. 6, pp. 733–745, December 2001.

[7] G. Siganos and M. Faloutsos, "Analyzing bgp policies: Methodology and tool," in *Proc. of IEEE INFOCOM*, 2004.

[8] Lixin Gao and Jennifer Rexford, "Stable internet routing without global coordination," *ACM/IEEE Transactions on Networking*, vol. 9, no. 6, pp. 681–692, December 2001.

[9] Giuseppe Di Battista, Maurizio Patrignani, and Maurizio Pizzonia, "Computing the types of the relationships between autonomous systems," in *Proc. of IEEE INFOCOM*, 2003.

[10] Geoff Huston, "Interconnection, peering and settlement," in *Proc. of INET*, June 1999.

[11] Cengiz Alaettinoglu, "Scalable router configuration for the internet," in *Proc. of IEEE IC3N*, October 1996.

[12] Lixin Gao, Tim Griffin, and Jennifer Rexford, "Inherently safe backup routing with bgp," in *Proc. of IEEE INFOCOM*, 2001.

[13] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra, "Routing policy specification language (rpsl)," *Internet RFC 2622*, 1999.

[14] D. Meyer, J. Schmitz, C. Orange, M. Prior, and C. Alaettinoglu, "Using rpsl in practice," *Internet RFC 2650*, 1999.

[15] "Reseaux ip europeenne (ripe)," http://www.ripe.net.

[16] Shi Zhou and Raul J. Mondragon, "The missing links in the bgp-based as connectivity maps," in *Passive and Active Measurement Workshop (NLANR-PAM03)*, April 2003.

[17] Albert-Laszlo Barabasi and Reka Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509–512, October 1999.

[18] Reka Albert and Albert-Laszlo Barabasi, "Topology of evolving networks: Local events and universality," *Physical Review Letters*, vol. 85, no. 24, pp. 5234–5237, December 2000.

[19] Bernard M. Waxman, "Routing of multipoint connections," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1617–1622, December 1988.

[20] Ramesh Govindan and Anoop Reddy, "An analysis of internet inter-domain topology and route stability," in *Proc. of IEEE INFOCOM*, 1997.

[21] Lakshminarayanan Subramanian, Sharad Agarwal, Jennifer Rexford, and Randy H. Katz, "Characterizing the internet hierarchy from multiple vantage points," in *Proc. of IEEE INFOCOM*, 2002.

[22] "National laboratory for applied network research (nlanr)," http://www.nlanr.net/.

[23] A. Lakhina, J. W. Byers, M. Crovella, and P. Xie, "Sampling biases in ip topology measurments," in *Proc. of IEEE INFOCOM*, 2003.