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

Danny Raz, Rami Cohen
Computer Science Department, Technion, IIT
{danny, ramic}@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, 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. Starting with the fundamental work of Faloutsos 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 a 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 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. Using the second moment allows us to reach conclusions regarding the structure of the system we were measuring, and in particular to estimate its total size. We present strong evidence to the fact that a considerable amount (more than 50%) 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. We also identify a new measure for graphs, the number of spanning trees needed to cover all the edges of a graph, that plays a significant role in the matching of theoretical graph model to the real structure of the revealed portion of the AS connectivity map.

Index Terms

Graph theory

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, inferring from the collected data the actual topological structure, and analyzing this structure in order to understand the inherently important characteristic and evolution of the system. 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, but largely because there is no direct way to retrieve information from a 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. (As was pointed out recently in the router topology context in [1]). In other words, in 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. Using the second moment allows us to reach conclusions regarding the structure of the system we were measuring.

This technique is applied to the Autonomous System (AS) level connectivity graph. 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 and the second moment method to approximate the actual number of missing links in the different available AS databases reported recently [2]. We present strong evidence to the fact that a considerable amount (about 50%) of the links in the AS level are still to be unveiled.

Another interesting result of our findings is the identification of a new measure for graphs, the number of spanning trees needed to cover all the edges of a graph. This measure plays a significant role in the matching of theoretical graph model to the real behavior of the revealed portion of the AS connectivity map. However, since BGP routing is policy based, we use an hierarchical model to create peering relations as indicated in [3] [4], and [5]. We then used policy based trees instead of shortest path trees, and compared the behavior of different graph models to the actual results of real measurements. Our findings indicate that the connectivity graph is composed by a superposition of the provider-customer subgraph, and the peer-peer subgraph (as proposed by [6]). We also observe using the gathered information, that each subgraph has a unique structure, and almost all the missing links are of type peer-peer.

The structure of this paper is as follows. First, in Section II, we formally define the model, and discuss the process of collecting peering relation information by trees. Then, in Section III we use the second moment technique approximate the actual number of the links in the AS level. In Section IV we reexamine our assumption, discuss policy based trees, and revisit the properties of the AS connectivity map. We finish with a short discussion in Section V.

II. Problem Definition and Formal Model

We model the AS level connectivity map 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 ASs. 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 will help us create a rigorous view of the discovery process, and hopefully will maintain the most important and relevant aspects of the discovery process while eliminating less important issues.

The base for many of the AS level mapping is BGP path data. This data contains the paths (in terms of ASs) to each of the relevant subnetworks. For simplicity, we will assume that this data, the path vector of an AS, contains paths to all other ASs 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. One of the well appreciated advantages of BGP is its ability to use policy routing rather than shortest path routing. Thus, the collection of all the path vectors from a given AS to all other ASs is a DAG. 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. As mentioned before, the NLANR data base [7] and the original Oregon [8] project contain a collection of such DAGs collected from several ASs over the last few years.

Again, in order to simplify our modelling of the data collection process, we assume for now that routes are done along shortest paths and the information retrieved form a single AS in a tree. This, of course, in not correct since routing in BGP is policy based, and in addition, BGP routing table may reflect more than one route for a destination AS. However, this assumption makes the discussion in the retrieval process formal and rigorous. Moreover, using traceroute both in the AS ([9]), and the router level, also produces similar trees (assuming traceroute is performed to all possible subnetworks), and thus the same approach could be adopted in different measurement scenarios. We re-examine the validity of this assumption and use a more complex hierarchical graph model in Section IV.

A. Covering graph by shortest path trees

One can now model the process of retrieving peering information by creating a shortest path tree (if all link lengths are equal, this is exactly the BFS - Breadth First Search tree) from a given node to all the nodes in the graph. The question of discovering peering relations translates now to the amount of edge covered by a union of BFS trees rooted at a given set of nodes. In other words, the amount of peering relations covered by a collection of BGP path vectors

\[1\] A list of ASs that provide access to the Looking Glass tool can be found in www.traceroute.com.
from say 40 ASs, corresponds to the amount of edges covered by a set of 40 BFS trees. Figure 1 depicts the amount of edge coverage achieved by performing this process at random (i.e., the nodes from which the trees are rooted are chosen uniformly at random) for several graphs. The figure shows that the amount of coverage strongly depends on the graph structure.

In this figure we present results regarding 5 different graphs, all have about 11600 nodes. The first graph is a full clique, a graph in which each node is connected to all other nodes. Note that in our graphs, where all edge lengths are equal, every edge must be covered by trees rooted at its end points, and therefore every edge must be covered by at least 2 trees; however, it may be covered by other trees as well. For the full clique graph each edge is covered exactly by two trees, and thus $N - 1$ trees will be needed to cover the entire graph. In the full clique graph, each BFS tree is just a star connecting the root to all other nodes. In any order of choosing the sources for the trees, the first tree will have $N - 1$ edges, the second one will add $N - 2$ new edges to the cover and so on. The overall number of edges covered by the first $i$ trees is then

$$i(N - (i + 1)/2) = \frac{2i}{N - 1} - \frac{i + 1}{N(N - 1)}$$

For small $i$’s (in the graph $i \leq 400$) this is very close to $\frac{2i}{N - 1}$ which looks very small because $N \approx 11600$.

The second graph described in Figure 1 is a random graph for which an edge between two nodes exists with probability 0.0005. This creates a graph with approximately 33000 edges for 11600 nodes. One can see that in this graph the marginal effect of adding new trees decreases very fast, and 47 trees are needed to cover the entire graph. Moreover, if we consider Figure 2 in which the amount of uncovered edges is plotted using a logarithmic scale, one can see that the coverage of the random graph is exponential. That is, a constant fraction of the uncovered edges is covered in each iteration. This process is very similar to a process in which at each step $N$ edges are chosen uniformly at random from the graph. Note that choosing a BFS tree is different in general, since the $N$ chosen edges are not independent and
not any group of $N$ edges form a tree. However, for the case of random graph with small edge probability, the tree covering process is similar to a random sampling process.

In order to understand this process, we plotted in Figure 4 the popularity of each edge in terms of the number of trees that covers it. The graph depicted the percentage of edges that are covered by less than a certain factor of the trees. For the random ($G(n, p)$) graph there are no unpopular edges, i.e., every edge is covered by at least 20% of the trees, and therefore a small number of trees is very likely to cover all edges and thus the entire graph. Moreover, almost all edges have popularity of 28%-34%, and thus the probability of adding an edge to the cover is indeed independent of the tree and is almost fixed.

The third graph we considered was a Barabasi-Albert (BA) graph with a very small core, and 3 outgoing edges per nodes (see [10]). Again, this brings us to about 33000 edges for 11600 nodes. Here the picture is different and almost 100 trees are needed to cover the graph. Also if we consider again Figure 2 we observe that the amount of coverage is not a constant factor from the uncovered edges, but this fraction becomes smaller.

If we plot the same information on a log-log scale, we can observe (see Figure 3) that the process is a power law process. This can be partly explained by the edge popularity as described in Figure 4. For the BA graph, there are a considerable amount of edges that are not popular, i.e. covered only by a small fraction of the possible BFS trees. Thus, covering these edges in the random process required more trees. The reason for this edge popularity behavior is the structure of the BA graph. Many of the shortest paths go via several very “heavy” nodes (nodes with a large degree), and thus links between two nodes with smaller degree, are most likely to be covered by only a very small number of trees. In fact, edge popularity is termed “load” in the research of scale free networks [11], and “betweenness connectivity” or BC in social studies [12], and it was shown that it exhibits a power law in BA graphs (and other scale free graphs).

The next two graphs we considered reflect the known AS connectivity map as reported in [2] \(^2\). The graph marked as DASC-ALL (DASC stands for Discovered AS Connectivity Grap) represents the full discovered set of peering

\(^2\)We used here the 2002 data which is available from http://topology.eecs.umich.edu/data.html
Fig. 3. The uncovered part of random shortest path trees (log-log scale)

Fig. 4. Edge popularity in terms of the portion of trees that cover it.
relations, and the one marked as DASC-ORG contains only the connectivity discovered by the Oregon cite ([8]). One can see from Figure 1 that the number of trees needed to cover the DASC-ALL graph is very big. In fact, 100 trees covered only about 85% of the edges. The number of trees needed to cover all the graph was more than 500 on average. This again can be partly explained by the fact that many of the edges, much more than even in the BA graph, have very small popularity (see Figure 4). If we consider only the Oregon data, then a smaller number of trees cover a bigger portion of the edges. This is due to two factors. The first factor is the overall number of edges which is much smaller and therefore the coverage percentage is higher. The second factor is that as can be observed from Figure 4, is that fewer edges are so unpopular, and thus less trees are need to cover these edges. Yet, many more trees are needed to cover the AS graphs than any of the tested models.

The main question is what is the source of this phenomenon, i.e. why are so many edges covered by a small number of trees. One answer to this question is that the graph representing the full AS connectivity map has this property (we discuss later in this paper, examples for graph the expose similar behavior). Another possible answer is that this behavior is not a characteristic of the full AS connectivity map, but a result of the sampling process and the fact that DASC is only a partial view of the entire AS connectivity map.

B. The AS connectivity sampling process

To address this point we have to further understand the sampling process that creates the discovered AS connectivity graphs. As described in [2], additional information is added to the basic BGP view. This information contains local BGP views from Looking Glass interfaces, and connectivity views retrieved from IRR data bases [13].

The IRR data base contains local connectivity information for the registered ASs. In terms of our graphs, this corresponds to discovering all the edges connected to a given node. These objects are referred to as stars for obvious
reasons. Thus the discovered AS connectivity graphs are created by covering edges by a number of trees, and then adding a considerable amount of stars.

We can now return to the source of the large number of unpopular edges in the DASC graph. We created Synthetic graphs using the above covering process. That is, we started with a BA graph and took a subgraph of it covered by 6 trees. We then added to this subgraph a set of 1500 stars to create another subgraph. In order to have about 33000 edges at the end of the process, we started with a BA graphs that has about 66000 edges, this graph is called BA-big in Figure 5, where the amount of cover by random trees is depicted. The same process was done for a random \( G(n, p) \) graph, and the results are depicted in Figure 7. As for the random graph, it seems that the only main factor is the size, i.e., the number of needed trees to cover the graph depends only on the number of edges, and the exponential behavior observed for all subgraphs. For the BA graphs, the situation is different. If we examine Figure 5, we can see that the subgraph created by trees converges to the small BA graph, while the one created by trees and stars, starts like the small graph, but converges to the large number of tree coverage of the large BA graph. This can be explained by the fact that the stars add to the graph a considerable amount of “unpopular” edges, and thus, we need to cover these edges by a large number of trees. This indicates that at least in this case the number of trees needed to cover the “sampled” graph is very close to the one needed for the “full” graph.

To investigate this point further, we plotted in Figure 8, the amount of cover by random trees for the 2001 and 2002 AS graphs. We compare all the discovered edges (as reported in [2]), and the BGP data obtained only from the Oregon server. One can see that the amount of coverage for the graphs that contain the IRR information is significantly higher, which indicates that the full (unknown) AS connectivity graph has a similar characteristic.
C. Minimum BFS coverage

One way to attack the problem of discovering whether a given graph is only a part of a bigger graph is to check if it was created by a union of a small number of spanning trees. That is, we want to find the minimum number of BFS trees that cover a given graph. This number is a characteristic of the graph. Note that it may be different from the numbers reported in Figure 1, since in the figure we use a random process and here we are looking at an optimal solution. For example, if we take a graph and consider the subgraph obtained by the union of 10 BFS trees, and then try to cover the resulting graph by BFS trees in a random process the number of trees may be considerably bigger than 10, while the optimal number of trees needed is at most 10.

One can formally define this characteristic as a graph theoretic problem in the following way. A set of nodes \( S = \{v_1, v_2, \ldots, s_k\} \) is a Shortest Path Tree (SPT) cover of the graph \( G \) if the union of all shortest path trees rooted at the nodes of \( S \) covers the entire graph. A graph \( G \) has a SPT cover \( k \) if there exists a set \( S \) of \( k \) nodes that is a STP cover for \( G \), and \( k \) is the smallest such number.

This novel characterization of graph is interesting also from the point of view of constructing a minimal set of measurement instrumentation that covers an entire graph. As pointed out before, for a full clique of size \( n \), the SPT cover is \( n - 1 \), and, of course, the SPT cover of any tree is 1. However, computing the SPT cover for an arbitrary graph is a hard problem. It is not difficult to see that this is a special case of the Minimum Set Cover problem (problem SP5 in [14]). However, it turns out that in fact SPT cover is as hard as the Minimum Set Cover problem. In order to prove such a claim, one needs to present a reduction from the Minimum Set Cover problem to the SPT cover problem. Such a reduction can be found [15], but it is out of the scope of this paper. Moreover, since this reduction is an approximating conserving reduction, one gets that the best known approximation for the SPT cover problem is an \( O(\log n) \) approximation. In light of the above discussion, it is impossible to calculate the SPT cover number of the graphs in hand, as they have more than 11,00 nodes.

However, we did show that the known portion of the AS connectivity map is characterized by a large number of BFS trees needed to cover it. We already showed that finding the optimum number of such trees in NP-hard and even approximating this number is computationally hard. An interesting theoretical question here is whether the SPT coverage number is monotone with respect to subgraph relation. That is, if \( G' \) is a a subgraph of graph \( G \) with the same node set, and \( G' \) has a SPT coverage number of \( k \), can \( G \) be covered by less than \( k \) shortest path trees? If this is the case then clearly the full (unknown) AS connectivity graph needs at least as many trees as DASC. Unfortunately, the monotonicity does not hold in general. There are simple examples for graphs than cannot be covered by less than \( 2k - 3 \) shortest path trees, but adding \( k - 1 \) edges to them, results in graph that can be covered by \( k \) shortest path trees. Such an example is where 2 cliques of size \( k \) are connected via a single link. It is not hard to verify that \( 2k - 3 \) shortest path trees are needed, but if we connect each of the remaining \( k - 1 \) nodes of each clique with the appropriate node in the other clique, we get a graph that can be covered by \( k \) shortest path trees.

Yet, the results of this section indicate that this is not the case here, and the full AS graph probably needs at least as many trees as the DASC graph. This brings us to the next interesting question: What makes a graph have this property? Before we address this question (in Section IV) we need to go back to the size of the AS connectivity graph.

![BFS Tree Cover Example](image-url)
As described earlier, the main question we are addressing in this paper is the overall size of the AS connectivity graph. We want to be able to do this 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 ASs is known (was 11600 in 2001, and about 13000 in 2002), 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.

The first approximation is to use the evidence provided in [2] for the fact that a considerable amount of data is missing from the BGP routing table information collected from Oregon, SwiNOG, and 10 other ASs. In Figure 7 in [2], the local (correct) number of neighbors is compared to the number of neighbors as reflected by the collected data. The number of edges in the collected BGP tables is reported to be 26,324 (Table 3 in [2]). If we try to estimate the least squares regression line according to this data we get that the ratio between the overall number of edges and the overall number of edges that were in the collected data graph is about 1 to 3.3. This implies that the collected data only covers about 1/3.3 of the full graph, and thus the number of edges in the full graph is about 26.00 times 3.3 which is 85,800 edges.

We have to note that this number is a very rough approximation due to several reasons. To start with, [2] does not provide the full numerical data. A more inherent problem is that the sampling space here is very small, and it may be correlated with the results. I.e., it could be the case that the partial information collected from the BGP tables have more accurate information regarding edges connected to heavy nodes (nodes with a large number of neighbors) or to nodes representing ASs in the USA. In such a case the accuracy of the results depends on these dependencies.

A second try to approximate the number of AS peering relations is to use the data added by the BGP information in the SwiNOG, and 10 other ASs. The idea is to look at the information added from each one of the trees of the 10 additional ASs. As described in great detail in the previous section, the amount of information added by the first 10 trees, does not reflect the overall graph size and the result depends strongly on the graph at hand. Clearly we have no information about the structure of the full graph. However we can create trees from the 10 ASs described in Table 1 in [2], since we know that their BGP information was added to the DASC graph. Thus adding the trees from the DASC is identical to adding the trees from the full AS connectivity graph which is unknown yet. However the fact that the union of these trees covers 22958 edges does not help much in revealing the overall size of the full graph, since as we showed in the previous section the amount of edges covered by trees depends heavily on the structure of the full graph.

Next we address the process of adding the star information from the Internet Routing Registry (IRR) information. As described in the previous section we model this process by covering edges by stars. We can estimate the size of the connectivity graph by using the degree of the stars, and assuming that the sample space is representative. If we look at the RIPE data base altogether there are 43,498 edges from 7,521 nodes, an average of 5.78 edges per node, or a total of 69,500 for the 12,000 nodes consider in the 2001 DASC. However, the authors in [2] consider some of the information void, and filter out, in two different methods, termed sanity check 1 and 2 some of the edges. If we consider the objects after the sanity checks we get an average degree of 4.83 for the first sanity check and 15.93 for the second. Indeed the authors of [2], explain that the two different sanity check methods differ in the way they favor nodes with large degree. These average outdegrees put the total size estimation between 58,00 and 190,00 edges, which indicates that the above estimations suffer dramatically from bias in the measurement process. That is, the results are too sensitive to the filters used, and thus inaccurate.

We use this data then in a different way that results in a more robust and thus accurate estimation. This 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. Consider again tables 2 and 3 in [2]. We take the “discovered” part of the graph to be 27,899 edges in the Oregon+RSs+LG graph of Table 3 in [2]. When the 8965 edges of Sanity check 1 are added we get a total of 32,903 edges. This means that 3961 out of the 8965 edges where already covered by the discovered part. This means that the probability of an edge to be discovered by the BGP data is 0.44, and thus the total number of edges can be approximated by $1/0.44 \times 27,899 = 63,144$. If we do the same calculation for Sanity check 2, we get that the probability of an edge to be discovered by the BGP data is 0.4, and thus the total number of edges is approximated by 68,690.

We refer here, and throughout this paper to the journal version of this paper available at http://topology.eecs.umich.edu/.
<table>
<thead>
<tr>
<th>Type of sub graph</th>
<th>Size of spanning trees</th>
<th>Size of sub graph</th>
<th>Size estimation I (degree) average/var</th>
<th>Size estimation II (edge overlap) average/var</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Edges</td>
<td>21042/144</td>
<td>10163/94</td>
<td>-</td>
<td>50790/476</td>
</tr>
<tr>
<td>Random Stars</td>
<td>21042/144</td>
<td>11913/94</td>
<td>50861/333</td>
<td>50832/733</td>
</tr>
<tr>
<td>Biggest Stars</td>
<td>21042/144</td>
<td>18113/0</td>
<td>80688/0</td>
<td>49442/558</td>
</tr>
<tr>
<td>Smallest Stars</td>
<td>21042/144</td>
<td>5944/0</td>
<td>24472/0</td>
<td>42199/390</td>
</tr>
</tbody>
</table>

**TABLE I**

**Random Graph of size 50812**

<table>
<thead>
<tr>
<th>Type of sub graph</th>
<th>Size of spanning trees</th>
<th>Size of sub graph</th>
<th>Size estimation I (degree) average/var</th>
<th>Size estimation II (edge overlap) average/var</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Edges</td>
<td>19927/600</td>
<td>11997/92</td>
<td>-</td>
<td>59896/690</td>
</tr>
<tr>
<td>Random Stars</td>
<td>19927/600</td>
<td>14018/393</td>
<td>59522/1730</td>
<td>60219/2016</td>
</tr>
<tr>
<td>Biggest Stars</td>
<td>19927/600</td>
<td>37789/0</td>
<td>179220/0</td>
<td>50829/356</td>
</tr>
<tr>
<td>Smallest Stars</td>
<td>19927/600</td>
<td>7501/0</td>
<td>30000/0</td>
<td>55919/603</td>
</tr>
</tbody>
</table>

**TABLE II**

**Graph Barabasi-Albert of size 59985**

This estimation seems to be much more robust with respect to the degree of the nodes in the sampling space. However, there are other factors that may have a strong impact on its accuracy. The first factor is the correlation (or negative correlation) between the area that is covered by the BGP data and the area that is covered by the IRR data. That is, if most BGP information comes from the USA, i.e. the trees are mostly rooted in the US, it is expected that many more edges in the US will be discovered, while edges that are further away from the sources are less likely to be discovered (see [1] for further discussion of this point). Thus, if we do indeed have a negative coloration, the true coverage of the BGP data should be better than 0.4 and the total estimation is too big.

Another factor that has a strong affect on the above estimation is the fact that the edges in our sampling space are indeed dependable. The fact that they all come from a set of stars implies that for each star the known data base contains at least one edge since it is built from trees. In other words, if our information comes from 1500 nodes, then 1500 edges must be covered by the discovered edges, since in the BGP data all ASs are connected. This factor has a greater affect if there are many stars with small degrees, and it becomes smaller as the average degree increases.

In order to verify that this method indeed predicts the actual value of the full graph, and the effect of these two factors is not critical, we conducted simple experiments with synthetics graphs we produced. The results are reported in Table I and Table II. For each of the two graphs, a BA graph with 12,000 nodes and 59,985 edges, and a random $G(n, p)$ graph with 12,000 nodes, and 50,812 edges we preformed the following tests. We created a “known topology” part by choosing a small number (6 in the BA case and 6 in the $G(n, p)$ case), and considering the union of their BFS tree edges. The size of this set and its variance are reported in the Size of spanning trees column. We then created an independent subgraph using 4 different methods. The first is a random independent set of edges, the rest are created using stars as described before. That is we chose a node and added to the set all edges connected to this node. The nodes were chosen either with random degree, large degree, or small degree. For each such selection we conducted a set of 100 Experiments, and reported the estimated overall graph size using the average degree, and the partial covering as discussed in the previous section. Note that in order to compute average degree, we need to consider the direction of the edge with respect to the star, and thus the degree is bigger than the number of edges divided by 1500.

The approximated graph sizes and the variances are reported for each set of tests. One can see that the set of random edges, gives (for both graphs) an excellent size estimation of less than 0.1% error. However, when we used random stars, the quality of the estimation drops for the BA graph and stayed very good for the random graph. The estimation for the BA graph is still about 0.4% from the actual value but the variance increases by a factor of 3 to about 2%. The reason for this phenomenon is the deviation in the degree of nodes in the BA model (and also in real AS topology). The size estimation according to the degree of random stars is very good (as expected from the Law of Large Numbers), but the variation is large, and if we consider the size estimation for biased stars we see that it hardly provides any meaningful information. On the other hand, using our partial covering techniques, the estimations are still fairly good for both choices, and the variances become much smaller.
These results indicate that our finding regarding the size estimations of the AS 2001 connectivity data is accurate. We can conclude then that the total number of edges should be around 70,000 edges and the average number of neighbors is about 11.6. Clearly we do not claim that this number is the exact one, and in fact there is no definition to the exact size of the AS connectivity graph. As pointed out before the peering relation keep on changing and the system we are trying to measure is ever evolving. Therefore the number specified above is just a general approximation of the size relevant to the 2001 data. The 2002 data mentioned before contains more edges (about 13,000), and the estimated sizes are indeed about 10% higher.

IV. REVISITING MODELS OF THE AS CONNECTIVITY MAP

Now, once we know the size (i.e. number of peering relations) of the AS connectivity graph we return to the problem of trying to model it. As explained before we want to model the full graph, and use the information we have that reveals only part of it.

If we go back to the data of Section II we see that for all graphs we checked 50 BFS trees discover almost all the edges. In Table III we presented data regarding several graphs. For each such a graph we present the number of edges discovered by 50 BFS tree, the number of edges discovered by 1500 stars, and the number of edges covered by both trees and stars. One can see that for all graphs almost all edges were revealed by the trees. In such a case, the Oregon Data itself should reveal the full connectivity graph, and the stars process would had add a negligible amount of vertices.

We already saw that in a full clique, all edges are covered exactly by two trees, so it may be the case that the DASC graph contains a very large clique. However, examining the degree distribution graph shows that there are less than 80 nodes of degree greater than 80, and thus the maximal clique size is no more than 80.

Another interesting structure that creates edges that are covered only by the two trees, rooted at the edge end points, is a 3 clique (triangle) that is connected to the rest of the graph via only one edge, or a small clique in which all nodes are connected to an external (usually heavy) node. This is the case where a small number of local ISPs are connected to a regional ISP, but also have direct peering among themselves. In such a case only trees rooted at one of the small ISP AS’s can detect the local edges. It is enough that 200 such small local cliques exist to bring the number of needed trees to 300-400. Of course we cannot say that we know that this is the situation, but many of the findings indicate that this may well be part of the reason for this characterization of the DASC graph. In order to further examine the source for this phenomenon, we reexamine some of the assumption we had made.

A. Hierarchy and policy based routing trees

Recall that routing in BGP is policy based and AS’s advertise paths according to their local policy. In this section we refer to this policy and show how it affects our model. In general, AS’s do not reveal their BGP policy and the types of their peering relationship with their neighbors. This kind of information cannot be retrieved directly from the gathered information. In [3], [4], and [5] the authors have discussed the peering relationship between AS’s and their topological structure. In [5] the authors classified the AS’s into four hierarchy groups according to their degree in the AS graph. In [4] the author classified the peering relationship between AS’s according to their degree. Thus, AS’s with similar degrees have a peer-peer relationship while other AS’s are connected by a customer-provider relationship. In [3] the authors classified the AS’s into five hierarchy groups using a multiple vantage point. In this hierarchy AS’s
from the same groups are connected by a peer-peer relationship while AS's from different groups are connected by a customer-provider relationship. In addition, they quantify the number of AS's in each group. In [16], and [17] the authors examined the routing policy that is used by AS's. They presented guidelines that are based on the commercial relationship between AS's. According to these guidelines, permitted paths do not include so called valleys nor steps.

As we discussed above, we consider two types of data collection processes. The first one is data that is obtained by BGP routing table and it is reflected in our model as a set of trees. Each tree is the shortest path, under the restriction of a policy, from a specific AS to all other AS's in the graph. Usually the policy of the AS's cannot be obtained and many AS's do not reveal their policy. Nevertheless, since the policy usually reflects the commercial relationship between neighbor AS's we use the guidelines that have been presented in [16] and [17].

Obviously, the full set of BGP routing tables or the full set of stars (i.e. routing tables or stars from all the AS's respectively) would give us the overall picture regarding the AS connectivity graph. Nevertheless, as we described in Section II only a couple of dozen AS's are willing to expose their BGP routing table while less than 20% are willing to expose their peering relationship.

In this section we reconstruct the process of data collecting on several graph models and we compare the results to the real behavior as it was reflected from the real process. However, instead of using BFS trees we use trees that reflect policy routing based on the hierarchical structure described above.

To do that, we divide the set of nodes $V$ into four hierarchy groups according to the vertex degree. Moreover, the number of AS's in each group increases in a log scale (see Table VI in [3]). Thus, our graphs consist of 11,000 nodes that are divided in the following way: 10 AS's (that have the highest degree) are in level 1, 90 AS's are in level 2, 900 AS's are in level 3, and 10,000 AS's (that have the lowest degree) are in level 4. For each model we generated three graphs that have 40,000, 70,000 and 100,000 edges respectively. We span each graph by 51 trees as described above. These trees reflect the Oregon route views database (see [8]) plus trees of other 10 AS's (see [2]). Then we span the graph by 1500 stars that simulate the IRR database (see [13]). We compare the simulation results (i.e. the subgraphs and the union of these subgraphs) to data that was presented in [2].

According to the data from [2], 51 BGP trees reveal 26324 peers from the AS's connectivity graph. For the second phase, there are two methods for collecting the stars information that differ in the way they filter invalid peering. The first one consist of 1855 stars that reveal 8965 peers. From these peers 6579 (i.e. 73%) are new, namely they were not revealed by the trees. The second, consist of 1558 stars that reveal 24821 peers. From these peers 16315 (i.e. 66%) are new.

Consider Table IV. In the Random graphs only a small amount of peers were revealed by the trees (5%-10%). Moreover, these trees revealed only a small portion of the vertices in the graph. One should observe that since the routing is policy based, connectivity does not impose reachability, thus even though the graph is connected there are many nodes that cannot be accessed. Since, the BGP trees did not reveal many peers, most of the peers that were revealed by the stars were new (88%-95%, depending of the graph size). The high percentage of new peers revealed by the stars of this model, and the small amount of vertices and peers revealed by the trees do not fit the real data.

In the Barabasi-Albert graphs (and similarly, in the Scale Free graphs) most of the vertices were revealed by the trees and the number of peers that were found by them was more than 50%. In this model new AS's prefer to be connected to AS's with the highest degree, thus the number of customer-provider peers in this model is bigger than the one in the random graph. This kind of connectivity enable to reach most of the vertices. Still, a large portion of the peers was

<table>
<thead>
<tr>
<th>Graph</th>
<th>BGP Trees</th>
<th>% vertices</th>
<th>Stars</th>
<th>Trees+Stars</th>
<th>Original Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>2038</td>
<td>17</td>
<td>16893</td>
<td>11649</td>
<td>19991</td>
</tr>
<tr>
<td>BA</td>
<td>22729</td>
<td>97</td>
<td>10953</td>
<td>27177</td>
<td>41888</td>
</tr>
<tr>
<td>Waxman</td>
<td>10240</td>
<td>57</td>
<td>10551</td>
<td>18552</td>
<td>44000</td>
</tr>
<tr>
<td>Scale Free</td>
<td>26244</td>
<td>99</td>
<td>8653</td>
<td>29921</td>
<td>40548</td>
</tr>
<tr>
<td>BA+Waxman</td>
<td>23686</td>
<td>88</td>
<td>8965</td>
<td>24821</td>
<td>42639</td>
</tr>
<tr>
<td>Real Data</td>
<td>26324</td>
<td>97</td>
<td>8965</td>
<td>32903</td>
<td>68700 (est.)</td>
</tr>
</tbody>
</table>

TABLE IV
RECONSTRUCTING PROGRESS
not revealed by these trees. Most of these peer connect AS’s in the fourth level and they have peer-peer relationship. According to our policy model these kind of peers can be found only from their endpoint. The stars in this model found 45%-49 new peers which is quite low comparing to the real data.

In the Waxman model 23%-27% of the peers were found by the trees and 73%-75% of the peers that were found by the stars were new. Although only 85% of the vertices were found in the biggest graph (i.e. the graph that has 99000 peers), all other data is similar to the real data.

In the previous section we showed strong evidences to the fact that the size of the AS connectivity graph is about 70,000 peers. Thus, we would like find a graph with the same size that reconstruct the data gathering process. Although the BA model reveal most of the AS’s, it seems that it finds too much peers by trees. In addition, only 45% of the peers that found by the stars are new compare to 70% in the real data. On the other hand, in the Waxman graph only 72% of the vertices and 17159 peers were found by the trees. A combination of these two models moderate the numbers of peers revealed by the trees (compare to the BA model) while almost 88% of the vertices were found. In addition, 67% of the peers that were found by the stars are new and the total number of peers is 35685. This graph was generated in two stages. In the first stage, edges were added according to the BA model while in the second stage edges were added according to the Waxman model.

B. Degree Distribution

In the last section we evaluated graph models by reconstructing the data collecting process. Nevertheless, This reconstruction by itself is not sufficient. In particular, the vertex degree distribution of the graph models that we examined do not fit to the distribution of the real data.

Many models that intend to describe the AS connectivity map focus on the degree distribution of the graph. In particular, most of these models impose that the distribution of the vertex degree of the graph is power-law. In [2] the authors presented a degree distribution of a real data that was collected mainly from Oregon data base and from IRR. Using the same database that has been used in [2] we reconstruct the graph of the degree distribution (see the upper part of Figure 10). We also present the graph of the current vertex degree distribution using a new data that has been gathered during 2004 (see the bottom part of Figure 10). In these graphs one can see that the data that was collected only from Oregon database fits the power law. In contrast, the combined data that was collected from Oregon and IRR has a curve. In this section we explain this results. In particular, we show that if the degree distribution of a subgraph follows the power law, then the degree distribution of the full graph does not necessarily follow this law. In addition, we use the collected information to give a better insight regarding the AS connectivity map.

In Table IV we show that in many graph models a set of couple of dozens BGP trees covers only small portion of the edges in the graph (in contrast to BFS trees). These results are the outcome of the BGP policy that prohibit many paths in the graph. In Table V we check how many edges (out of the edges that were revealed by the BGP trees) have peer-peer relationships and how many edges have customer-provider relationships.

From Table V one can see that most of the edges that were revealed by the BGP trees have customer-provider relationships while only negligible amount of edges have peer-peer relationships. This result can be easily explained by the BGP policy in which in most cases paths that contains peer-peer edges are prohibit since they contain what so called steps (see [16], and [17]). Moreover, in the power-low models (i.e. Barabasi-Albert and Scale Free models) almost 95% of the customer-provider edges are revealed by the trees (compare to 5% of the peer-peer edges).
Using this insight we can conclude that the Oregon data base, consisting of BGP trees, contains most of the customer-provider peering relationship in the AS connectivity map. Moreover, it contains solely this kind of peers (i.e. the amount of peer-peer edges in this database is negligible). Since the vertex degree distribution of Oregon database fits the power law, it imposes that in the model that describes the AS’s connectivity map, the customer-provider peering subgraph fits the power law.

In Figure 11 we demonstrate that the fact that a subgraph follows the power law distribution does not imply that the full graph does so too. In this figure the full graph is a superposition of two subgraphs. The first subgraph is a scale free graph that is consists of 25000 customer-provider edges while the second graph is a random graph consisting of 45000 peer-peer edges. In this graph (and as we observed above) 51 BGP trees discovered most of the customer-provider edges while peer-peer edges are almost not revealed. Thus, the subgraph that was spanned by the trees follows the power low distribution. However, the full graph is influenced by the random edges and therefore it does not fit the power law. In addition, the graph consisting of BGP trees and random stars reconstructs the distribution of the full graph in a less extremely way. To compare these synthetic graphs with real data, we plotted in Figure 10 the degree distribution of the 2002 data (both for the Oregon and the full DASCs. We also plotted current (2004) data collected from Oregon Route-views, and IRR. One can observe the same type of behavior, indicating that the rest of the missing links probably follow the same rule and are of peer-peer type.

V. DISCUSSION

The goal of this paper is to reveal more information regarding the AS connectivity map. We showed strong evidence indicating that more than 50% of the links are still missing from all known databases. We also showed that almost all missing links are of peer-peer type, and that the degree distribution of the peer-peer part of the connectivity map is considerably different from the one of the customer-provider part.

Note that the influence of the random subgraph on the overall distribution is especially in the vertices that have low degree since the variance of the vertex degree of the random graph is small.
It should be cleared that no one can provide an exact estimation for the number of the missing links. This is due to the fact that this number is ever changing, and peering relations between ASs keep on changing. However, depending on the specific usage of the AS connectivity map, the information we provided can be used to generate an approximate model that serve the specific need.

REFERENCES