Network Topology Generators: Degree-Based vs. Structural *

Hongsuda Tangmunarunkit
USC-ISI
hongsuda@isi.edu

Ramesh Govindan
ICSI
ramesh@icsi.berkeley.edu

Sugih Jamin
Univ. of Michigan
jamin@eecs.umich.edu

Scott Shenker
ICSI
shenker@icsi.berkeley.edu

Walter Willinger
AT&T Research
walter@research.att.com

ABSTRACT

Following the long-held belief that the Internet is hierarchical, the network topology generators most widely used by the Internet research community, Transit-Stub and Tiers, create networks with a deliberately hierarchical structure. However, in 1999 a seminal paper by Faloutsos et al. revealed that the Internet's degree distribution is a power-law. Because the degree distributions produced by the Transit-Stub and Tiers generators are not power-laws, the research community has largely dismissed them as inadequate and proposed new network generators that attempt to generate graphs with power-law degree distributions.

Contrary to much of the current literature on network topology generators, this paper starts with the assumption that it is more important for network generators to accurately model the large-scale structure of the Internet (such as its hierarchical structure) than to faithfully imitate its local properties (such as the degree distribution). The purpose of this paper is to determine, using various topology metrics, which network generators better represent this large-scale structure. We find, much to our surprise, that network generators based on the degree distribution more accurately capture the large-scale structure of measured topologies. We then seek an explanation for this result by examining the nature of hierarchy in the Internet more closely; we find that degree-based generators produce a form of hierarchy that closely resembles the loosely hierarchical nature of the Internet.

*At USC-ISI, this work was supported in part by the Defense Advanced Research Projects Agency under grant F30602-00-2-055. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Defence Advanced Research Projects Agency. At the University of Michigan, this project was funded in part by NSF grant number ANI-0082287 and by ONR grant number N000140110617. Sugih Jamin is further supported by the NSF CAREER Award ANI-9734145, the Presidential Early Career Award for Scientists and Engineers (PECASE) 1998, and the Alfred P. Sloan Foundation Research Fellowship 2001. Additional funding is provided by AT&T Research, and by equipment grants from Sun Microsystems Inc. and Compaq Corp.

SIGCOMM'02, August 19-23, 2002, Pittsburgh, Pennsylvania, USA.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Network topology*; I.6.4 [**Simulation and Modeling**]: Model Validation and Analysis

General Terms

Performance, Measurement

Keywords

Network topology, hierarchy, topology characterization, topology generators, structural generators, degree-based generators, topology metrics, large-scale structure

1. INTRODUCTION

Network protocols are (or at least should be) designed to be independent of the underlying network topology. However, while topology should have no effect on the *correctness* of network protocols, topology sometimes has a major impact on the *performance* of network protocols. For this reason, network researchers often use network topology generators to generate realistic topologies for their simulations.[1] These topology generators do not aspire to produce exact replicas of the current Internet; instead, they merely attempt to create network topologies that embody the fundamental characteristics of real networks.

The first network topology generator to become widely used in protocol simulations was developed by Waxman [48]. This generator is a variant of the classical Erdos-Renyi random graph [6]; its link creation probabilities are biased by Euclidean distance between the link endpoints. A later line of research, noting that real network topologies have a non-random structure, emphasized the fundamental role of *hierarchy*. The following from [51] reflects this observation:

[1] It should be noted that sometimes topology generators are used to tickle subtle bugs in protocols. However, for this purpose the emphasis is not on finding realistic topologies but on finding hard cases.

> ...the primary structural characteristic affecting the paths between nodes in the Internet is the distinction between stub and transit domains... In other words, there is a *hierarchy* imposed on nodes...

This reasoning quickly became accepted wisdom and, for many years, the network generators resulting from this line of research, Transit-Stub [10] and Tiers [14], were considered state-of-the-art. In what follows, we will refer to these as *structural* generators because of their focus on the hierarchical structure of networks.

These structural generators reigned supreme until the appearance of a seminal paper by Faloutsos *et al*. [17] in 1999. In that paper, the authors used measurements of the router-level and AS-level Internet graphs—the former having routers as nodes and the latter having ASs as nodes—to investigate (among other issues) the node degree, which is the number of connections a node has. They found that the degree distributions of these graphs are power-laws.[2]

The aforementioned structural generators do not produce power-law degree distributions. Many in the field seem to have concluded that this disparity, *by itself*, proved that structural generators were unsuitable models for the Internet. Subsequently, there have been an increasing number of proposals for topology generators that are designed primarily to match the Internet's degree distribution and do not attempt to model the Internet's hierarchical structure; for example, see [23, 28, 2, 31, 1, 8]. These *degree-based* topology generators embody the implicit assumption that it is more important to match a certain local property—the degree distribution—than to capture the large-scale hierarchical structure of the Internet. The rapid adoption of these degree-based generators suggests that this belief, while not often explicitly stated, is widely held.

This paper starts with a very different premise. We believe that it is more important for topology generators to accurately model the large-scale structure of the Internet (such as its hierarchical structure) than to faithfully reproduce its local properties (such as the degree distribution). In particular, we believe that the scaling performance of protocols will be more effected by these large-scale structures than by purely local properties.

While we cannot prove the correctness of our belief, this paper is devoted to exploring its implications. That is, we wish to determine which topology generators—degree-based or structural—produce better models of the large-scale structure of the Internet. Some have argued that this question is vacuous, because networks that do not match local properties of the Internet cannot possibly match its large-scale structure. But we claim the two properties—local and global—are separable. Consider, for example, a tertiary tree, a two-dimensional grid, and degree-four random network; each of these networks have exactly the same degree distribution (all nodes having degree four) but they obviously have very different large-scale structure. Similarly, one can define trees with any desired degree distribution (in particular, the one matching the Internet's degree distribution), and yet not alter the tree-like large-scale structure.

Thus, we believe, in contrast to much of the research community, that it is still an open question as to which network topology generators best model the Internet. This paper is devoted to addressing this issue. More specifically, after reviewing related work in Section 2, this paper proceeds to ask two questions.

[2]There is some disagreement about whether these are true power laws or are Weibull distributions or perhaps something else. For our purposes we don't care about the exact mathematical form of the distribution, merely that it can be closely approximated by a power-law or similar very long-tailed distributions.

Question #1 *Which generated networks most closely model the large-scale structure of the Internet?* To answer this question we must first determine what the Internet is and then decide how to measure the degree of resemblance between it and the generated networks. As we describe in Section 3.1 we use two representations of the Internet. The first representation is at the Autonomous System (AS) level, where ASs are nodes and edges represent peering relationships between ASs. We use BGP routing tables to derive the AS graph. The second representation is at the router level, where routers are nodes and an edge indicates that the corresponding routers are separated by one IP-level hop. The router graph comes from the SCAN project [20] which uses a series of traceroute measurements to map the Internet. The router graph represents the Internet at a much finer level of granularity, and has roughly 17 times more nodes and links than the AS-level graph. While they both are representations of the Internet, it isn't clear that, as graphs, they would have much in common. Thus, we consider these two *measured* graphs as distinct entities in our analysis, and separately ask which generated networks most resemble the AS-level graph and which most resemble the router-level graph. We should note that the structural topology generators were originally intended to model the router-level graphs, while the degree-based generators were not explicitly targeted at one or the other level of granularity.

Even though our topology data is the best we could obtain, it is clear that both of these measured graphs—the AS graph and the router graph—are far from perfect representations of the Internet. Not only are they subject to errors and omissions, but they also only reflect the topology and do not contain any information about the speed of the links. We do, however, approximately model an aspect of reality that has been shown to impact path lengths [43, 38] in Internet topologies—policy routing.

To measure the properties of the Internet graphs and the generated graphs, we use a set of three topology metrics described in Section 3.2. These metrics are intended to capture the *large-scale* structure of networks. Our methodology for picking these metrics was simple and, admittedly, ad-hoc. We computed *eight* different topology metrics (either reported in the literature or of our own definition) on the network topologies. Of these, we find that three basic metrics maximally distinguish our topologies: the addition of more metrics does not further distinguish between our topologies, but the removal of one or more of these three blurs some distinctions. Thus, the conclusions we draw are supported by all eight metrics (not all of our own design), but can be presented with only three of them. For space reasons, we present these basic three metrics and refer the reader to [42] for a complete discussion of all metrics.

While we are not aware of extensive prior work in the design of metrics to measure large-scale network properties, and while we have borrowed liberally from the work that exists, we fully recognize that our metrics may not adequately characterize network topologies and that additional work is urgently needed in this area. Moreover, the distinctions we draw from these metrics are rather qualitative in nature (we often are left asking *do these curves have roughly the same shape?*) and thus are subject to different interpretations.

These caveats notwithstanding, we use these metrics to compare the generated and measured networks. Our results, presented in Section 4 and augmented by additional results [42], suggest two findings. First, we find that the AS and router graphs have similar properties. One might expect (as did we) that, since they describe the Internet at such different levels, the AS and router graphs would have quite different characteristics; our results indicate otherwise. Second, we find that the degree generators are significantly better at representing the large-scale properties of the Internet, at both the AS and router levels, than the structural generators. Since our metrics measure large-scale structure and the degree generators focus only on very local properties, we expected the structural generators would easily be superior; again, our results indicate otherwise. This leaves us with the seeming paradox that while the Internet certainly has hierarchy, it appears that the large-scale structure of the Internet graphs is better modeled by network generators that completely ignore hierarchy! Resolving this paradox leads us to our second question.

Question #2 *Do the degree-based generator produce networks with hierarchy and, if so, how?* In Section 5 we introduce a measure of hierarchy, and use it to investigate the nature of hierarchy in the generated and measured graphs. We find that while the degree-based generators do not explicitly inject hierarchy into the network, the power-law nature of the degree distribution results in a substantial level of hierarchy—not as strict as the hierarchy present in structural generators, but significantly more hierarchical than, say, random graphs. This relatively loose form of hierarchy, produced merely by the presence of the power-law degree distribution, more accurately reflects the nature of hierarchy in the Internet than the strict hierarchy produced by the structural generators.

In summary, then, we find that the prevailing wisdom that degree-based generators are better models for Internet topologies, to which we had taken exception, is indeed correct. However, these degree-based generators are better models of the Internet not just because they slavishly imitate the degree-distribution but because this degree distribution (and the fairly random connection of nodes) leads to a loose form of hierarchy very similar to that in the Internet.

2. RELATED WORK

We have already mentioned several important areas of related work: the Waxman, Transit-Stub and Tiers topology generators, and Faloutsos *et al.*'s observations of power-law degree distributions in the Internet. We have also mentioned in passing several new degree-based generators [23, 28, 2, 31, 1]. They all attempt to generate networks with power-law degree distributions, but differ in the way in which nodes are connected. We describe some of these generators in slightly more detail in [42].

Perhaps closest in spirit to the work presented in this paper is the pioneering exploration of topology properties by Zegura *et al.* [51]. Their study considered various properties (biconnectivity and various kinds of network diameters) of random graphs (and variants thereof) and structural generators. We follow their lead but extend their study using a larger collection of metrics, adding measured networks and degree-based generators, and explicitly analyzing the degree of hierarchy. More recently, Barabasi *et al.* [3] have attempted to quantify the attack and error tolerance of random graphs and real-world "scale-free" networks. Finally, van Mieghem *et al.* [45] have shown that the Internet's hop count distribution (the distribution of path lengths in hops) is well modeled by that of a random graph with uniformly or exponentially assigned link weights. Some of the topology metrics used in our paper are based on the metrics introduced in these papers.

Also directly relevant is the work of Medina *et al.* [29]. They too compare random graph generators (such as Waxman), and hierarchical generators (such as Transit-Stub) to degree-based generators (such as the BRITE generator [28]). Their metrics for comparison include the tests in [17] for power law exponents of the degree distribution, the degree rank, the hop-plot and the eigenvalue distribution. They conclude that the degree and degree-rank exponents are the best discriminators between topologies among the metrics they considered. Using these metrics, they conclude that the BRITE generator was better than the Transit-Stub and Waxman generators in modeling the Internet. However, using the degree and degree-rank exponents as metrics means that topologies are evaluated solely on how well their degree distribution matches the degree distribution of the Internet. It is well known that Transit-Stub and other structural generators do not produce power-law degree distributions, and so it is no mystery that BRITE and other degree-based generators do a better job of matching the degree and degree-ranked exponents. However, the question we pose in this paper is: which class of generators most closely resemble the Internet *when looking at the large-scale properties of the Internet*? We believe this question has not been addressed by the work in [29] or elsewhere in the literature because networks with similar degree distributions can have very different large-scale properties (Section 1).

Two other recent pieces of work examine local properties of network topologies. Bu and Towsley [8] find that degree-based generators differ significantly in their clustering coefficients [47]. Their work proposes an alternative degree-based generator that more closely matches the clustering behavior of the measured AS graph. For completeness, we have incorporated both the clustering metric and the proposed generator in our analyses (Section 4). Vukadinovic *et al.* [46] evaluate the Laplacian eigenvalue spectrum of a variety of graphs, and conclude that the multiplicity of eigenvalues of value 1 differentiates AS graphs from grids and random trees. However, as claimed in [46], this measure of the spectrum reflects purely local properties of the graph (the number of degree 1 nodes, the number of nodes attached to degree 1 nodes *etc.*), while our work focuses on the large-scale structure. However, their result is consistent with our findings (and with the commonly held intuition that the AS graph is neither mesh-like nor tree-like).

Also relevant to our work is recent work on the analysis of graph measurements. Broido and Claffy [7] find that various properties of real-world graphs, including the degree distribution, are well-modeled by a Weibull distribution. Using extensive measurements of the AS graph, Chang *et al.* [12] show that the degree distribution of the AS graph deviates significantly from a strict power-law fit. As we have discussed in Section 1, our work merely assumes that the degree distribution is well approximated by a heavy tail and does not depend on the exact mathematical form of the distribution. Finally, Magoni *et al.* [27] study various graph theoretic and time evolution properties of the AS topology.

Our work would not have been possible without developments in Internet router-level topology discovery. Early work in this area used traceroutes from a small set of sources to several thousand hosts to compute a router-level map [32]. Subsequent work improved the coverage of the Internet address space by randomly selecting IP addresses [39], randomly selecting addresses from route entries in BGP tables [9], using a precomputed set of Web sites [13], or using heuristics to infer addressable parts of the IP space [20]. This last work also documents several techniques for improving completeness of the inferred topologies.

Several papers have addressed the impact of topology on protocol performance. For example, Phillips *et al.* [35] showed that graphs with exponentially increasing neighborhood sizes (*i.e.*, number of nodes within a certain radius increases exponentially with radius) approximately obey the Chuang-Sirbu multicast scaling law. In closely related work, Almeroth and Chambers [11] considered a variety of metrics for the efficiency of multicast trees. Wong and Katz [49] found that the amount of multicast state from randomly placed receivers differs qualitatively with different topologies. Radoslavov *et al.* [36] found similar results for other kinds of protocol performance questions.

Although there is a large literature on routing hierarchies, we are not aware of much work that has attempted to measure (as opposed to create, or utilize) hierarchy in network topologies. Two notable, and related, examples [18, 40], describe techniques for inferring hierarchical relationships (*e.g.*, provider-customer) in the AS topology. The latter work also classifies ASs into a five-level hierarchy.

Somewhat orthogonal to the questions considered in this paper is recent work attempting to explain the origin of power-law degree distributions. Ferrer i Cancho *et al.* [22] and Fabrikant *et al.* [16] have independently shown that, under certain conditions, power-law degree distributions can arise as a consequence of optimizing an objective function. Tangmunarunkit *et al.* [41] argue that, for the AS graph, the high variability of the degree distribution follows from the high variability of the distribution of AS sizes.

There has also been significant work in the non-networking literature exploring the properties of real-world networks. We do not intend to be exhaustive in our coverage of this work, but will mention some oft-referenced work. Watts and Strogatz [47] found that many real-world networks, such as the actor collaboration network and a section of the power grid, are well-modeled by the small-world phenomenon. Kleinberg *et al.* [25] analyzed properties of the World-Wide Web graph and proposed a new family of random graph models. Aiello *et al.* [1] proposed a random graph model for massive graphs and showed that this model captures some aspects of the AT&T call graph. Our work has been influenced by some of this work, but focuses primarily on communication network topologies.

3. NETWORKS AND METRICS

We now describe the topology generators and measured networks we analyze, and the set of topology metrics we use to do so.

3.1 Networks

We analyze three categories of network graphs: measured networks, generated networks, and canonical networks.

3.1.1 Measured Networks

We use two measured network topologies. Our first is the *AS* topology, representing inter-autonomous system (AS) connectivity, obtained from AS path information in backbone BGP routing tables. Nodes in this topology represent ASs, and links represent peering relationships between them. The particular topology we present in this paper was obtained from the Route Views archive (`routeviews.org`).

Our second measured topology is the Internet router-level (*RL*) topology. This is derived by inferring router adjacencies [20] in the Internet from traceroutes to carefully chosen sections of the IP address space. Nodes in this topology represent routers, and links connect routers that are one IP-level hop from each other. In passing, we note that this definition of a link does not distinguish shared media from point-to-point links. The former usually appear as completely connected subgraphs in the network topology.

Although these topologies are related, they reflect Internet connectivity at rather different scales. For example, the AS topology abstracts many details of physical connectivity between ASs and each AS represents a grouping of several (sometimes hundreds) topologically contiguous routers. Thus, these two graphs could have had very different properties, but, as we show in Section 4, they behave quite similarly with respect to our topology metrics.

Both these topologies are incomplete, to different degrees. They may not capture all the nodes in the network and, for the nodes that do appear in the topology, they may not include all adjacencies at each node. We hope, however, that the qualitative conclusions we draw in this paper will be fairly robust to minor methodological improvements in topology collection. A more serious problem is that these measured networks merely represent connectivity between nodes and links. In particular, neither the RL nor the AS graph contains any indication of the capacity of the underlying transmission link (or shared medium). Although techniques for estimating link capacities along a path are known ([15, 26]), they are reported to be fairly time consuming and, to our knowledge, no one has attempted to annotate the router-level graph of the entire Internet with link capacity information. We don't know how our conclusions would change if such an annotated graph were available.

These topologies are also, obviously, time varying. We have computed our topology metrics for at least three different snapshots of both topologies, each snapshot separated from the next by several months.[3] We find that the qualitative conclusions we draw in this paper hold across these different snapshots. Finally, we have also been careful to incorporate the effects of policy routing in computing our topology metrics. We use a variant of a simple routing policy (Section 3.2.1) that has been shown to match actual routing path lengths reasonably well [43]. In Section 4, we describe the impact of policy on our conclusions.

3.1.2 Generators

We consider three classes of network generators in this paper. The first category, *random graph* generators, is represented by the *Waxman* [48] generator. The classical Erdos-Renyi random graph model [6] assigns a uniform probability for creating a link between any pair of nodes. The Waxman generator extends the classical model by randomly assigning nodes to locations on a plane and making the link creation probability a function of the Euclidean distance between the nodes.

The second category, the *structural* generators, contains the *Transit-Stub* [10] and *Tiers* [14] generators. Transit-Stub creates a number of top-level transit domains within which nodes are connected randomly. Attached to each transit domain are several similarly generated stub domains. Additional stub-to-transit and stub-to-stub links are added randomly based upon a specified parameter. Tiers uses a somewhat different procedure. First, it creates a number of top-level networks, to each of which are attached several intermediate tier networks. Similarly, several LANs are randomly attached to each intermediate tier network. Within each tier (except the LAN), Tiers uses a minimum spanning tree to connect all the nodes, then adds additional links in order of increasing inter-node Euclidean distance. LAN nodes are connected using a star topology. Additional inter-tier links are added randomly based upon a specified parameter.

Both Transit-Stub and Tiers have a wide variety of parameters. Although we present our results for one instance of these topologies, [42] lists the sets of parameters we have explored. Section 4.4 discusses the impact of our parameter space exploration on our conclusions.

The third category is that of *degree-based generators*. The simplest degree-based generator, called the power-law random graph (PLRG) [1], works as follows. Given a target number of nodes N, and an exponent β, it first assigns degrees to N nodes drawn from a power-law distribution with exponent β (*i.e.*, the probability of a degree of k is proportional to $k^{-\beta}$). Let v_i denote the degree assigned to node i. Solely for the purposes of assigning links between nodes, the PLRG generator makes v_i copies of each node i.

[3] Aug 1999, April 2000 and May 2001 for the RL maps. March 1999, December 2000, April 2000, and May 2001 for the AS maps.

Type	Topology	Number of Nodes	Avg. Degree	Comment
Measured	RL	170589	2.53	May 2001
	AS	10941	4.13	May 2001
Generated	PLRG	9230	4.46	2.246
	Transit-Stub (TS)	1008	2.78	3 0 0 6 0.55 6 0.32 9 0.248
	Tiers	5000	2.83	1 50 10 500 40 5 20 20 1 20 1
	Waxman	5000	7.22	5000 0.005 0.30
Canonical	Mesh	900	3.87	30x30 grid
	Random	5018	4.18	Link prob = 0.0008
	Tree	1093	2.00	k=3,D=6

Figure 1: Table of network topologies used. See [42] for a description of parameters for the generated networks.

Links are then assigned by randomly picking two node copies and assigning a link between them, until no more copies remain.[4] For most of the rest of the paper, we focus almost exclusively on PLRG as the sole degree-based generator. However, the results for other degree-based generators, presented in Section 4.4, are qualitatively similar to those of PLRG.

3.1.3 Canonical Networks

Finally, our study also includes three *canonical* networks: the k-ary *Tree*, the rectangular grid or *Mesh*, and an Erdos-Renyi *Random* graph. We include these admittedly unrealistic networks because they help calibrate, and explain, our results on measured and generated networks.

3.2 Metrics

The goal of topology generators is not to produce exact replicas of the current Internet, but instead to produce graphs whose properties are similar to the Internet graph. In this paper we evaluate the quality of a topology generator by how well its generated networks match the large-scale properties of the Internet (both the AS and RL topologies) as measured by several topology metrics. The hard question, though, is: what properties are relevant to this comparison?

There is no single answer to this question, as the relevant properties may well depend on how the generated networks are being used. Moreover, even for a given purpose it is a matter of judgement as to what network properties are the most relevant. Thus, we recognize that the metrics we chose are in no way *definitive*, but merely reflect our own intuition.

Our list of metrics, which include many that have been reported in the networking literature and some graph-theoretic metrics that have plausible networking interpretations, are listed below:

- Neighborhood size (or *expansion*) [35].
- Resilience, the size of a cut-set for a balanced bi-partition [24].
- Distortion, or the minimum communication cost spanning tree [21].
- Node diameter (or *eccentricity*) distribution [51].
- Eigenvalue distribution [17].
- Size of a vertex cover [33].
- Biconnectivity (number of biconnected components) [51].
- The average pairwise shortest path between nodes in the largest component under random failure (when nodes are removed from the graph randomly) or under attack (when nodes are removed in order of decreasing degree) [3].

After computing these metrics on our topologies, we found that three (expansion, resilience and distortion) formed the smallest set of metrics that qualitatively distinguished our set of topologies into well-defined categories. We describe these metrics in this section, and discuss these qualitative distinctions in Section 4. We present the results for all of our other metrics in [42]. The fact that these three metrics also qualitatively differentiate between our canonical graphs—mesh, tree and the random graph (Section 3.2.1) serves as a simple sanity check for our methodology. Intuitively, we know that these canonical graphs are quite different from each other in ways that would be very important to networks, and therefore it is important that our metrics at least clearly differentiate them.[5]

We made one important assumption in deciding how to compute these metrics on our topologies—that they should be designed to ignore superficial differences, like differences in size. Our two measured topologies differ by an order of magnitude in size, and it is more convenient to compare the two against a set of generated and canonical networks. We describe our approach to this, a technique called *ball-growing*, in the next section.

3.2.1 The Three Basic Metrics

Rate of spreading: Expansion One key aspect of a tree is that the number of sites you can reach by traversing h hops grows exponentially in h. We capture this behavior with our *expansion* metric, denoted by $E(h)$. $E(h)$ is the average fraction of nodes in the graph that fall within a *ball* of radius h centered at a node in the topology. More precisely, for a given originating node v we compute the number of nodes that can be reached within h hops (the reachable set). We calculate the size of the reachable set for each node in the graph, average the result, and then normalize by the total number of nodes in the graph.

This definition is similar[6] to the reachability function described in [35] and to the hop-pair distribution defined in [17]. In fact, [35] has analyzed the expansion of some, but not all, of the topologies described in Section 3.1. We repeat those analyses here for completeness.

For our other metrics we use a technique, called *ball-growing*, based on these balls of radius h. We measure some quantity in a ball of radius h and then consider how that quantity grows as a function of h. This allows us to compare graphs of different sizes. The result of each such metric is not a single value but a function of h, and the dependence on h reflects the behavior of the quantity in question at different scales. We will use this technique in our other two metrics; expansion is merely the measure of the size (in terms

[4]This generator is not guaranteed to give a connected graph although, for reasonable values of β, it produces one large connected component. We pick this connected component for our analyses. Furthermore, this procedure can produce self-loops and multiple links between nodes. We ignore these superfluous links in our graphs.

[5]Many of the other metrics used in the literature are not as successful in differentiating these three canonical graphs.

[6]Unlike [35], $E(h)$ is expressed as a fraction of the total number of nodes in the graph, thus making it easier to compare graphs of different sizes in Section 4.

of the number of nodes that reside in the ball), and our other two metrics will measure other properties of the subgraph that resides within balls of radius h.

Implicitly, in computing balls of radius h, our definition includes all nodes to whom the *shortest path* from the center of the ball is less than or equal to h. For the AS and RL graphs, we extended this in a simple way to account for policy routing. In computing a *policy-induced* ball of radius h, we include all nodes to whom the *policy* path from the center of the ball is less than or equal to h, and only include links that lie on policy-compliant paths to those nodes. To do so, we use a policy model that is slightly more sophisticated than the one reported in [43]. At the AS level, this policy model computes the shortest AS path between two nodes that does not violate provider-customer relationships (an example of a path that would violate these relationship is one that traverses a provider, followed by a customer and then back to another provider). We use the results in [18] to infer provider-customer relationships. To compute the policy path in the RL graph, we first compute the corresponding AS level policy path, and then use shortest-paths within the sequence of ASs to determine a router-level policy path. We discuss policy-induced ball growing in greater detail in [42].

There is an important caveat about ball growing that is worth mentioning. We have said that ball growing allows us to study a graph at different scales. However, for some graphs, computing a metric on balls of different sizes is *not* equivalent to evaluating the metric on graphs of comparable sizes. A random graph is a good example of this; a ball of size N of a random graph may not itself be a random graph. However, balls of radius h from, respectively, a random network of size N and a random network of size $2N$ will be similar, as long as the diameters of both networks is larger than h. This is why we adopted the ball-growing approach.

The expansion metric allows us to easily distinguish the mesh from our other two canonical networks. For a mesh with N nodes, $E(h) \propto \frac{h^2}{N}$ while for the k-ary tree or a random graph of average degree k, $E(h) \propto \frac{k^h}{N}$. Thus, the mesh has a qualitatively lower expansion than the tree and the random graph. In passing, we note that our definition of expansion is different from the traditional graph-theoretic definition of expander graphs[7] which, for reasons we don't have space to explain here, is not appropriate for the task at hand.

Existence of alternate paths: Resilience If you cut a single link in a tree, the graph is no longer connected. In contrast, it typically requires many cut links to disconnect a random graph. Our second metric, *resilience* measures the robustness of the graph to link failures. In its definition we use a standard graph-theoretic quantity: the minimum cut-set size for a balanced bi-partition of a graph. For a graph with n nodes, this is the minimal number of links that must be cut so that the two resulting components have approximately $\frac{n}{2}$ nodes. We define the resilience $R(n)$ to be the average minimum cut-set size within an n-node ball around any node in the topology[8]. We make R a function of n not h—the number of nodes in the ball, not the radius of the ball itself—to factor out the fact that graphs with high expansion will have more nodes in balls of the same radius.

Computing the minimal cut-set size for a balanced bi-partition of a graph is NP-hard [24]. We use the well-tested heuristics described in [24] for our computations of $R(n)$.

A random graph with average degree k has $R(n) \propto kn$ and a mesh has $R(n) \propto \sqrt{n}$. The tree, of course, has $R(n) = 1$. Thus, the tree has qualitatively lower resilience than the other two graphs.

Tree-like behavior: Distortion While it appears somewhat unnatural and unmotivated, our final metric, *distortion*, comes from the graph theory literature [21]. Consider any spanning tree T on a graph G, and compute the average distance on T between any two vertices that share an edge in G. This number measures how T *distorts* edges in G, *i.e.,* it measures how many extra hops are required to go from one side of an edge in G to the other, if we are restricted to using T. We define the distortion of G to be the smallest such average over all possible Ts. Intuitively, distortion measures how tree-like a graph is. This definition is a special case of minimum communication cost spanning trees defined in [21].

For a given graph, distortion is a single number. As we did with resilience, we define the distortion $D(n)$ for a topology to be the average distortion of a subgraph of n nodes within a "ball" around a node in the topology. Computing the distortion can be NP-hard [37]. For the results described in this paper, we use the smallest distortion obtained by applying our own heuristics.[9]

The tree has $R(n) = 1$. The random graph and the mesh each have $R(n) \propto \log n$ [19].

Summary To more fully understand the distinctions made by our three metrics, we consider two other standard networks: a fully-connected network and a linear chain. A fully-connected network has extremely high expansion ($E(h) = 1$) and resilience ($R(n) \propto n$), and low distortion ($D(n) = 2$). A chain (linear) network (with N nodes) has extremely low values on all three: $E(h) = \frac{h}{N}$, $R(n) \propto 1$, and $D(n) = 1$. We don't use these for calibration because they have trivial expansion properties (all nodes within one hop, or one node at each hop) that doesn't work well with our ball-growing metric, but they are useful here.

If we divide behavior for each metric into *high* (H) and *low* (L), we can construct the following table which lists the properties of our five representative networks:

Topology	Expansion	Resilience	Distortion
Mesh	L	H	H
Random	H	H	H
Tree	H	L	L
Complete	H	H	L
Linear	L	L	L

Notice that each of the five networks has its own low/high signa-

[7]An N node bipartite graph from a vertex set A to a vertex set B is said to be an (a, b) expander if, every set of $n < aN$ nodes in A has at least $m > bN$ neighbors in B [34].

[8]For each node in the network, we grow balls with increasing radius. For the subgraph formed by nodes within a ball, we compute the number of nodes n as well as the resilience of the subgraph. We repeat this computation for all (for larger subgraphs, we repeated the computation for sufficiently large number of randomly chosen nodes, in order to keep computation times reasonable) other nodes, then average the sizes and resilience values of all subgraphs of the same radius.

[9]For each node in the network, we grow balls with increasing radius. For the subgraph formed by nodes within a ball, we compute the number of nodes in the ball. We then use an all-pairs shortest path computation on the ball. The node through which the highest number of pairs traverse is deemed to be the "center" of the ball. The subgraph's distortion value is determined by the distortion of the BFS tree rooted at the center. We repeat this computation for all (for larger subgraphs, we repeated the computation for sufficiently large number of randomly chosen nodes, in order to keep computation times reasonable) other nodes, then average the sizes and distortion values of all subgraphs of the same radius. We also use a simple divide and conquer algorithm suggested by Bartal [5]. This approach is known to compute distortions to within $O(\log(n))$ of the optimal solution. We should note that for all the topologies except mesh our own heuristics resulted in smaller distortion values than that obtained using this heuristic.

ture. Thus, this set of metrics is successful at distinguishing between the canonical networks.

We have not been able to find a canonical network with the *LHL* pattern. In fact, the complete graph is the only example we have of any network with high-resilience and low-distortion. The complete graph shows that these two properties (resilience and distortion) are not redundant (*i.e.,* they refer to different aspects of network structure). However, the artificiality of the complete graph, and the lack of simple examples of high-resilience and low-distortion networks might lead us to suspect that networks with high-resilience and low-distortion are unlikely to occur in practice. In fact, we find in Section 4 that the two Internet graphs have these properties.

Also missing are the combinations *LLH* and *HLH*. We conjecture that high distortion implies high resilience so these combinations are impossible.

4. RESULTS

We now describe the results of applying our three basic metrics to specific instances of measured, canonical, and generated networks (Figure 1). Some of the network generators allow a variety of input parameters. For these, we use particular instances of generated networks, whose parameters are described in Figure 1. In Section 4.4 we discuss the sensitivity of our results to parameter variations. Of the generated and canonical networks, only the PLRG qualitatively captures the degree distribution of the measured networks [42].

4.1 Expansion

Figures 2(a,d,g) plot the expansion $E(h)$ for our measured, generated, and canonical networks. Following our discussion in Section 3.2.1, Figure 2(a) shows that Tree and Random expand exponentially (up until the regime where almost all nodes are reached), although at slightly different rates. Mesh exhibits a qualitatively slower expansion. AS and RL also expand exponentially,[10] and their behavior doesn't qualitatively change when policy is considered. Of the generated networks, Transit-Stub (TS), PLRG, and Waxman expand exponentially, but Tiers shows a markedly slower expansion similar to Mesh.

In summary, then, we can categorize our networks into two classes, those that expand exponentially, and those that expand more slowly. Using our low/high terminology of Section 3.2.1, we say that Mesh and Tiers have low expansion, and all other networks exhibit high expansion.

Consistent with our initial assumptions (Section 3.2.1), we have drawn *qualitative* (and therefore somewhat subjective) distinctions. We ignore quantitative differences in metric values, such as different constants or slopes. We also do not use sophisticated curve-fitting techniques to infer the mathematical form of $E(h)$ for some of the measured and generated networks.

4.2 Resilience

Figures 2(b,e,h) plot the resilience function $R(n)$ for our measured, generated, and canonical networks. Of our canonical networks, Tree has the lowest resilience (Figure 2(b)). The minor variations in this function can be attributed to the heuristics we use to determine the cut-set. The resilience of Mesh increases with ball size, but more slowly than Random.

The measured networks exhibit a high resilience that is comparable with that of Random. However, RL and AS differ from each other quantitatively. Also, when policy routing is taken into account, the resilience of the RL and AS graphs decreases (the former by almost a factor of two), although its qualitative behavior as a function of ball size remains unchanged for both graphs. Of the generated networks, Waxman closely resembles Random, and Tiers closely resembles Mesh. TS has low $R(n)$[11], similar to Tree.[12] Finally, PLRG has high resilience, like Random, although it does not match Random as closely as Waxman does.

Following our low/high classification of Section 3.2.1, we then say that TS and Tree have low resilience, and all the other networks have high resilience.

4.3 Distortion

Figures 2(c,f,i) plot $D(n)$ for our measured, generated and canonical networks. The distortion of the Tree is low, whereas that for Mesh and Random are high.

By our reckoning, the measured networks (Figure 2(h)) have low distortion, more so when policy routing is taken into account. Their distortion, although it increases with n, appears qualitatively different from Mesh or Random. The same is true of most of the generated networks, with the sole exception of Waxman.

From this discussion, we conclude that Random, Mesh and Waxman all have high distortion. All other networks have low distortion.

4.4 Discussion

The preceding discussion reveals the following low/high classifications for our measured and generated networks:

Topology	Expansion	Resilience	Distortion	Comment
Mesh	L	H	H	
Random	H	H	H	
Tree	H	L	L	
Complete	H	H	L	
Linear	L	L	L	
AS, RL, PLRG	H	H	L	Like complete graph
Tiers	L	H	L	No counterpart
TS	H	L	L	Like Tree
Waxman	H	H	H	Like Random

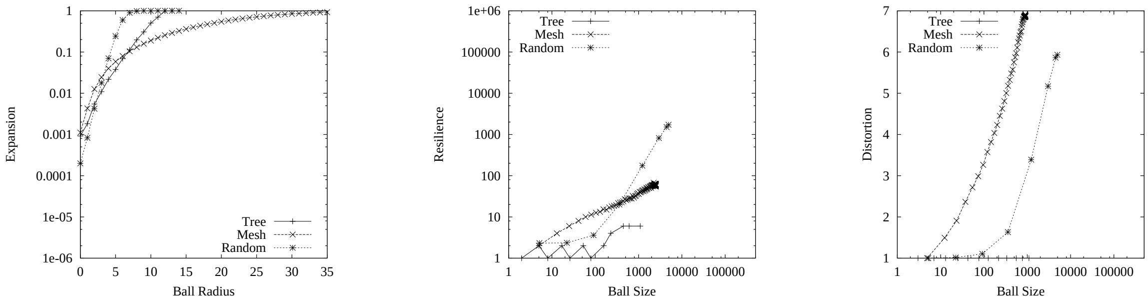
Both measured graphs have rapid expansion, high resilience, and relatively low distortion; that is, these networks can be seen as tree-like, except that they are resilient. Policy routing does not change this classification. Even though there is no *a priori* reason to assume that the AS and RL topologies would be qualitatively similar, our metrics suggest that they are quite similar, at least in terms of the properties measured by our metrics.[13]

[10]The finding that the expansion of the RL graph is exponential is not universally accepted [17]. However, at least two other studies agree with our conclusions [35, 44].

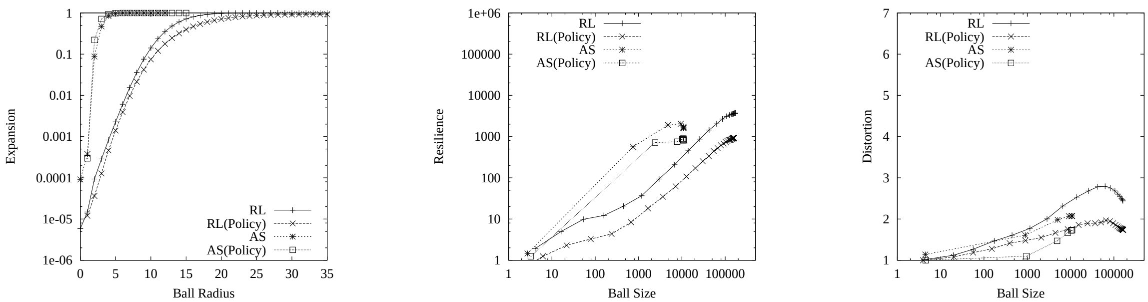
[11]TS has many parameters, one of which is the fraction of redundant transit-to-stub or stub-to-stub links. We tried varying this parameter (from 1% to 60%) in an attempt to increase the resilience of TS. When we do so, however, the distortion of TS increases to match that of the random graph.

[12]Notice that there are minor irregularities in $R(n)$ for TS. We attribute this to the observation that, of two balls of slightly differing size, a larger ball can have a lower resilience. For example, consider this contrived example of two completely connected networks each with n nodes joined by a single link. A ball of radius 1 centered on any node has a resilience of n; a ball of radius 3 centered on any node has a resilience of 1.

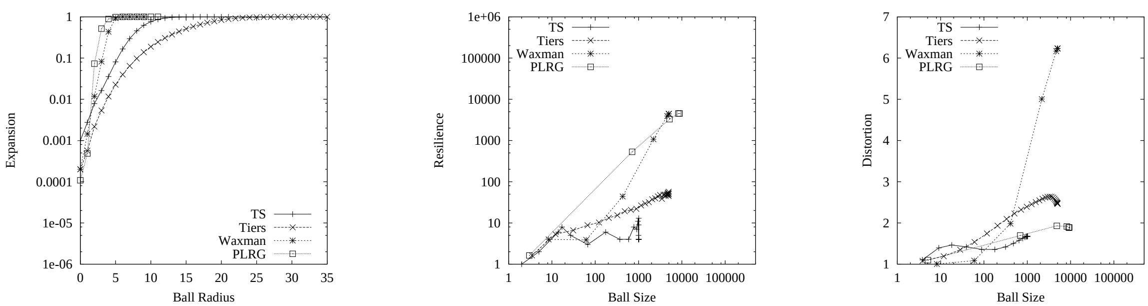
[13]The results presented here contain one instance of each of the AS and RL graphs. In fact, we computed these metrics for at least two other instances, generated more than six months apart from each other (see footnote 3 for dates). Moreover, the RL graph of August 1999 was approximately a factor of two larger than the later graphs (the size difference is due to the difference in the duration of exe-



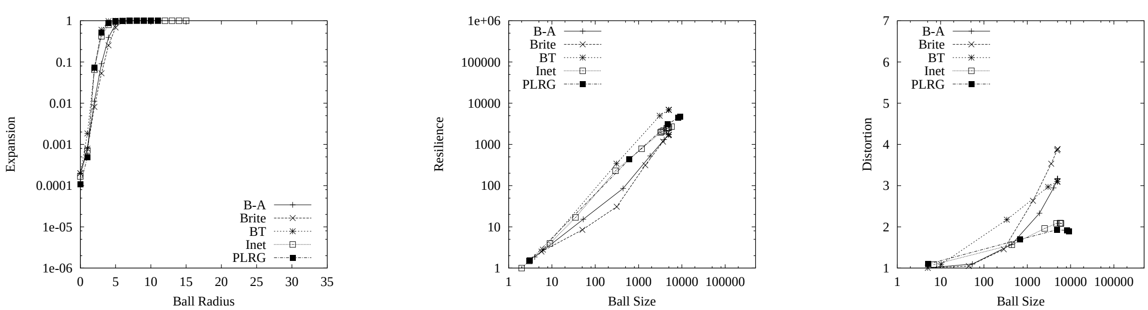
(a) Expansion, Canonical (b) Resilience, Canonical (c) Distortion, Canonical



(d) Expansion, Measured (e) Resilience, Measured (f) Distortion, Measured



(g) Expansion, Generated (h) Resilience, Generated (i) Distortion, Generated



(j) Expansion, Degree-Based Generators (k) Resilience, Degree-Based Generators (l) Distortion, Degree-Based Generators

Figure 2: Our three metrics: Expansion, Resilience and Distortion

Among the standard graphs, only the complete graph has the same low-high signature[14] as these measured graphs. Moreover, two of the generated graphs resemble a canonical network. TS resembles the Tree, and Waxman closely models Random. Tiers does not have a canonical counterpart; it resembles Mesh in two metrics, but has low distortion unlike the Mesh.

When comparing our measured graphs to the generated ones, we find that three of the generated graphs differ from the measured graphs in one particular metric: Tiers has low expansion, TS has low resilience, and Waxman has high distortion. Only the PLRG matches the measured graphs in all three metrics. Thus, we contend that PLRG produces graphs that are better qualitative matches to the Internet graphs than those produced by the other generators.

This conclusion holds for all other degree-based generators we tested. Figure 2(j-l) shows our three metrics for *four* other proposed degree-based generators: Brite version 1.0 [28], BA [4], BT [8] and Inet [23]. All of these can be classified, along with the PLRG, as having high expansion and resilience, and low distortion. These generators all produce graphs with a power-law degree distribution, but differ in the way nodes are connected together. In [42], we investigate other ways of connecting nodes, and find that our conclusions are robust to variations in node connectivity, provided the connectivity method incorporates some notion of random connectivity and the generated graph's degree distribution is qualitatively similar to that of the measured graphs.

These conclusions about generated networks hold for a wide variety of parameters [42]. While for most parameter values the results are in agreement with what we have presented here, it is possible to drive these generators to different operating regimes using extreme choices for parameters. For the Waxman generator, it is possible to introduce extreme geographic bias, thereby dramatically reducing the likelihood of having links between two nodes that are far apart. This also reduces the likelihood of obtaining a connected graph. In this regime, the largest connected component of the Waxman network has low expansion, low resilience and low distortion. It then resembles a minimum spanning tree overlaid on points on a plane, where edge weights are proportional to Euclidean distance. For two-level TS hierarchies with a large transit portion, TS tends toward a random graph. Finally, with Tiers, the average degree parameter can be reduced to the point where it starts to resemble a minimum spanning tree.

In addition to our three basic metrics, we have evaluated five other metrics [42].[15] Some of these were of our own devising, but many were taken from the literature. In all cases the results were consistent with the findings above. In many cases the metrics did not distinguish between different graphs, but whenever there was a clear distinction it was consistent with the grouping found by our three basic metrics. In fact, the three metrics stood out clearly because of their superior ability to distinguish between the various networks. We conclude that, even by these additional metrics, the PLRG resembles the AS and the RL graphs, the Waxman resembles Random, and TS[16] qualitatively matches the tree, modulo the observation that extreme choices of parameters can alter the properties of the generated graphs. The PLRG is the only generator with a power-law distribution of the rank of positive eigenvalues, a signature of the AS topology [17]. (The RL graph was too large to obtain its eigenvalue spectrum.) The diameter distributions have a similar bell-curve shape (with the Tree as the sole exception, as discussed in footnote 16), although with different magnitudes. The error tolerance [3] plots for all the graphs are qualitatively similar, but with different magnitudes. However, the measured networks have a peaked attack tolerance [3], a characteristic shared by PLRG and Tiers. The vertex cover metric of all graphs are quite similar to each other, and the biconnectivity metric of all graphs has a similar behavior with the exception of Mesh, Random, and Waxman.

In addition to these various metrics that are intended to measure large-scale structure, we did compute the clustering metric used in [8] on our various graphs. Using our ball-growing technique and looking at the overall curve's behavior, the PLRG graph had a behavior similar to that of the AS graph, but different from that of all other graphs including the RL. However, when merely looking at the value of the clustering coefficient computed on the whole graph, the PLRG (and the structural generators) exhibited significantly different clustering coefficients compared to either the AS or the RL. We conclude that while PLRG captures the large-scale properties of our measured graphs, it may not capture the local properties of these graphs.

5. HIERARCHY

We are now faced with a paradox. There seems little doubt that the Internet has a significant degree of hierarchy; at the router level network engineers routinely speak of *backbones* and at the AS level ISPs are broken into different "tiers." However, our results in Section 4 indicate that these hierarchical networks—both AS and RL—are better modeled by generators that make no attempt to create hierarchical structure. This section is devoted to resolving this paradox.

Our first task is to better understand what hierarchy is and how it might be measured. The notion of hierarchy revolves around the intuition that there is a set of *backbone* links that carry the traffic from many source-destination pairs; that is, the traffic is not evenly spread out among the links but instead is funneled into more central backbones. We therefore conjecture that a symptom of hierarchical structure is that some links are used more often than others. Here we are not referring to the level of traffic, which is a function of the sending patterns of individual hosts, but rather usage as measured by the set of node pairs (source-destination pairs) whose traffic traverses the link when using shortest path routing; we call this the link's *traversal set*.[17]

The most natural measure of hierarchy would be the size of the traversal set. This simple measure turns out to be misleading; for instance, access links (*i.e.* links with a single node on one end) have a traversal set of size $N-1$ (where N is the number of nodes in the network), which turns out to be a relatively large traversal set. We therefore chose instead to measure the (weighted) vertex cover of the traversal set. The vertex cover of a traversal set is the minimum number of nodes that need to be removed to eliminate at least one node from each pair in the traversal set. For instance, access links have a vertex cover of 1, since eliminating the singleton

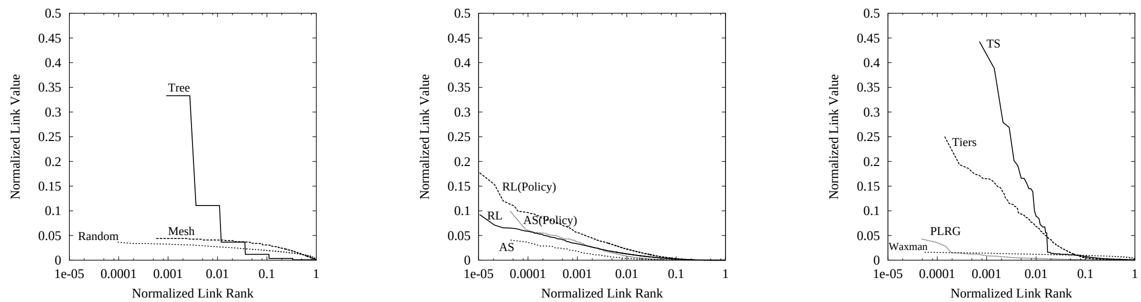
cution of the topology discovery software). Despite the differences in size and time of generation, these other measured graphs did not change our conclusions.

[14] We do not mean to suggest that the AS and RL graphs resemble the complete graph. The latter exhibits an extreme expansion behavior (all nodes are reachable within one hop) that the AS and RL do not.

[15] In addition to the metrics described in [42], we also tested many others (of our own devising), including the average path length between any two nodes in a ball of size n, and the expected max-flow between the center of a ball of size n and any node on the surface of the ball. These metrics, too, do not contradict our findings but do not add to them either.

[16] The diameter distribution for the tree is one-sided, but nevertheless resembles Transit-Stub.

[17] Recall that a "link" in a topology graph might represent various forms of shared media in the underlying Internet.



(a) Canonical (b) Measured (c) Generated

Figure 3: The link value rank distribution (x-axis on log scale)

node eliminates all pairs from the set. Intuitively, the vertex cover counts the smallest set of nodes affected by removal of the link. A link for which this number is high is more important (*i.e.* more nodes depend on this link) than links for which the number is low. We tested this hierarchy metric on several small example networks, and it produced results which coincided with our intuitive notion of the hierarchy in those graphs. To use this metric in the presence of multiple shortest paths, we had to use a weighted vertex cover.[18] We use well-known approximation algorithms [30] for computing weighted vertex covers.

For all topologies, we compute link values using shortest path routing. In addition, for the AS and RL topologies, we use the simple policy model described in Section 3.2.1 to evaluate link values using policy-constrained paths.

We expect that backbone links will have higher values than peripheral links.[19] Thus, the *distribution of these link values* is our measure of hierarchy; if all links have similar values then there is no hierarchy because usage is spread out evenly, and if only a few links have high link values then there is a small and well-defined backbone on which usage is concentrated (where, again, usage is not measured by the level of traffic but by the nature of the traversal set).

5.1 Link Value Distribution

Figures 3(a)-(c) show the link value distributions for the canonical, generated, and measured networks. In these plots, the x-axis plots the rank of a link according to its value (a higher rank indicating a higher value), normalized by the number of links in the topology. The y-axis depicts the link value normalized by the number of nodes in the network. Figures 4(a)-(c) plot the same data but on different scale. By examining these figures, we conclude that there exist three classes of hierarchy in our graphs: strict, moderate, and loose.

Consider Figure 3 first. These plots emphasize the distribution the highest valued links in the network. In terms of the magnitude of link values, the data reveals that the highest link values in Tree, TS, and Tiers are significantly higher than all the other topologies, and their link value distributions fall off rapidly. For the Tree and TS some links have link values above 0.3 but only about 10% have link values above 0.005. The distribution in Tiers falls off equally sharply, even though the highest link value is only 0.25. We say, by this measure, that these topologies have a *strict* hierarchy.

By examining Figure 4, our two other groupings become evident. From this figure, we see that RL[20], AS, and PLRG can be well described as having a *moderate* hierarchy.[21] These graphs have the property that, like the strict hierarchy graphs, the distribution of link values falls off quickly (less than 10% of the nodes have link values greater than 0.005) but the highest value links are significantly lower than those in the strict hierarchy graphs.

In contrast, the Mesh, Random graph and Waxman have a significantly more well spread link value distribution. Even though the highest link values are comparable to that of graphs in the previous category, almost 70% of the links in these graphs have link values about 0.05 and the distribution is very flat. We say that graphs in this category have a loose hierarchy (at best). This is consistent with generally accepted wisdom about the lack of significant hierarchy in the mesh and the random graph.

Finally, note that accounting for policy in computing the link values does not qualitatively alter our groupings. As expected, with policy routing since paths are more concentrated, the highest link values are larger than with shortest path routing, both for AS and RL.

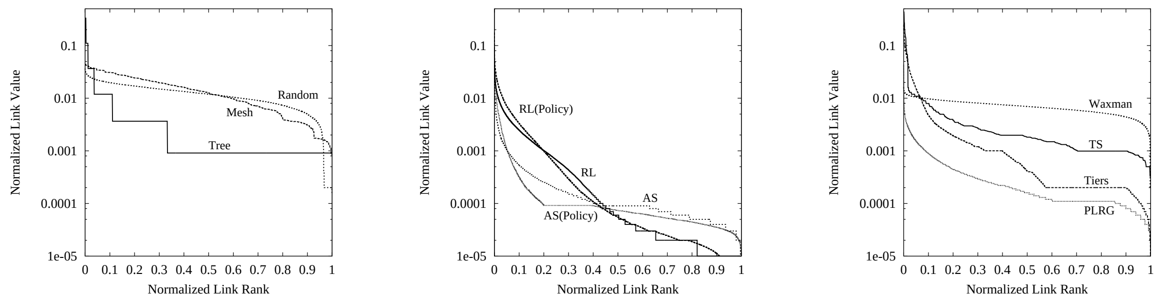
The table below depicts these qualitative groupings.

[18] First, we generalize the definition of the traversal set to include weights associated with node pairs. The weight $w(u, v, l)$ assigned to a node pair (u, v) for a link l is the fraction of the total number of equal cost shortest paths between u and v that traverse link l. Thus, if there are multiple shortest paths between a node pair, the contribution of the node pair is accordingly weighted. Consider now the bipartite graph formed by the traversal set. To each vertex u in this graph, we assign a vertex weight $W(u, l)$ which is simply the average $w(u, v, l)$ such that (u, v) belongs to the traversal set. We define a link's *value* to be the minimum weighted vertex cover in the bipartite graph.

[19] We have actually verified, for several of our topologies, that this expectation holds: the highest valued links in TS are in the transit cloud; in Tiers they are in the WAN; in the AS graph, they connect well-known national backbone, and in the RL graph they occur in, or between, these backbones. This provided a sanity check on our approach to measuring hierarchy.

[20] Computing the link values for the full RL graph is computationally expensive. Therefore, we compute the link values of the *core* topology (generated by recursively removing degree 1 nodes) instead. In previous work, we have found that link values (computed in a slightly different way) computed on the core map correlate well with link values obtained from a full map.

[21] In fact, the other degree-based generators that we evaluated in Section 4.4 also fall into this category (see [42]).



(a) Canonical (b) Measured (c) Generated

Figure 4: The link value rank distribution (x-axis on linear scale)

Topology	Strict	Moderate	Loose
Mesh			x
Random			x
Tree	x		
AS, RL, PLRG		x	
Tiers	x		
TS	x		
Waxman			x

From these groupings we make two important observations.

- The structural generators construct a much stricter form of hierarchy than is present in the measured graphs. This suggests a possible explanation for why they do not qualitatively match the measured networks by our topology metrics (Section 4).
- PLRG qualitatively models the hierarchy present in AS and RL graphs, even with policy routing accounted for. This resolves our paradox to some extent. Although not explicitly hierarchically constructed, PLRG does capture the moderate hierarchy in our measured networks. A question remains: what aspect of PLRG graphs is responsible for this hierarchy? We address this in the next subsection.

5.2 Correlation between link usage and degree

To better understand the hierarchical structure of these graphs, we compute the correlation between a link's value and the lower degree of the nodes at the end of the link. A high correlation between these two indicates that high-value links connect high degree nodes. Figure 5 shows the correlations for the nine networks under consideration.

The PLRG has extremely high correlation. There is absolutely no explicit structure built into this graph. The only links that have (relatively) high values are the ones that connect two nodes with (relatively) high degrees. In the PLRG graph the long-tailed nature of the power-law degree distribution means that there are numerous nodes with very high degrees. One can think of these high-degree nodes as "hubs" and the high value links—the *backbone* links—are those that connect two hubs. In this sense, the hierarchy in a PLRG arises entirely from the long-tailed nature of its degree distribution.

The Random graph also has a relatively high correlation. In this graph, there is absolutely no explicit structure built in. The only links that have (relatively) high values are the ones that connect two nodes with (relatively) high degrees. However, the Random graph has a very limited distribution of degrees, and so the spread of link values is similarly limited, resulting in very limited hierarchy.

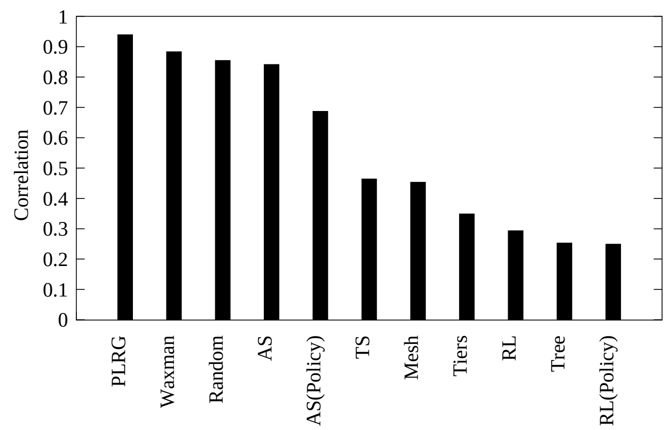


Figure 5: Correlation between minimum degree and link value

In contrast, the Tree has the lowest level of correlation. Unlike the PLRG, the Tree's hierarchy comes from the structure—from the deliberate way in which the nodes are connected—and not from the degree distribution. The correlation that is present is because the leaves have a lower degree than the other nodes, and the associated links have the lowest link values in the tree.

The AS and Waxman graphs have relatively high correlation, while the Mesh, TS, Tiers, and RL have relatively low levels of correlation. This is consistent with our reasoning above, that the hierarchy in the structural generators (Tiers and TS) arises, like the Tree, from the deliberate placement of links. The fact that the AS graph has higher correlation than the RL graph, even though they have very similar levels of hierarchy, may indicate that the hierarchy in the RL graph is due to the deliberate placement of links while in the AS graph the hierarchy is more related to the degrees of the nodes (that is, to the peering relationships between the highly connected ASs that form the "backbone" of the AS graph).

In summary, given the high correlation between link value and degree of the attached nodes, we surmise that the hierarchy in degree-based generators arises from their long-tailed degree distribution. Structural generators show no such correlation, and the hierarchy arises from explicit construction. The RL graph shows less correlation, suggesting that its hierarchy is deliberately constructed, even though its link value characteristics are quite similar to the PLRG.

6. DISCUSSION

We began this paper by questioning the widely accepted belief that degree-based generators, by the very fact that they match the degree distribution of the Internet, are superior to structural generators. We claimed that it is more important that topology generators capture the large-scale structure of the Internet than to reproduce the purely local properties such as the degree distribution. We further argued that, despite the widespread acceptance of degree-based generators, it was still an open question as to which family of generators—structural or degree-based—would better capture these large-scale properties. The goal of this paper was to answer this question.

Our preliminary results suggest that:

- Degree-based generators capture the large-scale structure of the measured networks surprisingly well, at least according to our metrics, and are significantly better than structural generators.
- The hierarchy present in the measured networks is looser and less strict than in the structural generators, and this is well captured by the hierarchical structure in degree-based generators. This may explain why these generators better match our measured topologies in terms of our metrics.
- The hierarchy in degree-based generators arises from the long-tailed distribution of degrees, and the backbone links are merely the links connecting two high-degree nodes. The hierarchy in the RL graph is not highly correlated with degree (and thus is due to the deliberate placement of links) while there is a higher correlation in the AS graph.

These results should not be interpreted as obviating the structural generators. The focus in this paper has been on which family of generators best model the large-scale structure of the Internet, which has restricted our attention to rather large graphs (the smallest generated graph had 1000 nodes). Choosing a small (less than, say, 100 node) topology on which to run network simulations is an entirely separate question. As noted in [50], a power-law distribution is almost meaningless if the number of nodes is small. With only a few nodes, it is unlikely that the degree distribution will be able to create the implicit hierarchy necessary for modeling networks. It may well be that the current structural generators, or ones yet to be devised, are better choices for small-scale simulation studies. Finally, structural generators might be more appropriate for topology models that incorporate bandwidth, topology or geography.

Acknowledgements

The Center for Grid Technologies at ISI allowed us the use of their resources for our computations. Mark Handley and Fabio Silva helped configure additional resources. Finally, Ashish Goel gave us valuable feedback on earlier versions of the paper.

7. REFERENCES

[1] W. Aiello, F. Chung, and L. Lu. A Random Graph Model for Massive Graphs. In *Proc. of the 32nd Annual Symposium on Theory of Computing*, 2000.

[2] R. Albert and A.-L. Barabasi. Topology of Evolving Networks: Local Events and Universality . *Physical Review Letters*, 85:5234–5237, 2000.

[3] R. Albert, H. Jeong, and A.-L. Barabasi. Attack and Error Tolerance of Complex Networks. *Nature*, 406, 2000.

[4] A.-L. Barabasi and R. Albert. Emergence of Scaling in Random Networks. *Science*, 286:509–512, 1999.

[5] Y. Bartal. Probabilistic Approximations of Metric Spaces and its Algorithmic Applications. In *Proc. 37th IEEE Symposium on Foundations of Computer Science*, pages 184–193, October 1996.

[6] B. Bollobás. *Random Graphs*. Academic Press, Inc., Orlando, Florida, 1985.

[7] A. Broido and K. C. Claffy. Internet Topology: Local Properties. In *Proceedings of SPIE ITCom 2001*, Denver, CO, August 2001.

[8] T. Bu and D. Towsley. On Distinguishing Between Internet Power-Law Generators. In *Proc. of IEEE Infocom*, 2002.

[9] H. Burch and B. Cheswick. Mapping the Internet. *IEEE Computer*, 32(4):97–98, April 1999.

[10] K. Calvert, M. Doar, and E. Zegura. Modelling Internet Topology. *IEEE Communications Magazine*, June 1997.

[11] R. C. Chalmers and K. C. Almeroth. Modeling the Branching Characteristics and Efficiency Gains in Global Multicast Trees. In *Proceedings of the IEEE Infocom 2001 (to appear)*, Anchorage, Alaska, USA, April 2001.

[12] H. Chang, R. Govindan, S. Jamin, W. Willinger, and S. Shenker. On Inferring AS-Level Connectivity from BGP Routing Tables. In *Proc. of IEEE Infocom*, 2002.

[13] K. C. Claffy and D. McRobb. Measurement and Visualization of Internet Connectivity and Performance. http://www.caida.org/Tools/Skitter/.

[14] M. Doar. A Better Model for Generating Test Networks. In *Proceeding of IEEE Global Telecommunications Conference (GLOBECOM)*, November 1996.

[15] A. B. Downey. Using pathchar to Estimate Link Characteristics. In *Proceedings of the ACM SIGCOMM*, 1999.

[16] A. Fabrikant, E. Koutsoupias, and C. Papadimitriou. Heuristically Optimized Trade-offs. `http://www.cs.berkeley.edu/ christos/`.

[17] C. Faloutsos, P. Faloutsos, and M. Faloutsos. On Power-Law Relationships of the Internet Topology. In *Proceedings of the ACM SIGCOMM*, Sept. 1999.

[18] L. Gao. Inferring autonomous system relationships in the internet. In *Proc. IEEE Globecom*, San Francisco, CA, 2000.

[19] A. Goel and K. Munagala. Extending Greedy Multicast Routing to Delay Sensitive Applications. Technical report, Stanford Univ. Tech Note STAN-CS-TN-99-89, July 1999. Short abstract appeared in the Symposium on Discrete Algorithms, 2000.

[20] R. Govindan and H. Tangmunarunkit. Heuristics for Internet Map Discovery. In *Proceedings of the IEEE Infocom*, Tel-Aviv, Israel, March 2000.

[21] T. C. Hu. Optimum Communication Spanning Trees. *SIAM Journal of Computing*, 3:188–195, 1974.

[22] R. F. i Cancho and R. V. Sole. Optimization in Complex Networks. Condensed Matter Archives, `http://xxx.lanl.gov/abs/cond-mat`, November 2001.

[23] C. Jin, Q. Chen, and S. Jamin. Inet: Internet Topology Generator. Technical Report CSE-TR-433-00, EECS Department, University of Michigan, 2000.

[24] G. Karypis and V. Kumar. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. *SIAM Journal on Scientific Computing*, 20(1):359–92, 1998.

[25] J. Kleinberg, S. R. Kumar, S. Rajagopalan, P. Raghavan, and A. Tomkins. The Web as a Graph: Measurements, Models

and Methods. In *International Conference on Combinatorics and Computing*, 1999.

[26] K. Lai and M. G. Baker. Measuring Link Bandwidths Using a Deterministic Model of Packet Delay. In *Proceedings of the ACM SIGCOMM*, 2000.

[27] D. Magoni and J.-J. Pansiot. An Analysis of Autonomous System Network Topology. *ACM Computer Communication Review*, 31(3), July 2001.

[28] A. Medina, A. Lakhina, I. Matta, and J. Byers. BRITE: An Approach to Universal Topology Generation. In *Proceedings of MASCOTS 2001*, Cincinnati, OH, August 2001.

[29] A. Medina, I. Matta, and J. Byers. On the Origin of Power-Laws in Internet Topologies. *ACM Computer Communications Review*, 30(2), April 2000.

[30] R. Motwani. *Lecture Notes on Approximation Algorithms - Vol I*. Department of Computer Science, Stanford University.

[31] D. Palmer and G. Steffen. On Power-Laws In Network Topologies. In *Proceedings of IEEE Globecom*, 2000.

[32] J.-J. Pansiot and D. Grad. On routes and multicast trees in the Internet. *ACM SIGCOMM Computer Communication Review*, 28(1):41–50, January 1998.

[33] K. Park. Impact of topology on traceback techniques. Private communication.

[34] D. Peleg and E. Upfal. Constructing disjoint paths on expander graphs. In *STOC: ACM Symposium on Theory of Computing (STOC)*, 1987.

[35] G. Phillips, S. Shenker, and H. Tangmunarunkit. Scaling of Multicast Trees: Comments on the Chuang-Sirbu Scaling Law. In *Proceedings of the ACM SIGCOMM*, Sept. 1999.

[36] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin. On Characterizing Network Topologies and Analyzing Their Impact on Protocol Design. Technical Report 00-731, University of Southern California, Dept. of CS, February 2000.

[37] A. Rényi. On the Enumeration of Trees. In *Combinatorial Structures and Their Applications*, pages 355–360. Gordon and Breach, Science Publishers, June 1969.

[38] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson. The End-to-End Effects of Internet Path Selection. In *Proceedings of ACM SIGCOMM*, Boston, MA, September 1999.

[39] R. Siamwalla, R. Sharma, and S. Keshav. Discovering Internet Topology. Unpublished manuscript.

[40] L. Subramanian, S. Agarwal, J. Rexford, and R. Katz. Characterizing the Internet Hierarchy from Multiple Vantage Points. In *Proc. of IEEE Infocom*, 2002.

[41] H. Tangmunarunkit, J. Doyle, R. Govindan, S. Jamin, W. Willinger, and S. Shenker. Does AS Size Determine AS Degree? *ACM Computer Communication Review*, October 2001.

[42] H. Tangmunarunkit, R. Govindan, S. Jamin, and S. S. W. Willinger. Network Topology Generators: Degree-Based vs. Structural. Technical report, Computer Science Department, University of Southern California, 2002. Technical Report 02-760.

[43] H. Tangmunarunkit, R. Govindan, S. Shenker, and D. Estrin. The Impact of Policy on Internet Paths. In *To appear, Proc. of IEEE INFOCOM*, Anchorage, AK, 2001.

[44] R. van der Hofstad, G. Hooghiemstra, and P. van Mieghem. On the Efficiency of Multicast. Submitted for publication.

[45] P. van Mieghem, G. Hooghiemstra, and R. van der Hofstad. A scaling law for the hopcount. Technical report, Delft University of Technology, 2000.

[46] D. Vukadinovic, P. Huang, and T. Erlebach. A Spectral Analysis of the Internet Topology. Technical report, ETH Zurich, 2001.

[47] D. J. Watts and S. H. Strogatz. Collective Dynamics of Small-World Networks. *Nature*, 363:202–204, 1998.

[48] B. M. Waxman. Routing of Multipoint Connections. *IEEE Journal of Selected Areas in Communication*, 6(9):1617–1622, December 1988.

[49] T. Wong and R. Katz. An Analysis of Multicast Forwarding State Scalability. In *Proceedings of the 8th IEEE International Conference on Network Protocols (ICNP 2000)*, Osaka, Japan, November 2000.

[50] E. Zegura. Thoughts on Router-level Topology Modeling. The End-to-end interest mailing list.

[51] E. Zegura, K. L. Calvert, and M. J. Donahoo. A Quantitative Comparison of Graph-Based Models for Internet Topology. *IEEE/ACM Transactions in Networking*, 5(6), 1997.