# Network Topologies, Power Laws, and Hierarchy

Hongsuda Tangmunarunkit
USC-ISI

Ramesh Govindan
USC-ISI

Sugih Jamin
Univ. of Michigan

Scott Shenker
ACIRI

Walter Willinger
AT&T

**Abstract**

It has long been thought that the Internet, and its constituent networks, are *hierarchical* in nature. Consequently, the network topology generators most widely used by the Internet research community, GT-ITM [7] and Tiers [11], create networks with a deliberately hierarchical structure. However, recent work by Faloutsos *et al.* [13] revealed that the Internet's degree distribution — the distribution of the number of connections routers or Autonomous Systems (ASs) have — is a power-law. The degree distributions produced by the GT-ITM and Tiers generators are not power-laws.

To rectify this problem, several new network generators have recently been proposed that produce more realistic degree distributions; these new generators do not attempt to create a hierarchical structure but instead focus solely on the degree distribution. There are thus two families of network generators, *structural* generators that treat hierarchy as fundamental and *degree-based* generators that treat the degree distribution as fundamental.

In this paper we use several topology metrics to compare the networks produced by these two families of generators to current measurements of the Internet graph. We find that the degree-based generators produce better models, at least according to our topology metrics, of both the AS-level and router-level Internet graphs. We then seek to resolve the seeming paradox that while the Internet certainly has hierarchy, it appears that the Internet graphs are better modeled by generators that do not explicitly construct hierarchies. We conclude our paper with a brief study of other network structures, such as the pointer structure in the web and the set of airline routes, some of which turn out to have metric properties similar to that of the Internet.

## 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 [36]. This generator is a variant of the classical Erdos-Renyi random graph [5]; its link creation probabilities are biased by Euclidean distance between the link endpoints. A later line of research noted that real network topologies have a much more complicated structure than random networks and emphasized the fundamental role of *hierarchy*. The following from [39] reflects this observation:

> ...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, GT-ITM [7] and Tiers [11], 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.

[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.

A recent paper by Faloutsos *et al.* [13] cast doubt on the extent to which these structural generators accurately reflect the current Internet. They used recent 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 structural generators do not produce power-law degree distributions. In order to more accurately model the degree-distribution data, several generators have recently been proposed [18, 22, 2, 24, 1] that achieve power-law degree distributions. We will call these *degree-based* generators because of their focus on the degree distribution.

The Internet research community now has two families of generators; one family built around the degree distribution as the fundamentally important consideration, and the other family built around structure and hierarchy as the fundamentally important consideration. The guiding principles of the two families of generators are very different; the degree distribution is a very local property, having to do with the numbers of connections at individual nodes, while hierarchy is very nonlocal and describes the overall structure of the network. In spite of the recent surge in papers on network topology generators, there has been very little evaluation of which family of generators produce more realistic networks. In this paper, we attempt such a comparison. Underlying our investigation is the issue of whether the hierarchical structure or the degree distribution is more fundamental in modeling the Internet.

More specifically, after reviewing related work in Section 2, this paper proceeds to ask three questions.

**Question #1** *Which generated networks most closely resemble 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 [16] 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 20 times more nodes and links than the AS-level graph. While they both reflect the same overall structure 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 or about policy routing.[3]

To measure the properties of the Internet graphs and the generated graphs, we use a set of topology metrics described in Section 3.2. These metrics, derived in part from those in [28], are intended to capture some basic *large-scale* or *overall* properties of network structures, as opposed to purely local properties; for instance, they measure how resilient the network is to failed links rather than, say, the correlation between the degrees of nodes at two ends of a link. Thus, we are making a judgement – based more on intuition than on fact – that the large-scale properties are more important, or more fundamental, than these local properties.

[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.

[3] The latter is particularly important when asking questions about paths from one point on the graph to another; we use shortest path routing on these graphs, and the true policy-driven routes might be significantly different [32, 30]. At least one of the generators that we study [7] incorporates simple policy models. We do not include these in our analyses.

The degree distribution is a rather local property. Even though the structural generators do not match the degree distribution of the real Internet topologies, they could nonetheless create networks with large-scale properties that were very similar to the Internet graphs. In fact, this was our expectation.

Our set of metrics — three basic ones plus five additional ones — are an initial attempt to characterize network topologies. While we are not aware of extensive prior work in this area, and have borrowed liberally from the work that exists, we recognize that our metrics do not yet adequately characterize network topologies and that additional work is warranted in this area. Moreover, the results 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 in Appendix B, 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 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 Internet graphs are better modeled by network generators that completely ignore hierarchy! Resolving this paradox leads us to our second question.

**Question #2** *Is there any relationship between hierarchical structure and power-law degree distributions?* In Section 5 we introduce some measures of hierarchy, and then use these 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 the 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.

Some, but not all, of the degree-based generators are quite general in nature; once their degree distribution has been set, the nodes are connected essentially at random.[4] Such generators do not embody any Internet-specific heuristics, and thus might be reasonable models of a larger class of networks. To investigate this issue, we asked a third question.

**Question #3** *Do our findings apply to other "real" network structures?* In Section 6, we apply our metrics to several other network structures, such as the web and the set of airline routes. We find that several of these other networks do indeed resemble the Internet, at least in a very rough way, and are well-modeled by the same degree-based generators. Thus, when searching for explanations for why the Internet has its current structure, and why the degree-based generators are such good models of the Internet, we would be well-advised to look for explanations that have fairly general applicability and are not restricted to the particular details of the Internet.

## 2 Related Work

We have already mentioned several important areas of related work: the Waxman, GT-ITM 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 [18, 22, 2, 24, 1]. They all attempt to generate networks

[4]Some of the degree-based generators do not precompute a degree distribution but instead generate one through a growth process (*e.g.,* [22]) and others do not connect the nodes randomly (*e.g.,* [18]).

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 Section 4.5.

Perhaps closest in spirit to the work presented in this paper is the pioneering exploration of topology properties by Zegura *et al.* [39]. 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.* [34] 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.* [23]. 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 [22]). Their metrics for comparison include the tests in [13] 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 [23] or elsewhere in the literature because networks with similar degree distributions can have very different large-scale properties. 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. Our focus in this paper is on large-scale structure, not on the purely local issues like the degree distribution, and thus we choose different metrics for our evaluations.

Our work would not have been possible without the recent 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 [25]. Subsequent work improved the coverage of the Internet address space by randomly selecting IP addresses [31], randomly selecting addresses from route entries in BGP tables [6], using a precomputed set of Web sites [9], or using heuristics to infer addressable parts of the IP space [16]. This last work also documents several techniques for improving completeness of the inferred topologies.

Several recent papers have addressed the impact of topology on protocol performance. For example, Phillips *et al.* [27] 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 [8] considered a variety of metrics for the efficiency of multicast trees. Wong and Katz [37] found that amount of multicast state from randomly placed receivers differs qualitatively with different topologies. Radoslavov *et al.* [28] 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. One notable example [14], describes a viable technique for inferring hierarchical relationships (*e.g.*, provider-customer) in the AS topology. This work assumes, as we do, that one signature of a hierarchy is that most paths go up the hierarchy, and then down.

There has been a recent explosion 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 [35] 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.* [20] 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.[5] 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 use in this paper was obtained from the routing table at a router[6] that peers with more than 20 other backbone routers.

Our second measured topology is the Internet router-level (*RL*) topology. This is derived by inferring router adjacencies [16] 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 thousands) 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 may be 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. However, we are not so sanguine about two other problems with our measured networks, both due to the fact that these measured networks merely represent connectivity between nodes and links. First, the measured networks (in particular the RL network) contains no indication of the capacity of the underlying transmission link (or shared medium). Although techniques for estimating link capacities along a path are known ([12, 21]), 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. Second, there is no information about routing policy associated with the RL topology. In this paper we use shortest path routing, but some of our results might be affected by more realistic models of routing. We are currently trying to include policy routing in some of our analysis, but have made little progress so far.

[5] In some previous studies [27] the MBone was included. Currently, the MBone (the DVMRP virtual multicast overlay) is in such rapid decline (a few hundred nodes now as opposed to more than 4000 nodes in 1998) as to not constitute a representative network topology.

[6] route-views.oregon-ix.net

### 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* [36] generator. The classical Erdos-Renyi random graph model [5] 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* [7] and *Tiers* [11] 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, Appendix C 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 $\beta$, it first assigns degrees to $N$ nodes drawn from a power-law distribution with exponent $\beta$ (*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 $i$. Links are then assigned by randomly picking two node copies and assigning a link between them, until no more copies remain.[7] For most of the rest of the paper, we focus almost exclusively on PLRG as the sole degree-based generator. However, in Section 4.5 we discuss several other degree-based generators.

### 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 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. For instance, some might argue that it is important to match the details of the

[7] This generator is not guaranteed to give a connected graph although, for reasonable values of $\beta$, 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.

Internet, such as the total number of nodes and links. It would be hard to definitively refute this position without hard scientific evidence. Thus, we recognize that the metrics we chose are in no way *definitive*, but merely reflect our own judgement.

Our choice of metrics were governed by three assumptions. First, we assume that measures of large-scale structure of the Internet are more relevant than purely local quantities. An example of a metric for large-scale structure is the resilience of the graph to link failure. A local property is the correlation between the degrees of nodes at the ends of a link. Second, we assume that the metrics 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. Lastly, we assume that the relevant properties should distinguish well-known graphs, like our canonical graphs, from each other. That is, we intuitively know that these canonical graphs – mesh, tree, and random graph – are quite different from each other in ways that would be very important to networks, and at the very least our metrics should reveal the distinctions between them quite clearly.

We now use this last assumption to guide the development of three basic metrics, which we describe in turn.

### 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 in $h$ hops (the reachable set). We calculate the size of the reachable set for each node in the graph, average the result, and then normalize the total number of nodes in the graph.

This definition is similar[8] to the reachability function described in [27] and to the hop-pair distribution defined in [13]. In fact, [27] has analyzed the expansion of some, but not all, of the topologies described in Section 3.1. We repeat those analyses here for completeness.

The use of the *ball-growing* technique, in which we measure some quantity in balls of radius $h$ and then consider how that quantity grows as a function of $h$, allows us to compare graphs of different sizes (because, for each $h$, we are measuring the same sized balls in both networks). However, 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 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 $h$.

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.

The expansion metric is defined in terms of the shortest path on the graph. That is, a node $v$ is said to be in the ball of radius $h$ around $v_c$ if the shortest path from $v_c$ to $v$ is no greater than $h$. This definition does not, but could (especially in the context of RL), include policy constraints. For example, we could (as we intend to in future work) amend the definition to count only those nodes $v$ within the ball whose policy paths from $v_c$ to $v$ are no greater than $h$. Other work has shown that policy can have significant impact on Internet paths [32, 30].

**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*

[8] Unlike [27], $E(h)$ 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

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.[9] 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[10]. 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 [19]. We use the well-tested heuristics described in [19] 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 [17]. 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[11] of $G$ to be the smallest such average over all possible $T$s. Intuitively, distortion measures how tree-like a graph is.

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 [29]. For the results described in this paper, we use the smallest distortion obtained by applying several of our own heuristics to sample spanning trees. All of these heuristics use the total number of all-pairs shortest paths traversing a link to inform the selection of links forming the spanning tree.[12]

The tree has $R(n) = 1$. The random graph and the mesh each have $R(n) \propto \log n$ [15].

**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 |

[9] 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.

[10] 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.

[11] This definition is a special case of minimum communication cost spanning trees defined in [17].

[12] 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 our heuristics to sample spanning trees. The subgraph's distortion value is determined by the spanning tree with the least distortion. 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.

Notice that each of the five networks has its own low/high signature. Thus, this set of metrics is successful at distinguishing between the well-known networks, which was one of our guiding principles in our search for metrics.

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.

### 3.2.2 Additional Metrics

We find in Section 4 that our three basic metrics are effective in distinguishing between the various generated and measured networks we consider. However, as we have argued above, the choice of metrics is inherently arbitrary and a matter of judgement. To make sure our results were not swayed by a prejudicial choice of metrics, we augmented our study with several other metrics from the literature (which, where appropriate, we adapted to use our ball-growing technique, or to consider distributions instead of single-valued metrics). We briefly discuss the results of these metrics in Section 4.4 and show detailed graphs in Appendix B.

The following is a list of metrics we have experimented with:

- The distribution of node diameter[13] [39].
- The distribution of eigenvalues [13].
- The average vertex cover of the subgraph within a ball of size $n$ [26].
- The average biconnectivity (number of biconnected components) in a ball of size $n$ [39].
- 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].

# 4 Results

| Type | Topology | Number of Nodes | Avg. Degree | Comment |
|---|---|---|---|---|
| Measured | RL | 228263 | 2.81 | August 1999 |
| | AS | 9107 | 4.02 | December 2000 |
| 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 |

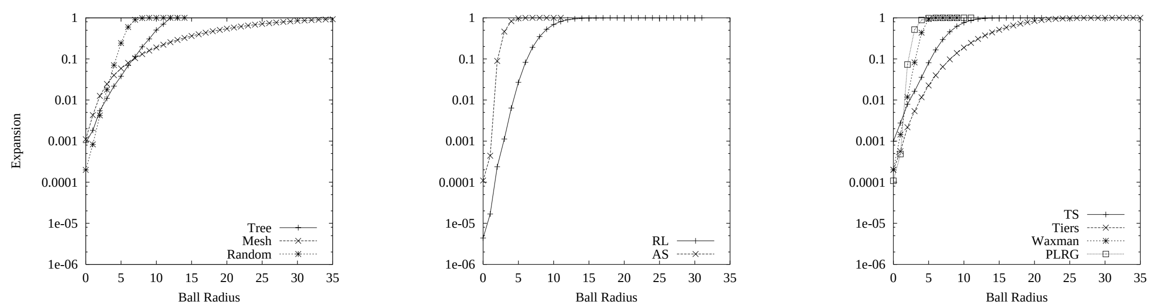
Figure 1: Table of network topologies used. See Appendix C for a description of parameters for the generated networks.

[13]*Node diameter* is synonymous with eccentricity

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.

We present the degree distributions for our real, measured and generated networks in Appendix A. Of the generated and canonical networks, only the PLRG qualitatively captures the degree distribution of the measured networks.

## 4.1 Expansion




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

Figure 2: Expansion

Figure 2 plots 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.[14] Of the generated networks, Transit-Stub (TS), PLRG, and Waxman expand exponentially, but Tiers shows a markedly slower expansion similar to Mesh.

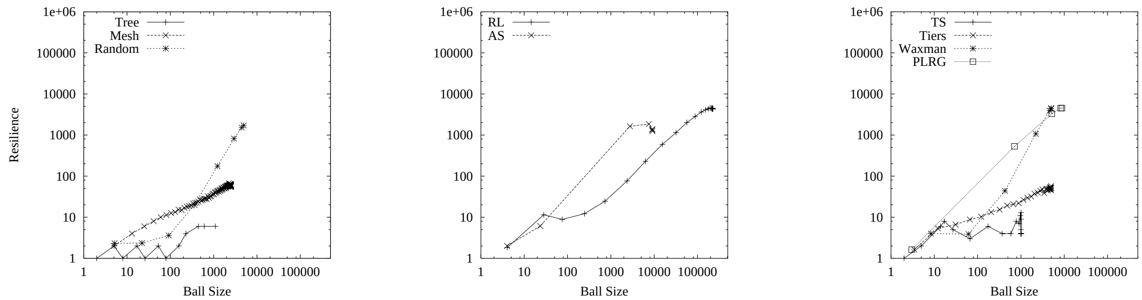
In summary, then, we can categorize our networks into two classes, those that expand exponentially, and those that expand more slowly. Using our low/high terminology of Section 3.2.1, we say that Mesh and Tiers have low expansion, and all other networks exhibit high expansion.

We emphasize that, in drawing these distinctions, we have made *qualitative* (and therefore somewhat subjective) comparisons. 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. Our emphasis on qualitative comparison is consistent with our initial assumption (see Section 3.2.1) that the goal of topology generators is not to produce exact replicas of the Internet, but to produce graphs that have similar large-scale properties. It is also consistent with the unquantifiable incompleteness of our Internet graphs.

## 4.2 Resilience

Figure 3 plots the resilience function $R(n)$ for our measured, generated, and canonical networks. Of our canonical networks, Tree has the lowest resilience (Figure 3(a)). The minor variations in this function can be attributed to

[14]The finding that the expansion of the RL graph is exponential is not universally accepted [13]. However, at least two other studies agree with our conclusions [27, 33].




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

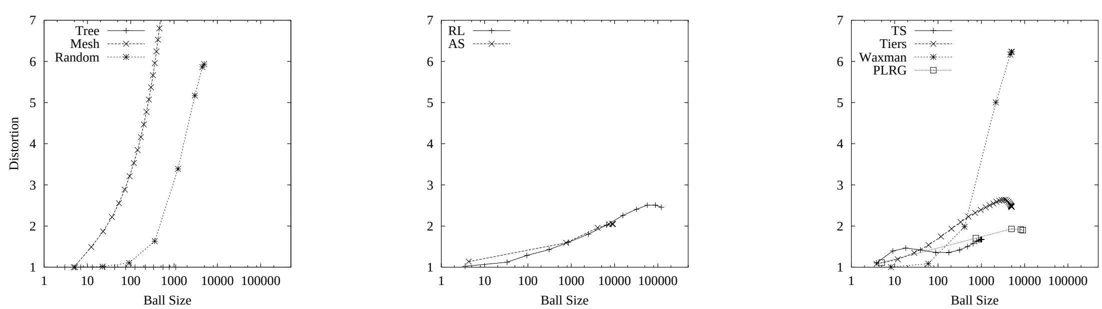
Figure 3: Resilience

the heuristics we use to determine the cut-set. The resilience of Mesh increases with ball size, but perhaps 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. Of the generated networks, Waxman closely resembles Random, and Tiers closely resembles Mesh. TS has low $R(n)$, similar to Tree.[15] 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




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

Figure 4: Distortion

Figure 4 plots $D(n)$ for our measured, generated and canonical networks. The distortion of the Tree is low, whereas that for Mesh and Random are high. The $D(n)$ for Mesh and Random in Figure 4(a) deviate from their known $\log n$ form. We attribute this to our heuristics for sampling low-distortion spanning trees.

[15] 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.

By our reckoning, the measured networks (Figure 4(b)) have low distortion. 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. As we remarked in Section 1, even though there is some relationship between the AS and RL graphs (for example, a link between two ASs in the AS-level topology implies the existence of at least one link between routers belonging to these ASs in RL), these two graphs reflect network structure at very different scales. There is no *a priori* reason to assume that the AS and RL topologies would be qualitatively similar. But our metrics suggest that they are quite similar, at least in terms of the properties measured by our metrics[16].

Among the standard graphs, only the complete graph has the same low-high signature[17] 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. In Section 4.5 we investigate other degree-based generators to see whether this conclusion is robust to variations in node connectivity.

These conclusions about generated networks hold for a wide variety of parameters. We list the various parameter settings that we have explored, for each of these generators in Appendix C. We do not include the corresponding plots, for lack of space. 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

[16]The results presented here contain one instance each of the AS and RL graphs. In fact, we computed these metrics for at least one other instance, generated more than six months earlier, of both these graphs. In particular, the previous RL graph was approximately a factor of two smaller than the latter graph (the size difference is due to the difference in the duration of execution of the topology discovery software). Despite the differences in size and time of generation, these other measured graphs did not change our conclusions.

[17]We should hasten to add, of course, that 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.

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.

We have said that the choice of the three metrics is inherently arbitrary. We have tried[18] roughly 8 other metrics. 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, our choice of our basic three metrics (as opposed to the others we tried) was partially dictated by 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[19] qualitatively matches the tree.[20] Looking at the graphs in Appendix B in more detail, the PLRG is the only generator with a power-law distribution of the rank of positive eigenvalues, a signature of the AS topology [13].[21] The diameter distributions have a similar bell-curve shape (with the Tree as the sole exception, as discussed in footnote 19), 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.

### 4.5 Other Power-Law Network Generators: Does Connectivity Matter?

Until now, we have used a single degree-based generator, the PLRG. The PLRG generator uses a particularly simple technique for connecting nodes (Section 3.1.2): it clones each node as many times as the degree assigned to it, then uniformly randomly connects the clones. However, given a set of nodes with a particular degree distribution (such as a power-law distribution), nodes can be connected in different ways to satisfy the degree requirements.

One class of approaches to node connectivity is exemplified by the Brite [22] generator. This generator incorporates an evolutionary model, described in [4], for power-law degree distribution graphs. In this model, the graph is grown incrementally, with newly appearing nodes randomly connecting to already existing nodes, but in proportion to their degrees. Brite incorporates additional features, such as geographic bias in establishing links; we did not explore these features in our study. A slight variant, which we call the B-A model [2], incorporates link addition and re-wiring; with a small, but uniform probability a link can be added between two nodes, or an existing link can reattach from one endpoint to another based on preferential connectivity. Both models generate networks with power-law degree distributions.

Another class of approaches initially assigns node degrees from a power-law degree distribution, similar to the PLRG. Unlike the PLRG, however, these approaches connect nodes using different rules. For example, after conducting a feasibility test on the generated degree distribution to see if the resulting graph would be connected, the Inet [18] generator creates a spanning tree among nodes of degree larger than one, connects degree one nodes to this spanning tree with proportional connectivity,[22] then satisfies the degrees of remaining nodes in decreasing degree order. Another generator [24] connects the nodes randomly, without cloning.

Other variants of these *random connectivity* techniques for power-law degree distributions exist. Examples

[18]In addition to the metrics described in Appendix B, we also defined some of our own. Examples include 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.

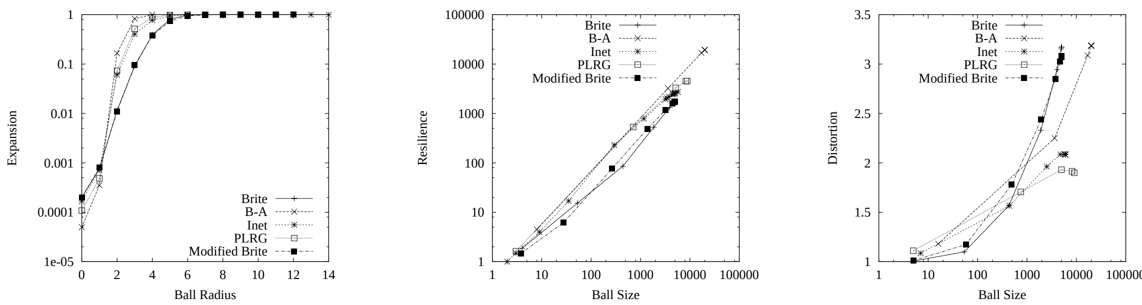
[19] The diameter distribution for the tree is one-sided, but nevertheless resembles Transit-Stub.

[20]Modulo the observation that extreme choices of parameters can alter the properties of the generated graphs.

[21]The RL graph was too large to obtain its eigenvalue spectrum.

[22]The likelihood of attaching to a node is proportional to its degree

include: start with the highest degree (or lowest degree) nodes and connect to other nodes either uniformly, or in proportion to the degree, or in proportion to the “unsatisfied” – assigned degree minus the number of links already assigned to the node – degree.




(a) Expansion

(b) Resilience

(c) Distortion

Figure 5: PLRG Variants

How do these random connectivity variants compare? We have computed our three metrics for all the connectivity variants described above, and some more. Figure 5 plots our three metrics for some of these variants. We conclude that they are all qualitatively similar with respect to our metrics. The Brite and B-A generators have a slightly different distortion curve. In examining their degree distributions we noticed that, in the Brite generator, the largest degree is often significantly less than that in other variants. Furthermore, the B-A generator has fewer low-degree nodes. To test whether their connectivity methods are responsible for the difference, we reconnected the Brite and B-A graphs using the PLRG connectivity method. To do this, we created a new graph each of whose nodes had the same degrees as the corresponding nodes in the Brite or B-A graph, but whose links were created using the PLRG connectivity algorithm described in Section 3.1.2. In Figure 5, we show the result for a Brite graph reconnected using the PLRG connectivity method (we call this the *modified Brite* graph). We find that this network resembles the original Brite network with respect to the distortion metric. The same conclusion holds for a B-A network reconnected using the PLRG connectivity method (not shown).

From these experiments, we conclude that what seems to determine the qualitative behavior of these degree-based generators is the degree distribution, *not* the connectivity method. In particular, slight variations in degree distribution (such as having too few low degree nodes, or not having high enough large degree nodes) result in significant metric differences. In constrast, we found (in experiments we do not have space to report on here) that the metric properties are essentially the same for all of the random connectivity methods we explored. Even for the uniformly random connectivity method, where nodes are not necessarily connected in proportion to their degrees, the large-scale metrics are qualitatively similar to the PLRG.

In addition to these *random connectivity* variants, there exist *deterministic connectivity* variants. One such variant is as follows. Start with the highest degree node, add one link each from this node to each lower degree node in decreasing degree order (skipping nodes whose degree has already been satisfied), then repeat for the next highest degree node whose degree has not been satisfied. We have computed our three basic metrics for these variants of power-law degree-distribution graphs. Lack of space prevents us from including these results but, not surprisingly, deterministic connectivity results in graphs that are quite different from the PLRG (and thus different from the AS and RL graphs).

In summary, then, degree-based generators seem qualitatively similar (in the sense of Section 4.4) to the RL and AS topologies regardless of connectivity method, so long as that method incorporates some notion of random connectivity and the generated graph’s degree distribution is qualitatively similar to that of the measured graphs.

# 5 Hierarchy

We are now faced with a paradox. There seems little doubt that the Internet has a significant degree of hierarchy; network engineers routinely speak of *backbones*, and even 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 there are two symptoms of hierarchical structure.

The first 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*.[23] If there are multiple shortest paths between a node pair, that node pair appears in the traversal set of each link in every shortest path. While it would seem natural to merely measure the size of the traversal set, we choose instead the following more complicated measure. We define a link's *value* to be the fewest number of nodes that "cover" the traversal set. By "cover" we mean the minimal number of nodes that need to be removed so that all pairs in the traversal set have at least one node removed. We realize that this definition appears unnecessarily complicated, but experience with the simpler variants on toy topologies convinced us that they could be seriously misleading. Our definition of "cover" corresponds to a vertex cover on a bipartite graph formed by nodes in the traversal set. We use well-known approximation algorithms [10] for computing vertex covers.

We expect that backbone links will have higher values than peripheral links. The distribution of these link values is our first 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.

We note that the link value measure would be strongly affected by policy routing. Because such information is missing from the topology, we were not able to account for policy in our current treatment. In addition, the capacity of a link (also missing from our topology) might be another useful indicator of its level in the hierarchy.

The second symptom of hierarchy is that paths tend to first go up, and then down, the levels of the hierarchy. That is, a path between two nodes at the edge of the network works its way up the hierarchy until it reaches the backbone, and then works its way back down. Therefore, our second measure of hierarchy looks at the series of link values along a path and asks what fraction of paths have an "up-down" pattern.

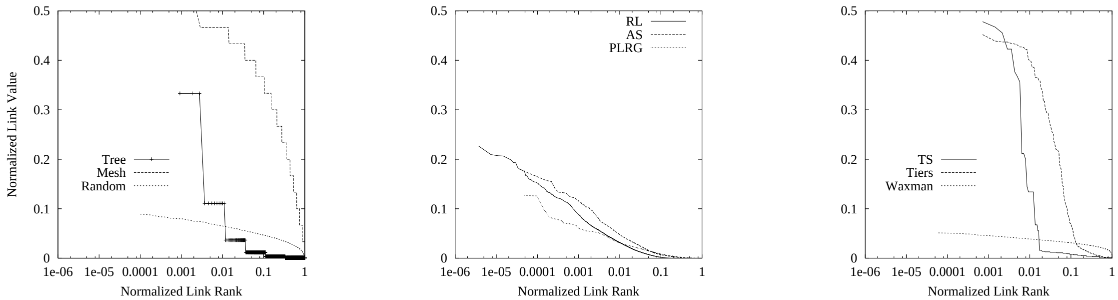
In the next two sections we apply these two measures of hierarchy to our measured, generated, and canonical networks.

## 5.1 Link Value Distribution

The link value distribution reveals the extent to which usage is concentrated on a small set of backbone links. Figures 6(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.

The data reveals four rough groupings. The distributions of the Tree, TS, and Tiers all fall extremely rapidly. For the Tree and TS some links have link values above $0.25$ but only about 1% have link values above $0.05$; this indicates a very strict hierarchy. The distribution in Tiers falls off somewhat less sharply, with 10% of the links

[23] 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 6: The link value rank distribution

having values above $0.05$. To sanity check our definition of link value, we have for Tiers (respectively TS), verified that the high values links occur only in the WAN (respectively transit).

Random and Waxman have very flat distributions, with almost all links having values below $0.1$. This indicates the complete lack of hierarchical structure. The third group consists of the RL[24] and AS graphs, together with the PLRG. Their link value distributions decrease at a roughly logarithmic rate. Again, as a sanity check, we examined the AS graph links with the ten highest link values; we found that they all connect large national or international backbones. While there are significantly fewer links in RL and AS with high value than with low, the drop-off in the distribution is much more gentle than that of the strictly hierarchical graphs (TS, Tiers, and Tree).

The Mesh is in its own category. Unlike Tree, its link value distribution falls off gradually.

As we might have suspected, the link value distributions suggest that the Tree, TS, and Tiers have strict hierarchies, Random and Waxman have no hierarchies. What is somewhat more surprising is that the results indicate that the RL, AS and PLRG have very similar hierarchical structures; looser than that of the Tree but significantly more hierarchical than the Random graph.

What is slightly troubling about these results is that the Mesh, hardly thought of as a hierarchical graph, seems to have a fair degree of hierarchy (more than the Internet graphs!). The high-value links are those in the center of the Mesh, because paths tend to travel through the center of the network.

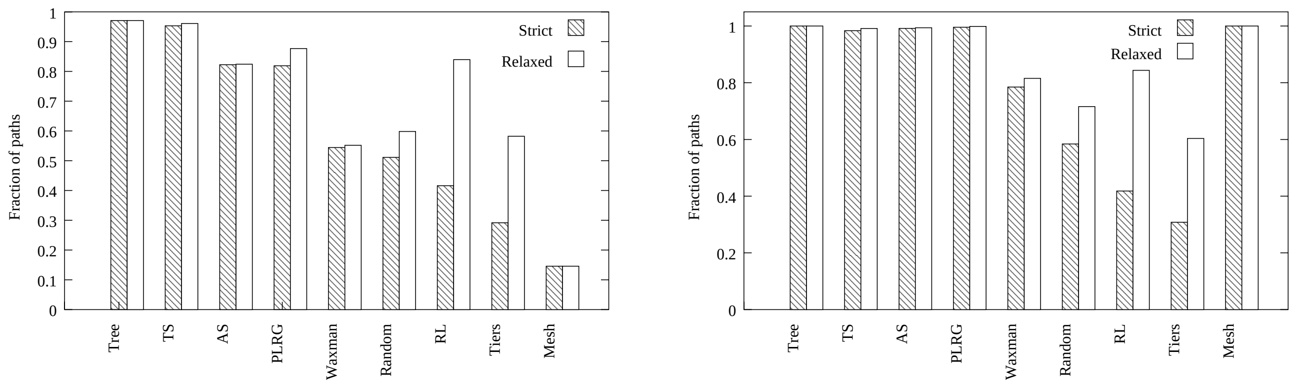
### 5.2 Up/Down Analysis

In hierchical graphs, we expect that most paths travel an “up-down” path, in that the values of the links along the path first increases (as the path climbs the hierarchy) and then decreases (as the path goes down the hierarchy). However, some paths may only go down, and others only up, and still others may be flat. We call each of these *valid* paths in a hierarchical structure because none of these path types counteracts our notions of hierarchy. By contrast, invalid paths are those with a local minimum link value somewhere in the interior of the path.

Our preceding definition of an up-down path is *strict* in the sense that it labels as invalid a path even when the link values have only small local minima (*e.g.,* the dips are small). Because the link values are fine-grain, our strict definition might mask the existence of hierarchies in some networks. For this reason, we also use a

[24]For the RL graph, computing the link values for the nearly 300,000 links is computationally expensive. For about 1400 links for which the traversal set was large, we were unable to compute link values even given fairly generous computing resources. For these links, we approximated their link values from their normalized link values in the corresponding *core* topology (the core topology is generated from the original RL topology by recursively removing degree 1 nodes). We think this is a good approximation because, of the links for which we could compute link values in the RL topology, we found excellent (over 97%) correlation between those values and the corresponding values from the core map.

*relaxed* definition for up-down and valid paths in which we count a local minimum being present only if the value is less than the value of the link preceding the nearby local maximum. It may help to illustrate with examples. The sequence of path values $1, 2, 3, 2.9, 3, 2, 1$ would be rendered invalid by our strict definition, but would be considered an up-down path by our relaxed definition. The sequence of path values $1, 2, 3, 1, 3, 2, 1$ would be considered invalid by both definitions. Figures 7(a)-(b) show the fraction of all paths that fall into the up/down and the valid categories by our two definitions.




(a) Fraction of up/down paths



(b) Fraction of valid paths

Figure 7: Link value up/down analysis and correlation

Over 95% of the paths in the Tree and TS networks are up-down, and fewer than 60% of the paths in the Random and Waxman networks are up-down (Figure 7(a)). Thus, the former networks are extremely hierarchical and the latter networks have little hierarchy as judged by both hierarchy metrics. This is consistent with what we know about the nature of hierarchy (or lack thereof) in these graphs. The AS and PLRG graphs, consistent with the link value distribution results, have a level of hierarchy intermediate between the Tree/TS and the Random/Waxman pairs.

However, this up-down metric does yield some differences with the previous metric. The Mesh, which has a surprisingly hierarchical distribution of link values, has very few up-down paths, the lowest of any of the networks we tested. Thus, the second hierarchy metric signals the Mesh's lack of hierarchy. Note that the percentage of valid paths is quite high (Figure 7(b)), but many of these paths are either rising or falling or flat, and very few of them are up-down.

The two somewhat unexpected results in Figure 7 are the low values, using the strict definition, for the up-down paths for the RL and Tiers. These numbers increased significantly under the relaxed definition of up-down (and were the only networks whose numbers changed significantly). The RL graph has, under the relaxed definition, roughly the same fraction of up-down paths as the AS and PLRG graphs. Thus, the three graphs appear to have very similar levels of hierarchy. The lower fraction of up-down paths for the strict up-down definition suggests that the RL graph has many paths where the link values have slight variations, but are still generally consistent with an up-down trajectory. In contrast, the Tiers graph remains well below the TS graph even under the relaxed definition. The construction of the Tiers network, even though it is based on a hierarchical organization, has links at the highest level that are not heavily used. These links have low values, and thus when they are traversed they result in large dips (large enough that even our modified definition does not accept them as up-down).

These two measures of hierarchy suggest that TS and Tree have very strict hierarchies, Mesh, Random, and Waxman have very little hierarchy, and Tiers has some significant concentration of paths (as indicated by the distribution of values) but that many of its paths are not up-down. The AS and RL graphs have very similar levels of hierarchy, representing a looser structure than TS or Trees, but much more hierarchical than, say, the Random or Waxman graphs.

The results on the PLRG network are surprising in two ways. First, PLRG has a degree of hierarchy that is extremely close to that of the AS and RL graphs. The parameters of the PLRG graph were not tuned to make this match happen, so we find the high degree of agreement somewhat surprising. Second, and more fundamentally, we were surprised that the PLRG graph has any hierarchy at all! The nodes in the PLRG graph are connected randomly, so where does this hierarchy come from?

### 5.3 Correlation between link usage and degree

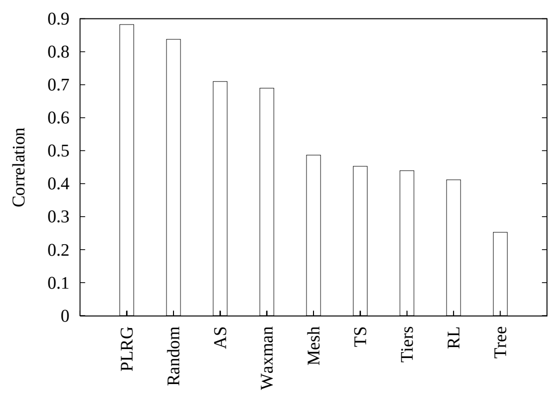



Figure 8: Correlation between minimum degree and link value

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 heavily used links connect high degree nodes. Figure 8 shows the correlations for the nine networks under consideration.

The PLRG and Random networks have extremely high correlations. There is absolutely no explicit structure built into these two graphs. The only links that have (relatively) high values are the ones that connect two nodes with (relatively) high degrees. 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. In contrast, 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 heavily used 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.

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.

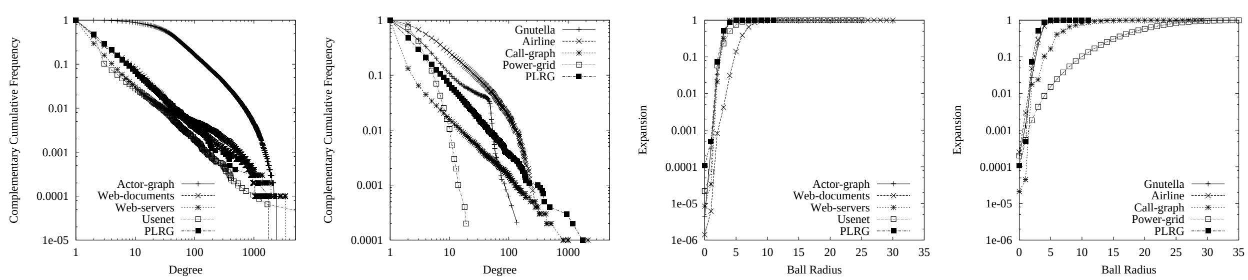
The AS and Waxman graphs have relatively high correlation, while the Mesh, TS, Tiers, and RL have relatively low levels of correlation. This is consistent with our reasoning above, that the hierarchy in the structural generators (Tiers and TS) arises, like the Tree, from the deliberate placement of links. 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 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 Other Networks

Our third question is: Do our conclusions apply to other “real” networks? To answer this question, we obtained instances of the following networks:

- The *power grid* network of the Western United States and Canada. Nodes on this network are generating stations and distribution stations, and links represent transmission lines.
- The *airline* network in which nodes are airports, and a link between two nodes represents the existence of a direct flight between them.
- The *actors* graph representing collaborations between actors in movies.
- A six-hour segment of a *call-graph* from the AT&T telephone network.
- The *Usenet* news server network. Nodes are news servers and links represent server feeds. This network was inferred from server-level path information in Usenet message headers.
- The network of connections between *gnutella* users.
- The network in which nodes are Web servers, and a link between two nodes denotes the existence of a page in one that references a page in the other.[25] We call this the *Web-server* network.
- Finally, the *Web-document* network, in which nodes are documents, and two nodes are linked if one contains a hyperlink to the other.



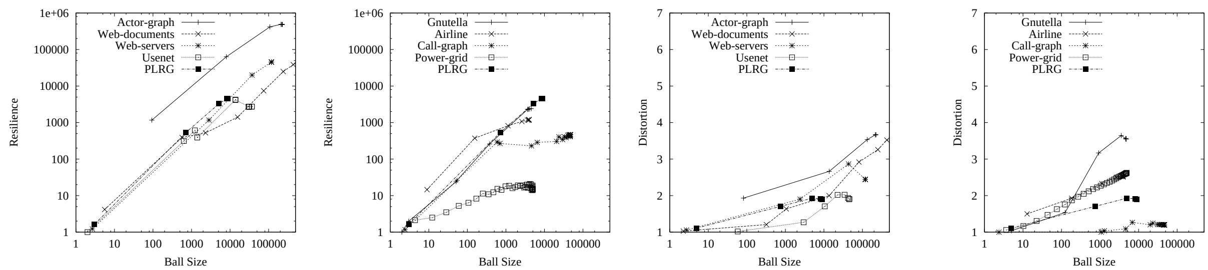

(a) Degree Distribution (b) Degree Distribution (c) Expansion (d) Expansion

Figure 9: Our metrics for other real networks

Figures 9 (a) and (b) plot the degree distributions of these networks. All networks, with the exception of the power-grid and possibly gnutella, have long-tailed degree distributions. The web-document and web-server graphs, as well as the call graph, appear to exhibit power-law degree distributions. The rest have other forms.

The expansion plots (Figures 9(c) and (d)) reveal that the power-grid clearly has low (Section 3.2.1) expansion. Similarly, the resilience graphs (Figures 10(a) and (b)) indicate that the power-grid has low resilience. Of the graphs shown above to have long-tailed degree distributions, the call-graph is the only one that is not qualitatively

[25] Although this graph is directed, by definition, we consider the undirected version in computing our metrics.




(a) Resilience (b) Resilience (c) Distortion (d) Distortion

Figure 10: Our metrics for other real networks

consistent with the PLRG. Its distortion (Figures 10(c) and (d)) is significantly lower than the others, and its resilience is slightly different from that of the other networks.

Given our computational resources, it was infeasible to compute the link usage values for most of these "real" networks. Of the graphs with long-tailed degree distributions, only the airline network was amenable to link usage value computation. We found that its hierarchy signatures matched that of the PLRG.

From this analysis, we conclude that many other real-world networks may be well-modeled by the PLRG generator. This is interesting, and suggests that, in looking for the reasons why the Internet is well-modeled by degree-based generators, we should perhaps look for explanations that are not necessarily specific to data communication networks.

# 7 Discussion

The work presented here is preliminary, and is only a first step towards an understanding of the issues we've addressed. The data on which we based our analysis – the measured network graphs – have several methodological drawbacks. They are incomplete, in that some nodes and links are missing. Moreover, the graphs only show connectivity, and do not reflect the link speeds nor policy routing. We are exploring ways to improve these aspects of the graphs, but have made little progress so far.

Our topology metrics also present problems. The selection of metrics is inherently arbitrary, and our choices may not reflect the most relevant aspects of networks. To broaden our treatment, we augmented our study, as described in Appendix B, with a collection of metrics from the literature; the results from these other metrics appears to be consistent with those from our metrics. However, the analysis of all of these metrics is qualitative, and therefore somewhat subjective. Subsequent work from other researchers will be needed to ensure that our own private biases did not distort the results.

With these caveats duly noted, our results suggest, somewhat tentatively, that:

- While the AS and RL graphs describe the Internet on very different levels of granularity, their properties, as measured by our metrics, seem quite similar.
- Degree-based generators are reasonably good models of the measured networks, and are significantly better than structural generators.[26]

[26] While the focus in this paper has been on which family of generators best model the Internet, we should note that we have been restricting our attention to rather large graphs (the smallest generated graph had 1000 nodes). Choosing a small (less than, say, 100 nodes) topology on which to run network simulations is an entirely separate question. As recently noted in [38], 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.

- 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.
- 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.
- Power-law random graphs are also good models for a wide variety of other real-life network structures.

## Acknowledgments

The IRL at the University of Michigan and the Center for Grid Technologies at ISI allowed us the use of their resources for our computations. We are particularly grateful to Thomas E. Stoklosa and OAG for information about airline connectivity. Hyunseok Chang and AT&T provided us with the call graph. Finally, Ashish Goel gave us valuable feedback on earlier versions of the paper.

## References

[1] AIELLO, W., CHUNG, F., AND LU, L. A Random Graph Model for Massive Graphs. In *Proc. of the 32nd Annual Symposium on Theory of Computing* (2000).

[2] ALBERT, R., AND BARABASI, A.-L. Topology of Evolving Networks: Local Events and Universality . *Physical Review Letters 85* (2000), 5234–5237.

[3] ALBERT, R., JEONG, H., AND BARABASI, A.-L. Attack and Error Tolerance of Complex Networks. *Nature 406* (2000).

[4] BARABASI, A.-L., AND ALBERT, R. Emergence of Scaling in Random Networks. *Science 286* (1999), 509–512.

[5] BOLLOBÁS, B. *Random Graphs*. Academic Press, Inc., Orlando, Florida, 1985.

[6] BURCH, H., AND CHESWICK, B. Mapping the Internet. *IEEE Computer 32*, 4 (April 1999), 97–98.

[7] CALVERT, K., DOAR, M., AND ZEGURA, E. Modelling Internet Topology. *IEEE Communications Magazine* (June 1997).

[8] CHALMERS, R. C., AND ALMEROTH, K. C. 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).

[9] CLAFFY, K. C., AND MCROBB, D. Measurement and Visualization of Internet Connectivity and Performance. http://www.caida.org/Tools/Skitter/.

[10] CORMEN, T., LEISERSON, C., AND RIVEST, R. *Introduction to Algorithms*. McGraw-Hill, 1990.

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

[12] DOWNEY, A. B. Using pathchar to Estimate Link Characteristics. In *Proceedings of the ACM SIGCOMM* (1999).

[13] FALOUTSOS, C., FALOUTSOS, P., AND FALOUTSOS, M. On Power-Law Relationships of the Internet Topology. In *Proceedings of the ACM SIGCOMM* (Sept. 1999).

[14] GAO, L. Inferring autonomous system relationships in the internet. In *Proc. IEEE Globecom* (San Francisco, CA, 2000).

[15] GOEL, A., AND MUNAGALA, K. Extending Greedy Multicast Routing to Delay Sensitive Applications. Tech. rep., Stanford Univ. Tech Note STAN-CS-TN-99-89, July 1999. Short abstract appeared in the Symposium on Discrete Algorithms, 2000.

[16] GOVINDAN, R., AND TANGMUNARUNKIT, H. Heuristics for Internet Map Discovery. In *Proceedings of the IEEE Infocom* (Tel-Aviv, Israel, March 2000).

[17] HU, T. C. Optimum Communication Spanning Trees. *SIAM Journal of Computing 3* (1974), 188–195.

[18] JIN, C., CHEN, Q., AND JAMIN, S. Inet: Internet Topology Generator. Tech. Rep. CSE-TR-433-00, EECS Department, University of Michigan, 2000.

[19] KARYPIS, G., AND KUMAR, V. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. *SIAM Journal on Scientific Computing 20*, 1 (1998), 359–92.

[20] KLEINBERG, J., KUMAR, S. R., RAJAGOPALAN, S., RAGHAVAN, P., AND TOMKINS, A. The Web as a Graph: Measurements, Models and Methods. In *International Conference on Combinatorics and Computing* (1999).

[21] LAI, K., AND BAKER, M. G. Measuring Link Bandwidths Using a Deterministic Model of Packet Delay. In *Proceedings of the ACM SIGCOMM* (2000).

[22] MEDINA, A., AND MATTA, I. BRITE: A Flexible Generator of Internet Topologies. Tech. Rep. BU-CS-TR-2000-005, Boston University, 2000.

[23] MEDINA, A., MATTA, I., AND BYERS, J. On the Origin of Power-Laws in Internet Topologies. *ACM Computer Communications Review 30*, 2 (April 2000).

[24] PALMER, D., AND STEFFEN, G. On Power-Laws In Network Topologies. In *Proceedings of IEEE Globecom* (2000).

[25] PANSIOT, J.-J., AND GRAD, D. On routes and multicast trees in the Internet. *ACM SIGCOMM Computer Communication Review 28*, 1 (January 1998), 41–50.

[26] PARK, K. Impact of topology on traceback techniques. Private communication.

[27] PHILLIPS, G., SHENKER, S., AND TANGMUNARUNKIT, H. Scaling of Multicast Trees: Comments on the Chuang-Sirbu Scaling Law. In *Proceedings of the ACM SIGCOMM* (Sept. 1999).

[28] RADOSLAVOV, P., TANGMUNARUNKIT, H., YU, H., GOVINDAN, R., SHENKER, S., AND ESTRIN, D. On Characterizing Network Topologies and Analyzing Their Impact on Protocol Design. Tech. Rep. 00-731, University of Southern California, Dept. of CS, February 2000.

[29] RÉNYI, A. On the Enumeration of Trees. In *Combinatorial Structures and Their Applications* (June 1969), Gordon and Breach, Science Publishers, pp. 355–360.

[30] SAVAGE, S., COLLINS, A., HOFFMAN, E., SNELL, J., AND ANDERSON, T. The End-to-End Effects of Internet Path Selection. In *Proceedings of ACM SIGCOMM* (Boston, MA, September 1999).

[31] SIAMWALLA, R., SHARMA, R., AND KESHAV, S. Discovering Internet Topology. Unpublished manuscript.

[32] TANGMUNARUNKIT, H., GOVINDAN, R., SHENKER, S., AND ESTRIN, D. The Impact of Policy on Internet Paths. In *To appear, Proc. of IEEE INFOCOM* (Anchorage, AK, 2001).

[33] VAN DER HOFSTAD, R., HOOGHIEMSTRA, G., AND VAN MIEGHEM, P. On the Efficiency of Multicast. Submitted for publication.

[34] VAN MIEGHEM, P., HOOGHIEMSTRA, G., AND VAN DER HOFSTAD, R. A scaling law for the hopcount. Tech. rep., Delft University of Technology, 2000.

[35] WATTS, D. J., AND STROGATZ, S. H. Collective Dynamics of Small-World Networks. *Nature 363* (1998), 202–204.

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

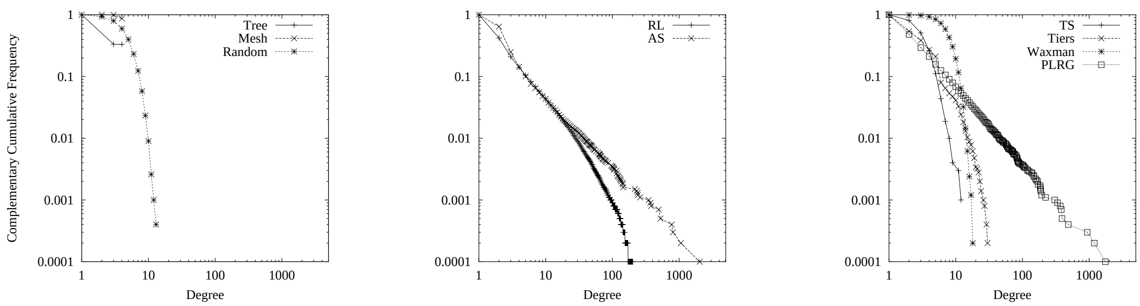
[37] WONG, T., AND KATZ, R. An Analysis of Multicast Forwarding State Scalability. In *Proceedings of the 8th IEEE International Conference on Network Protocols (ICNP 2000)* (Osaka, Japan, November 2000).

[38] ZEGURA, E. Thoughts on Router-level Topology Modeling. The End-to-end interest mailing list.

[39] ZEGURA, E., CALVERT, K. L., AND DONAHOO, M. J. A Quantitative Comparison of Graph-Based Models for Internet Topology. *IEEE/ACM Transactions in Networking 5*, 6 (1997).

# A Degree Distributions of Generated and Real Networks

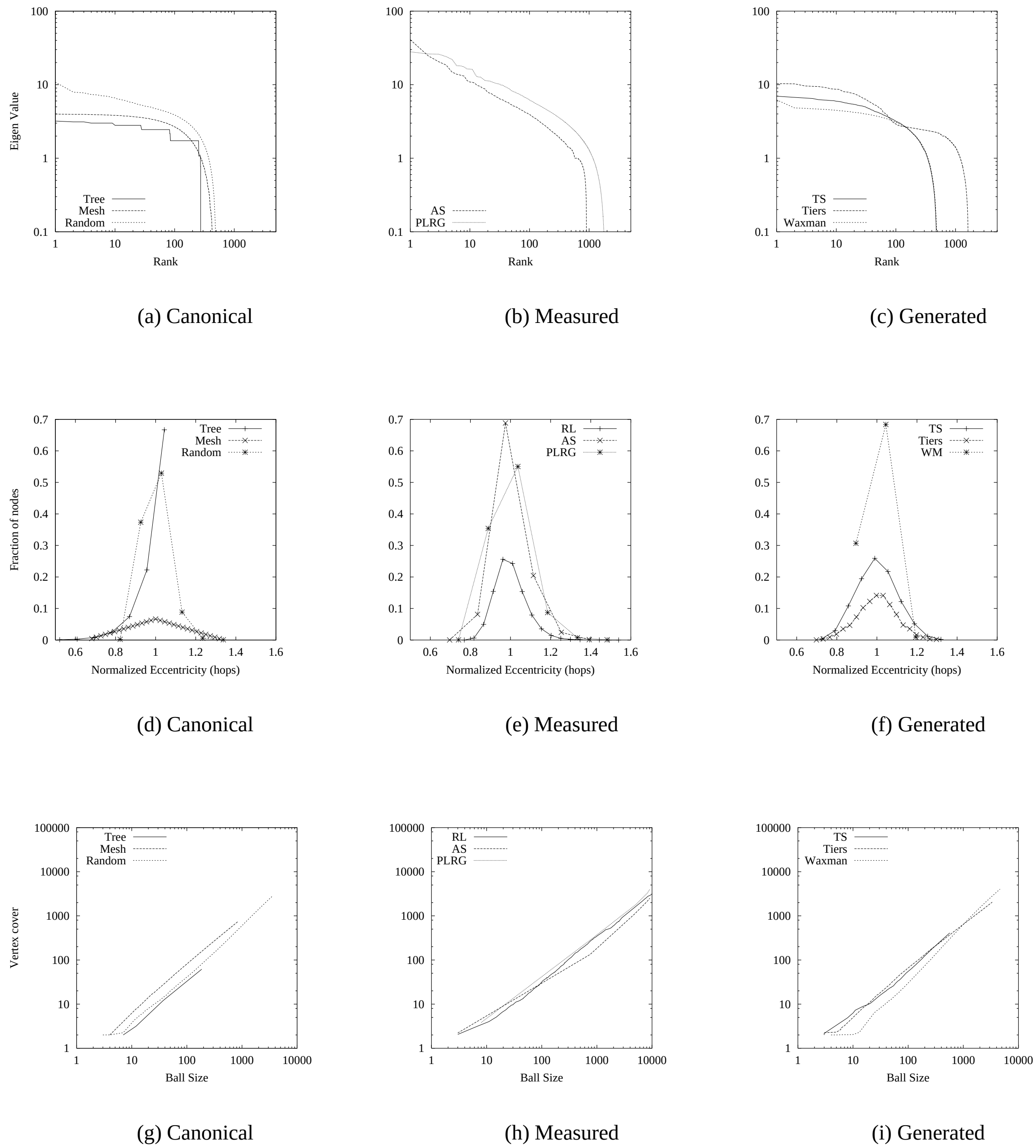
We first present data on the degree distribution of the three real networks, along with the generated networks. We include this merely to confirm the Faloutsos conclusions (at least for the AS graph).




(a) Canonical (b) Real (c) Generated

Figure 11: Degree Distributions for various graphs

# B Results for Other Metrics

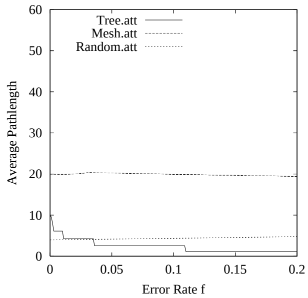



(a) Canonical (b) Measured (c) Generated

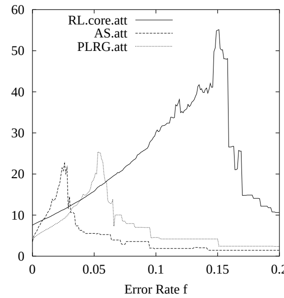
(d) Canonical (e) Measured (f) Generated

(g) Canonical (h) Measured (i) Generated

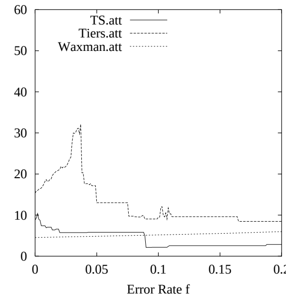
Figure 12: Plots (a)-(c) depict the distribution of eigenvalues of a graph plotted against their rank [13]. Plots (d)-(f) depict the distribution of node diameters. This is a modified version of the graph diameter metric proposed in [39]. Finally, graphs (g)-(i) depict the vertex cover of the subgraphs within balls of radius $n$.



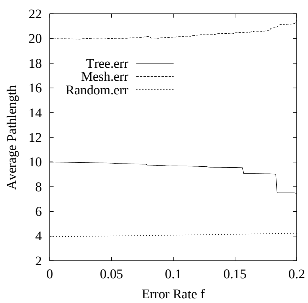

(a) Canonical, attack



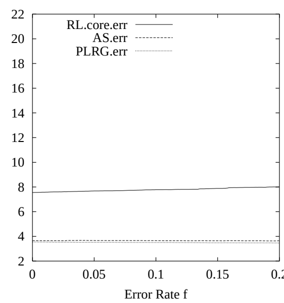

(b) Measured, attack



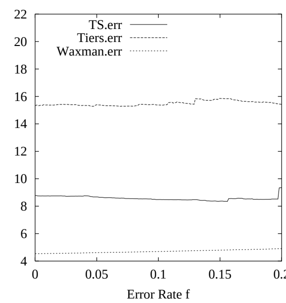

(c) Generated, attack




(d) Canonical, error

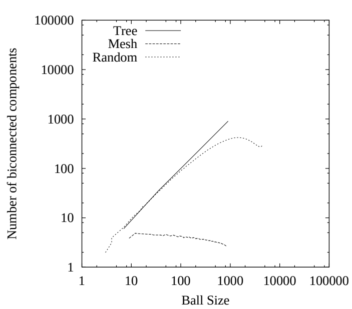



(e) Measured, error

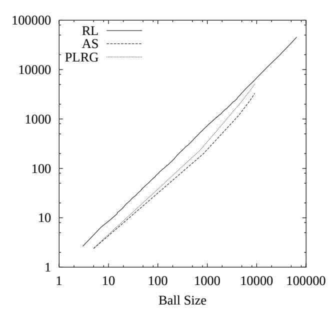



(f) Generated, error

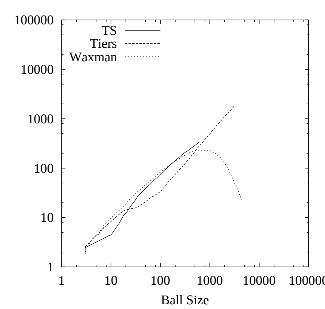
Figure 13: Figures (a)-(c) depict the attack tolerance [3] of our networks. This measures the average pathlength of the largest connected component when increasingly larger fractions of nodes are removed, in order of decreasing degree. Figures (d)-(f) plot the error tolerance; the average path length when nodes are removed randomly.




(a) Canonical




(b) Measured




(c) Generated

Figure 14: The number of biconnected components within a subgraph defined by a ball of size $n$, as a function of ball size.

# C Parameter Space Exploration

For a given sized graph, the power-law random graph takes a single parameter: $\beta$ (Section 3.1).

The parameters of Transit-Stub (TS) are listed in the order they appear in the table: the number of stub domains per transit-node, the number of random transit-to-stub edges, the number of random stub-to-stub edges, the number of transit domains, the edge probability among transit domains, the number of nodes per transit domain, the edge probability among nodes in a transit domain, the number of nodes per stub domain, and the edge probability among nodes in a stub domain

The parameters of Tiers are listed in the order they appear in the table: the number of WANs (limited to 1 in the current implementation), the number of MANs per WAN, the number of LANs per MAN, the number of nodes per WAN the number of nodes per MAN, the number of nodes per LAN, the intranetwork redundancy for WAN nodes, the intranetwork redundancy for MAN nodes, the intranetwork redundancy for LAN nodes, the internetwork redundancy for MAN to WAN, and the internetwork redundancy for LAN to MAN.

The parameters of the Waxman generator include the number of nodes in the topology, an $\alpha$ value, and a $\beta$ value (the latter governs the extent of geographic bias and the former the link probability).

| Topology | Number of Nodes | Average Degree | Comment |
|---|---|---|---|
| PLRG | 8037 | 2.79 | 2.550144 |
| | 9114 | 3.47 | 2.358213 |
| | 9230 | 4.46 | 2.246677 |
| | 10091 | 4.61 | 2.253182 |
| TS | 1008 | 2.78 | 3 0 0 6 0.55 6 0.32 9 0.248 |
| | 1008 | 2.51 | 3 0 0 6 0.6 6 0.45 9 0.57 |
| | 2550 | 2.89 | 1 0 0 1 0.5 50 0.05 50 0.05 |
| | 2550 | 2.89 | 1 5 5 1 0.5 50 0.05 50 0.05 |
| | 2550 | 2.89 | 1 10 10 1 0.5 50 0.05 50 0.05 |
| | 2550 | 5.01 | 1 0 0 1 0.5 50 0.1 50 0.1 |
| | 5550 | 3.44 | 3 8 12 10 0.4 15 0.25 12 0.27 |
| | 10100 | 4.98 | 1 0 0 1 0.2 100 0.05 100 0.05 |
| Tiers | 1000 | 2.81 | 1 20 4 200 20 5 9 9 1 9 1 |
| | 5000 | 2.83 | 1 50 10 500 40 5 20 20 1 20 1 |
| | 10000 | 2.37 | 1 100 10 1000 50 4 3 3 1 3 3 |
| | 10000 | 2.47 | 1 100 10 1000 50 4 6 6 1 3 3 |
| | 10000 | 2.68 | 1 100 10 1000 50 4 10 10 1 10 3 |
| | 10000 | 3.09 | 1 100 10 1000 50 4 20 20 1 20 3 |
| | 10000 | 2.35 | 1 50 20 1000 100 4 3 3 1 3 3 |
| | 10500 | 2.72 | 1 50 50 500 100 2 3 3 1 3 3 |
| | 10500 | 2.12 | 1 100 0 500 100 0 6 6 1 3 3 |
| Waxman | 1000 | 5.06 | 1000 0.050 0.20 |
| | 1762 | 2.03 | 5000 0.005 0.05 |
| | 4476 | 2.82 | 5000 0.005 0.10 |
| | 5000 | 7.22 | 5000 0.005 0.30 |
| | 5000 | 10.82 | 5000 0.005 0.50 |
| | 4444 | 2.79 | 5000 0.010 0.05 |
| | 4967 | 5.03 | 5000 0.010 0.10 |
| | 5000 | 14.42 | 5000 0.010 0.30 |

Figure 15: Parameters explored for structural generators