A Multi-Level Approach for Evaluating Internet Topology Generators

Ryan Rossi
Department of Computer Science
Purdue University
rrossi@cs.purdue.edu

Sonia Fahmy
Department of Computer Science
Purdue University
fahmy@cs.purdue.edu

Nilothpal Talukder
Department of Computer Science
Rensselaer Polytechnic Institute
talukn@rpi.edu

***Abstract*—The topology of a network (connectivity of autonomous systems (ASes) or routers) has significant implications on the design of protocols and applications, and on the placement of services and data centers. Researchers and practitioners alike need realistic topologies for their simulation, emulation, and testbed experiments. In this work, we propose a multi-level framework for analyzing Internet topologies and their evolution. Our multi-level framework includes novel measures, evaluation strategies, and techniques for automatically learning a representative set of graph measures. We apply our framework to analyze topologies from two recent topology generators, Orbis and WIT, according to how well they achieve their advertised objectives. The generated topologies are compared to a set of benchmark datasets that approximate different views of the Internet in the data (trace-route measurements), control (BGP), and management (WHOIS) planes. Our results demonstrate key limitations of popular generators, and show that the recent Internet clustering coefficient and average distance are not time-invariant as assumed by many models. Additionally, we develop a taxonomy of topology generators, and identify key challenges in topology modeling.**

I. Introduction

Accurate representation of network topologies plays a critical role in designing protocols [30], predicting performance [22], and understanding robustness and scalability of the future Internet [23], [18], [31]. Many current topology generators attempt to capture key static properties of the Internet or to capture Internet evolution. The most recent Internet topology generators [26], [38], [36], [25], [4] aim at generating representative topologies of different sizes (number of nodes); however, understanding and modeling the driving forces behind Internet evolution remains a significant challenge.

Looking back on Internet topology modeling research over the past ten years, Roughan *et al.* [33] note that the majority of prior work makes simplifications that lead to misleading findings and ill-conceived ideas about the Internet. In this work, we examine this prior research on topology modeling quantitatively. We propose a taxonomy for topology generators, analyze the claims they make, and introduce a multi-level evaluation framework. In doing so, we seek to address the following questions: **(1)** How do we quantitatively assess a topology generator? and **(2)** How accurately do current topology generators model both the static and the evolutionary properties of the Internet?

Several previous studies examined the average degree of the Internet Autonomous System (AS) graph, its average path length, and its clustering coefficient and showed them to remain constant as the Internet topology evolves, *e.g.,* [37], [26]. These findings are based on RouteViews datasets from the 1990s–2006 time frame. We will show different characteristics in more recent RouteViews data (Fig. 1 and 2). Further, several generators aim to maintain "any arbitrary metric" of a generated topology constant as the number of nodes increase, *e.g.,* [26], [27]. However, there is evidence of the Internet topology transitioning from a hierarchical to a flat topological structure [14], [13], [24]. These observations of a change in the characteristics of the Internet are consistent with our findings, and explain why topology generators that assumed a prior structure or process based on previous patterns may fail to predict recent evolution of the Internet topology.

We compare generated topologies to real and synthetic datasets based on a multi-level approach that includes novel measures, evaluation strategies, and techniques for automatically learning a representative set of graph measures. In contrast to prior work that assumed that certain measures are sufficient and necessary to evaluate generated topologies, we find that the measures previously studied are not enough to demonstrate the superiority of a generator over another. There are often other factors to consider. For instance, we found that under certain conditions, Orbis generates topologies that are approximately isomorphic to the initial topology used as input. Therefore, any graph metric used to evaluate these topologies will match almost exactly. As a result, it may be important to require some variance between the graph used as input and the generated graphs.

The contributions of this paper can be summarized as follows. *First,* we propose a novel multi-level framework for evaluating topologies and their evolution. This multi-level framework includes graph, node, and link measures, and learning and evaluation methods. *Second,* we study the clustering coefficient and average distance in recent Internet AS topology data, and demonstrate that they are not time-invariant as assumed by earlier work. *Third,* we apply our multi-level framework to evaluate two recent popular topology generators, Orbis [26] and WIT [38], according to whether they produce graphs that match their advertised objectives. The topology generators are compared to a variety of datasets that approximate different views of the Internet at the *data* (traceroute measurements [9]), *control* (BGP [2]), and *management* (WHOIS [1]) planes. We also present a few results on the RocketFuel [34] and the HOT [25] router-level topologies.

We structure the remainder of the paper as follows. Section II presents a taxonomy for topology generators. We also discuss in detail two recent topology generators, Orbis and WIT. In section III, we propose a framework for analyzing network properties based on matrix factorization techniques that range across three-levels of structural granularity. Section IV describes the datasets and process used to compare real and synthetic topologies. In section V, we evaluate the topology models using our multi-level approach. We conclude with a summary of our main findings.

II. Taxonomy and Challenges

We introduce a taxonomy of prominent topology generators in Table I. This taxonomy is based on the main principle used in generating topologies (random-walk, optimization, preferential attachment, geometry), the type (parametric, nonparametric), and the topology (AS, router-level (RL)) constructed.

Parametric generators assume a particular functional form or mechanism, whereas data-driven generations (nonparametric models) make fewer assumptions about the functional form. A generator may estimate parameters from data or simply assume an underlying mechanism. If an underlying mechanism or principle is used in a topology generator (*e.g.,* WIT), the generator is tailored to a specific type of topology such as the Internet AS topology, whereas topology generators that perform parameter estimation from data can typically generate many types of networks (*e.g.,* social or biological networks).

Models based on *preferential attachment* [3], [5], [28], [21], [8], [40] may be too restrictive to model the Internet evolution, since the decision to link to another node is based on degrees of nodes, giving high importance to highly connected nodes. Generators based on random-walks can suffer from similar problems. Optimization-based generators [11], [15] consider solving the *optimization* problem between the benefit and improved connectivity of the network. However, economic considerations of the connectivity are not typically considered in the optimization. GT-ITM [10] is one of the earliest generators and is primarily based on the *hierarchical* nature of the Internet, which appears to be changing [14], [13], [24]. Tangmunarunkit *et al.* [35] find that degree-based models match a set of measures better than these early hierarchical models. Though recent generators accurately match the power-law distribution, they fail to capture certain network measures summarized in Table II.

TABLE I
TAXONOMY OF TOPOLOGY GENERATORS

GENERATORS	PROCESS	Model Type	Topology
WIT [38]	**Random-walks**	Parametric	AS
RSurfer [7]		Parametric	N/A
Orbis [26]	**Optimization**	Data-driven	AS & RL
HOT [25]		Parametric	RL
Mod. HOT [11]		Parametric	AS
AB [3]	**Preferential Attachment**	Parametric	N/A
BRITE [28]		Data-driven	AS & RL
Inet [21]		Parametric	AS
GLP [8]		Parametric	AS
SWT [22]	**Geometry**	Parametric	AS & RL
GT-ITM [10]		Parametric	AS & RL

Below, we formally describe two recent and well-cited Internet topology generators, Orbis and WIT, which we use as case studies in the remainder of this paper.

A. Orbis: From Degree to Topology

Orbis uses a series of measures based on degree correlations that monotonically capture more global structures [26]. An initial topology is given as input and randomly rewired such that the dK-distribution of the input topology is preserved. The first few dK distributions are: $0K$ (average degree), $1K$ (degree distribution: $P(k) = n(k)/n$), $2K$ (joint degree distribution: $P(k_1, k_2) = m(k_1, k_2)\mu(k_1, k_2)/(2m)$, where $\mu(k_1, k_2) = 2$ if $k_1 = k_2$, otherwise 1), $3K$ (wedges and triangles), and so forth. The topologies are rescaled by simply stretching the target distribution and preserving it under random rewiring. The chosen value of d must be small in practice due to the increase in complexity as d increases. Additionally, as d increases, the space of possible graphs that can be generated exponentially shrinks, yielding topologies that are only slightly different from the input training topology. Of course, if the input graph varies only slightly from the generated graph, then both graphs exhibit nearly identical characteristics, and the generated graph is not too useful. With small values of d, Orbis is limited to preserving only local characteristics. As we will show, constructing graphs with $d = \{0, 1, 2, 3\}$ captures a few related local measures, but fails to capture global characteristics. Furthermore, as the generated graphs are rescaled, the accuracy of capturing the local measures depends on the rescaling technique, becoming increasingly inaccurate as the size of the topology increases.

B. WIT: From Random Walks to Topology

The WIT model uses a simple multiplicative stochastic process, $u_i(t) = \lambda_i(t)\ u_i(t-1)$, where $u_i(t)$ is the unscaled wealth of node i at time t and $\lambda_i(t)$ is a random variable. This process captures the "wealth" of ISPs over time [37], [38]. The wealth (or weight) for each ISP is used to add or remove links based on a given threshold.

More formally, the wealth of a node is updated each iteration (or time). If the updated node weight exceeds a given threshold $w_i(t) - z_i(t) > C + T$, then a link is added by randomly walking l-steps until an arbitrary node z is reached and a link is placed between the nodes. In the above threshold check, $w_i(t)$ is the normalized wealth for node i, and $z_i(t) = C \cdot d_i(t)$ (depends on the current degree $d_i(t)$

of node i and the expected link cost $C = w_0 \cdot c$, where w_0 and c are only some of the input parameters). Furthermore, T must be carefully chosen to avoid an oscillating (degenerate) case where a link is added and then the next time deleted, indefinitely. Similarly, if the node weight is below a threshold $w_i(t) - z_i(t) > -T$, then the node randomly chooses one of its links to remove. A new node x is added and a link is initially created by randomly selecting a node y and linking to that node. Additionally, a second link is added by randomly walking l-steps as described previously.

C. Challenges in Topology Generation

Topology modeling remains a difficult challenge [33]. As discussed above, some models like WIT do not estimate parameters using an input topology, but rather rely on the accuracy of assumptions made about the underlying growth mechanism and the ability of this mechanism to accurately match the forces driving the evolution of the Internet topology. Selecting the parameters for WIT requires detailed knowledge of the model and the behavior of the Internet over time, which is not very well-understood. The lack of parameter estimation makes WIT difficult to use in simulations or other practical applications. The initial evaluation of WIT in [38] uses the optimal parameters for RouteViews, but provides little intuition for selecting these parameters for a given AS topology.

TABLE II
MEASURES AND RELATED NETWORK CHARACTERISTICS

	Measure	Importance in Computer Networks
LOCAL	Degree	Fault tolerance, local robustness
	Assortativity	
	Clustering coefficient	Path diversity, fault tolerance, local robustness
GLOBAL	Distance	Scalability, performance, protocol design
	Betweenness	Traffic engineering, potential congestion points
	Eigenvector	Network robustness, performance, clusters/hierarchy, traffic engineering

Consider the space of all graphs that preserve a set of measures and suppose we select an arbitrary graph from this space. This graph may preserve the set of measures, but not capture the characteristics of the system (*e.g.,* AS topology). Therefore, a graph that preserves a set of measures is not guaranteed to capture the important characteristics of the system under investigation. From this perspective, WIT attempts to model the evolution of the AS topology directly, whereas Orbis generates topologies that preserve a set of measures. Thus, the WIT model would fail when the underlying process used in the formation and growth of the Internet changes or if the model does not accurately mimic the underlying process, whereas Orbis would fail if the set of characteristics are incomplete with respect to the characteristics implied by the actual AS topology.

III. A MULTI-LEVEL FRAMEWORK

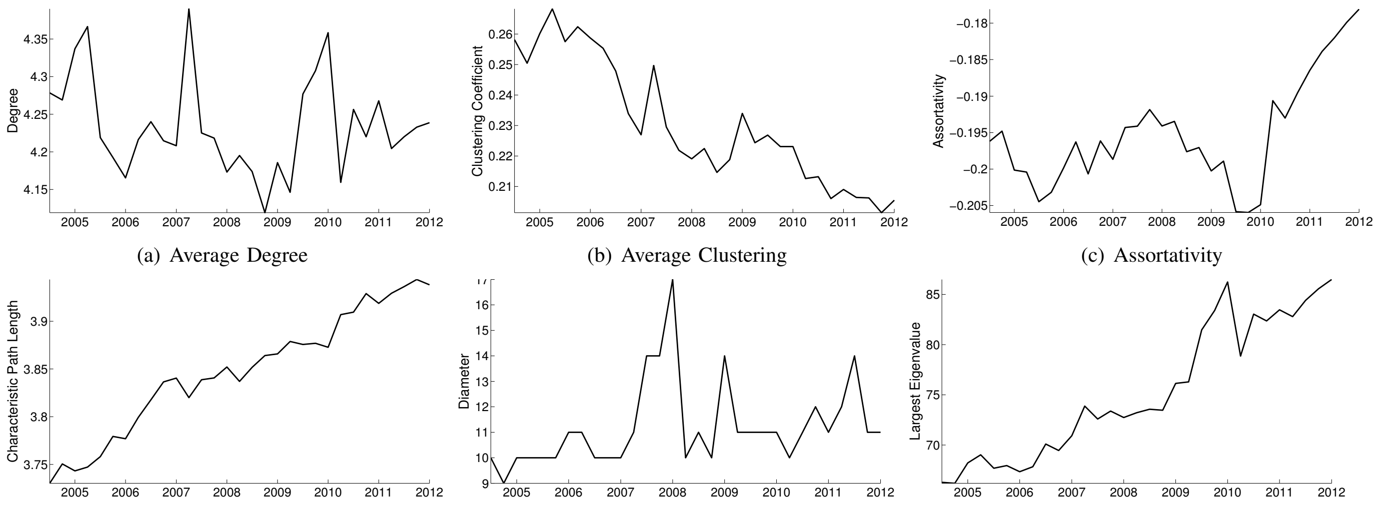
At the heart of our topology analysis framework are multi-level evaluation criteria based on three general classes of measures: graph, node, and link measures. Graph measures (i.e., scalar measures) compute a single value for the entire network graph, whereas node and link measures compute a value for each node or link, respectively.

The three evaluation granularities form an approximate ordering, link measures $\geq$ node measures $\geq$ graph measures, according to the difficulty of preserving a property from that level. For instance, constructing a graph that preserves the average degree is easier than constructing a graph that preserves the degree distribution. Clearly, the node and link measures capture graph structure and connectivity better than single point statistics. A topology generator that consistently preserves graph measures is said to be validated at this level (i.e., the least constrained evaluation granularity).

The majority of previous work was not sufficiently quantitative in its evaluation of topology generators, leading to an incomplete and often misleading depiction of their accuracy and representativeness. Some generators are evaluated using only single point statistics while others focus on only local properties such as degree and average local-clustering coefficient. Further, most of the previous work did not examine community structure.

As a basis for our multi-level approach, we use a number of *graph*, *node*, and *link* measures using matrix factorizations. These measures and their relation to AS-level and router-level topologies have only received limited attention. The majority of previous work uses the spectrum of the Laplacian matrix (which indicates connected components and sparse cuts), and ignores the adjacency matrix (which can indicate properties such as the number of paths). Nevertheless, matrix-factorization-based characteristics have been described by some as critically important [27], [17], [12], yielding tight bounds for (1) distance-related characteristics, (2) clustering-related characteristics, and (3) graph resilience under node/link removal. Most graphs with large eigenvalues exhibit small diameters, expand faster, and are more robust with respect to link or node removal [27]. The eigenvectors also cluster tightly connected or similar nodes, whereas large eigenvalues may imply more node and link-disjoint paths. In addition to capturing a significant amount of global information about the resulting topology, the eigenvectors also identify local characteristics such as degree-patterns and clustering of various sizes. Table II summarizes a number of measures and their corresponding network characteristics.

Much of the previous work evaluates topologies by visually comparing the similarity of measures. Instead, we propose using distance or divergence measures to more *quantitatively* compare graph, node, or link measures. Further, instead of choosing a set of measures, we use an approach that *automatically learns* a set of "representative" node measures based on the original graph. Our approach then computes these same features using the generated graphs. After that, the



(a) Average Degree (b) Average Clustering (c) Assortativity

(d) Characteristic Path Length (e) Diameter (f) Largest Eigenvalue

Fig. 1. **Graph Measures.** Evolutionary characteristics of the AS topology as observed from RouteViews (2004 to 2012). The majority of previous models assume the Internet topology (and measures applied to it) are time-invariant. However, as shown above, we find evidence of a recent transition in the topological structure, most notably seen from 2007-present (28K and beyond) which is consistent with [24]. We observe the average clustering coefficient decreasing, indicating there are on average fewer triangles (triad relationships) being formed between ISPs, even as the number of ISPs and connections increase. This behavior also helps explain the increase in characteristic path length: the topology is becoming more spread out and flat due to fewer triangles being formed. Further, the assortativity coefficient is becoming more positive; ISPs are now connecting to others that have a similar amount of connections (gradually losing the hierarchical structure). These findings help explain the transition to a flat topology [14], [13], [24], while allowing us to understand the speed of this change and predict its future impact on the Internet.

node measures can be compared and evaluated. This approach eliminates the human bias of selecting measures to evaluate a generator.

A. Graph Measures

In addition to the traditional graph (or scalar) measures (*e.g.,* average degree k, assortativity coefficient r, average clustering $\bar{c}$, average distance ($\bar{d}$), we propose using the largest singular value (λ_1), network conductance ($\lambda_1 - \lambda_2$), radius, and diameter.

A positive assortativity coefficient indicates that nodes tend to link to nodes with similar degree. Note that the Internet AS graph has negative assortativity, although its value is changing.

A wedge is a 2-length path. The set of wedges W_u centered at u is given by $W_u = d_u(d_u - 1)/2$ where d_u is degree. A wedge $\{(u, w)(w, v)\}$ forms a *triangle* if there exists an edge (u, v). Let T_u be the set of triangles centered at u. Then, the local clustering coefficient is $C_u = |T_u|/|W_u|$. We use the average local-clustering coefficient $C = \sum_{u \in V} C_u$ [39], as opposed to the global clustering coefficient [29].

The difference between the two largest eigenvalues denotes the *network conductance* which is also known by some as the performance of a network [27]. Each of these scalar values is computed from the eigenvectors and eigenvalues (or, if appropriate, the singular-vectors or singular-values) of the graph adjacency matrix. Note that a graph measure may also be computed by applying any summary function (e.g., `avg`, `sum`, `max`, `min`, `var`) to a node measure.

B. Node Measures

To analyze the local and global node-level properties of networks, we primarily use the following measures:

$-$ *Network Values*. Plot of the eigenvector components (indicators of network value) corresponding to the largest eigenvalue.

$-$*Scree Plot*. Plot of the k largest eigenvalues (or singular-values) versus their normalized rank using a log-scale.

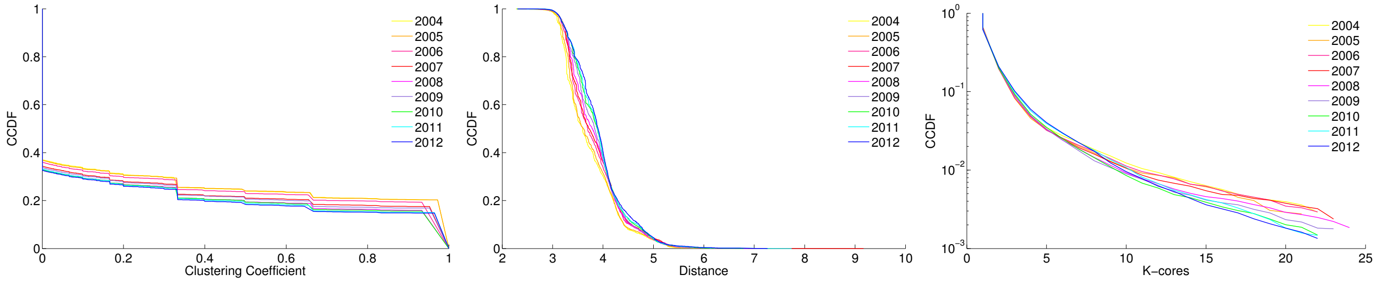
$-$*K-walks: A Class of Local and Global Measures*. We propose using a simple class of measures, denoted as *k-walks*, capable of measuring both local and global properties of graphs by adjusting a single parameter. A *k-walk* of a vertex u is the number of walks of length k rooted at u. The number of walks from node u to node v in a graph $\mathcal{G}$ with length k is $(\mathbf{A}^k)_{uv}$. The k-walk measure of a graph adjacency matrix $\mathbf{A}$ is given by $\sigma_k(\mathbf{A}) = \mathbf{A}^k \mathbf{e}$, where $\mathbf{e}$ is the unit vector. If $k \to \infty$ then we have the principal eigenvector and the other extreme where $k = 1$ results in the degree distribution. Intermediate values $1 \leq k \leq \infty$ give other properties of the graph going from the most local property of degree to the most global property of the principal eigenvector. This metric provides a formal way to bound the similarity of two graphs with respect to k.

$-$*K-core*. A k-core is a vertex induced subgraph where all vertices have degree at least k. The core number of a vertex v is the largest k such that v is in a k-core.

$-$*Others.* We also use traditional measures such as degree distribution, clustering coefficient, distance, eccentricity, betweenness, among others. Note that the number of triangles T_u and wedges W_u from Section III-A are also node measures.

C. Link Measures

Link measures lie at the finest evaluation granularity. First, we apply a technique to order the nodes with respect to the magnitude of their coordinates along the principal direction. This procedure reveals significant link structures, connectivity



(a) Clustering Coefficient (b) Distance (c) K-cores

Fig. 2. **Node Measures.** The node measures are changing between 2004 and 2012 (RouteViews), consistent with our findings in Fig. 1

patterns, and block structures/clustering. Second, we analyze the network characteristics more accurately by computing the closest k-approximation of the topology resulting in the weighting, suppression, or creation of links. Similar techniques have been used in information retrieval and various fields that require a low-dimensional representation that preserves the most significant information with minimum loss. Both methods allow us to identify the most significant properties preserved in the resulting topology.

D. Communities

We evaluate the *community structure* using Louvain's modularity [6], which was previously shown to be important for studying the Internet AS graph [19]. The evaluation strategy below is defined for a single topology, but the strategy is applied for each of the generated topologies and the corresponding benchmark topology. Our evaluation strategy is as follows:

1. Given a topology, extract communities. Let k_{max} denote the number of communities that have at least r nodes. We use $r = 100$ to discard small insignificant communities.
2. For each community $k = 1, ..., k_{max}$, induce the subgraph S_k such that S_k consists of all the nodes in the community k and the links between these nodes.
3. For each community subgraph $S_k = S_1, ..., S_{k_{max}}$, evaluate the connectivity properties using graph measures and node measures.

Once the community subgraphs have been extracted over the benchmark/generated topologies, there are several ways to evaluate them. One way is to compare the number of communities extracted for each topology. We also evaluate the largest community using graph and node measures.

E. Quantitative Metrics

Most previous work evaluates topologies by visually comparing values of individual measures. Instead, we use a few "metrics" to more quantitatively evaluate the generators.

Metrics for Graph Measures. For graph measures, we first select a set of measures and compute them for each graph. Let $\mathbf{x}$ be the "true" measures (e.g., from RouteViews) and $\hat{\mathbf{x}}$ be the estimated measures computed on a generated topology from an arbitrary generator, then the normalized root-mean-square error (NRMSE) is defined as:

$$D_{NRMSE}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{\mathbb{E}[(\mathbf{x} - \hat{\mathbf{x}})^2]}{\max(\mathbf{x}, \hat{\mathbf{x}}) - \min(\mathbf{x}, \hat{\mathbf{x}})}.$$

This measure is expressed as a percentage, where lower values indicate less residual variance. If the vector of measures $\mathbf{x}$ is identical to $\hat{\mathbf{x}}$ then the NRMSE is 0. If instead we consider a time-series $\mathbf{x}(t), 0 \leq t \leq t_{\max}$ of measures from real data and a time-series $\hat{\mathbf{x}}(t), 0 \leq t \leq t_{\max}$ from a generator, then NRMSE is simply computed at each time t. Since it is normalized, we can simply average the values over time giving us a single measure of how well a given generator tracks the important characteristics over time.

Metrics for Node Measures. For node measures, we use two metrics for quantitatively evaluating generators. The Kolmogorov-Smirnov (KS) statistic assesses the distance between two CDFs. The KS-distance is computed as the maximum distance between two distributions where x represents the range of the random variable and F_1 and F_2 represent two CDFs: $KS(F_1, F_2) = \max_x |F_1(x) - F_2(x)|$. KS varies between 0 and 1. We also use the Kullback-Leibler divergence (or simply KL divergence) to evaluate the difference between two PDFs. KL divergence computes the average number of bits required to represent a measure from the benchmark topology when using the measure distribution from the generated topology. KL measures the average number of extra bits required to represent the benchmark topologies original distribution when using the generated topologies distribution. It is defined formally as

$$D_{KL}(P||Q) = \sum_i P(i) \ln \frac{P(i)}{Q(i)}.$$

We add a small value ϵ to Q (and renormalize) before computing KL divergence since it is undefined for distributions that have some values with zero probabilities. One might also smooth the distribution, but both approaches provide similar results.

If needed, the values of each measure can be appropriately binned to create a distribution. Unless otherwise specified, we created 10 equally-spaced bins over the range of values from the true distribution P and used these bins for Q as well.

F. Learning Graph Measures

Instead of selecting a set of graph measures, we automatically learn a representative set of graph measures recursively [20].

1. The process starts by computing degree (in/out/total edges) and egonet measures (in/out egonet). Note the egonet includes the node, its neighbors, and any edges in the induced subgraph on these nodes.
2. After computing these, the existing measures of a node are aggregated using sum/mean. The measures are aggregated over all neighbors, and also over in/out edges.
3. Next, the algorithm prunes correlated measures (eliminating the redundant measures).
4. The aggregation proceeds recursively over the current set of measures, until no new measures are retained.

For more details, see [20]. While the extracted measures are more difficult to interpret, these measures are shown to capture the main connectivity patterns present in the graph and therefore they are extremely useful for comparing generators.

IV. Methodology

A. Datasets

We compare graphs from topology generators to AS topologies based on the Skitter traceroute [9], RouteViews' BGP tables (RV) [2], and RIPE's WHOIS [1] datasets. These are the same datasets used by Mahadevan *et al.* in [26] and all except RouteViews data were obtained from the authors' web site. In addition to these AS-level topologies, we compare to the HOT [25] and RocketFuel [34] router-level topologies. The HOT topologies were also retrieved from the web site of Mahadevan *et al.* The RocketFuel topologies were obtained from RocketFuel project site.

To study the most recent evolutionary characteristics of the Internet, we obtained time series of BGP routing tables (in Cisco and Zebra format) from the Oregon RouteViews project [2]. We estimated the AS-level topology by taking the union of all AS-paths in the routing tables as performed by Gao [16]. We extracted AS-level subgraphs for the years 2004 to 2012 from the last 25 BGP tables of March, June, September, and December of the corresponding year (except for 2012 when we consider the last 25 BGP tables from February 2012). Unlike [13], we do not distinguish customer-provider and peering links, and simply follow the approach taken in WIT evaluation [37], [38].

B. Evaluating Topology Generators

Let $\mathcal{P}$ denote the process (*e.g.*, Orbis, WIT, or the actual Internet) used to generate a set of graphs $\mathcal{G}$ of any size. Further, let G_n be a generated graph of size n nodes from the distribution of graphs from a generator denoted $P(\mathcal{G}|\mathcal{P})$. The set of graph measures (*e.g.*, degree, clustering coefficient) computed over that graph is denoted as a function $\mathcal{M}(\cdot)$. The objectives of topology generators are formally defined as:

1) Given a graph $G_n^\star$ of size n, generate a graph G_n of the same size such that $\mathcal{M}(G_n) \approx \mathcal{M}(G_n^\star)$
2) Given a graph $G_n^\star$ of size n, generate a graph G_m of size m where $m \geq n$ such that $\mathcal{M}(G_m) \approx \mathcal{M}(G_n^\star)$
3) Given an ordered sequence of graphs $G_t^\star$ for $t = 1, 2, ..., m$, generate a corresponding sequence of graphs G_t for $t = 1, 2, ..., m$ such that each G_t is the same size as $G_t^\star$ and $\mathcal{M}(G_t) \approx \mathcal{M}(G_t^\star)$,

where $\mathcal{M}(G_n) \approx \mathcal{M}(G_n^\star)$ represents the fact that the set of measures applied to each graph (and the corresponding distributions) are approximately equal. Objectives 1 and 2 are for Orbis [26] while 3 is for WIT [38]. The first objective is a relaxed version of WIT's objective since it considers a graph at a single time point. The second objective assumes properties of the graph at a smaller size remain unchanged (time-invariant) as the graph grows.

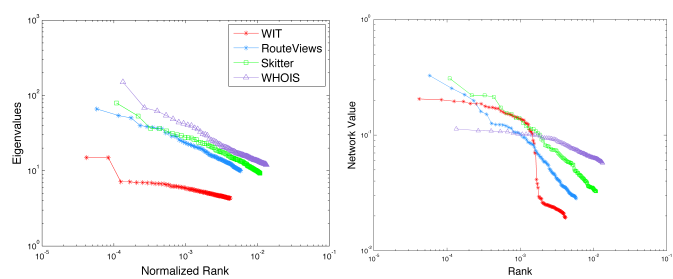
WIT does not estimate the parameters of a dataset and thus can be compared directly with a graph such as RouteViews, Skitter, and WHOIS. For Orbis, we estimate the parameters of each dataset and generate the corresponding topology. To evaluate the rescaling algorithm of Orbis, we use the first snapshot (of RouteViews) and apply the rescaling algorithm to generate a sequence of topologies. This evaluation strategy allows us to evaluate the objectives of each generator separately using the benchmark topologies. We use the optimal parameters for WIT given in [38]; for Orbis, we follow the methodology described in [26] and use the implementation provided by the authors.

V. Evaluation

As a case study, we apply our multi-level framework (graph, node, and link measures) to evaluate each of the two generators Orbis and WIT according to whether it produces graphs that match its advertised claims. Several results have been removed for brevity, but can be found in ref. [32].

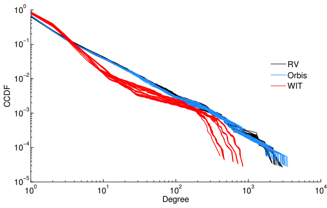
A. Graph Measures

The Orbis topology generator attempts to preserve "any arbitrary" set of measures as the size of the graph increases [26]. However, we find that as the number of nodes increases, the measures from the generated topologies increasingly deviate

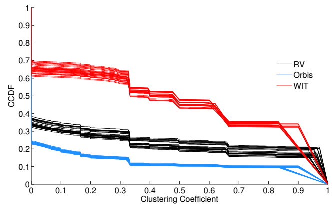


(a) Scree plot (b) Network values

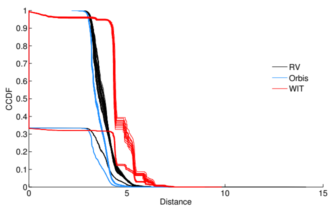
Fig. 3. The WIT topologies are evaluated with three different AS topologies (RouteViews, Skitter, and WHOIS) using two node measures (Scree and top 100 network values). In all cases, WIT is very different. **(a)** WIT's eigenvalues, including the largest, are much smaller than the benchmark topologies. The network values in **(b)** indicate two distinct clusters of nodes which have different structural properties (shown by the backwards S shape). The topologies analyzed above were from March 2004, but similar results are obtained using topologies at different times.



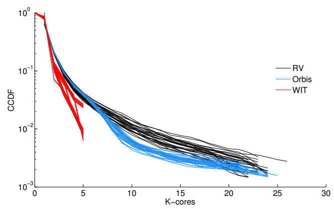
(a) Degree



(b) Clustering Coefficient



(c) Distance



(d) K-cores

Fig. 4. **Node Measures.** Comparison of the generated graphs from WIT and Orbis to RouteViews. Orbis is shown to generate graphs with distributions closer to those of RouteViews.

TABLE III
QUANTITATIVE EVALUATION OF ORBIS USING KS DISTANCE.

	Deg.	CC	Ecc.	Kcores	PR	EigDiff	Net-Value
HOT	0.009	0.000	0.000	0.078	0.067	**0.588**	0.131
RF	0.013	**0.450**	0.000	0.088	**0.215**	**0.629**	**0.680**
WHOIS	0.059	**0.480**	**0.224**	0.060	**0.536**	0.169	0.159
SKITTER	0.010	**0.211**	0.029	0.009	**0.342**	0.096	0.182

from the input topologies. This deviation is largest when applying Orbis on the HOT, WHOIS, and Skitter topologies (results removed for brevity). However, for generating topologies of the same size, we find that Orbis is able to accurately preserve many of the graph measures.

The WIT model was designed and evaluated under the assumption that the clustering coefficient and distance related measures are time-invariant [38], but Fig. 1 indicates that the structure of the Internet is changing. The figure shows that average clustering is decreasing in the last few years, in contrast to previous observations [37], [38]. One possible explanation is that the tier 1-2 ISPs are merging leading to a dense network core, while tier 3-4 ISPs (*e.g.,* content providers) are expanding [14]. These recent changes in the Internet since 2007–2008 [24] impact models such as WIT that directly rely on preserving the distance and clustering coefficient over time. Thus, WIT does not preserve the scalar measures over time, especially for the recent Internet AS data. This result will be seen in Fig. 6.

B. Node Measures

Following the multi-level framework, we analyze the topologies and generators in more detail using node measures. In Fig. 2, we observe a clear transition in the node measures, indicating that the structure of the Internet AS graph during 2004 to 2012 is indeed changing.

Fig. 3 plots the largest network values and singular-values of WIT and compares these with the corresponding benchmark topologies. Both significantly deviate (the values are shifted and are of a different shape), even compared to RouteViews (used in the design and validation of WIT). Preserving the properties of a single graph in time is a relaxation of the WIT objective. However, while WIT reasonably tracks the *graph measures* over time, it is not as good at preserving *node measures* at any given time (Fig. 3).

We also evaluate Orbis using these measures and k-walks (see [32]). Overall, we observe that Orbis preserves the scree plots with reasonable accuracy (for topologies of equivalent sizes), but the rescaled topologies become increasingly different as a function of the number of nodes.

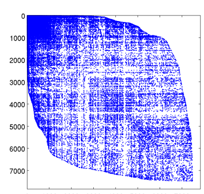
In Table III, we use KS distance (range is from 0 to 1) to measure the difference between the node measures on the original graph and those on the graphs Orbis generated (KL gave similar results). Clearly, Orbis does not preserve many of the node measures from RocketFuel (RF). For the other graphs, Orbis has trouble preserving the clustering coefficients and global measures such as PageRank or EigDiff. As expected, Orbis does preserve the degree distribution.

In Fig. 4, we compare RouteViews to WIT and Orbis. Overall, Orbis better matches the distributions than WIT.

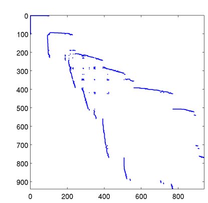
C. Link Measures

For evaluating the link structures of the topologies, we first order the nodes using the principal singular-vector. Ordering the nodes by their distance from the main direction (how well they fit with the most significant connectivity patterns and link structures) as shown in Fig. 5 provides evidence of the formation of clusters or groups (shown as block structures).

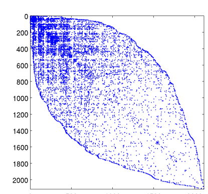
In Fig. 5, we evaluate whether Orbis generates topologies



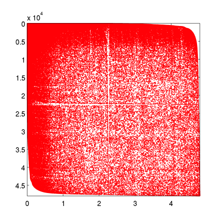
(a) WHOIS



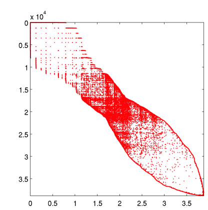
(b) HOT



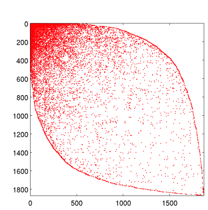
(c) RocketFuel



(d) Orbis (WHOIS)



(e) Orbis (HOT)



(f) Orbis (RocketFuel)

Fig. 5. Evaluating the connectivity and clustering of the generated topologies. The x-axis and y-axis of the above plots represent nodes of the adjacency matrices which are ordered by the principal singular-vector. **(a)**-**(c)** visualize the adjacency matrices for the benchmark topologies. The adjacency matrices for the corresponding generated topologies from Orbis are shown in **(d)**-**(f)**. The clustering properties (block structures) are found to be different in all topologies. For instance, **(c)** clearly shows signs of clusters (blocks/perpendicular lines), but the corresponding Orbis topology in **(f)** has no signs of the clusters previously observed and is much smoother. The main reason is that Orbis randomly rewires the links without considering the *community structure*.

TABLE IV
EVALUATING THE COMMUNITY STRUCTURE OF THE TOPOLOGIES.

		Communities	Q	Nodes	Edges	Degree	CC	Assortativity	C-path	Radius	Diameter
2004	ROUTEVIEWS	*24*	*0.65*	3951	13360	3.38	0.45	-0.56	2.74	3	6
	ORBIS	*46*	*0.48*	957	2254	2.36	0.10	-0.39	3.01	4	8
	WIT	*57*	*0.92*	755	2653	3.51	0.64	-0.25	2.75	4	7
2011	ROUTEVIEWS	*34*	*0.68*	6048	18496	3.06	0.22	-0.42	3.27	5	9
	ORBIS	*60*	*0.48*	2347	5640	2.40	0.12	-0.49	2.91	4	8
	WIT	*66*	*0.94*	2095	11727	5.60	0.45	-0.08	3.44	5	10

(and rescaled topologies) that preserve the link structures, communities and their connectivity. Communities can be loosely defined as sets of vertices with more connections inside the set than outside. We find that Orbis (WIT gives similar results [32]) does not capture the community structure (rewires links without considering communities). Intuitively, there are many graphs that preserve local properties (*e.g.,* degree, clustering), but of these graphs, there are significantly fewer that preserve the underlying communities.

D. Community Measures

In Table IV, we compute the communities of RouteViews in 2004 and again in 2011, and do the same for the generated graphs from Orbis and WIT. We find that WIT and Orbis overestimate the number of communities in RouteViews (for both 2004 and 2011), and this leads to underestimating the size of the largest community (in nodes and edges). Overall, the Q statistic from modularity indicates the degree to which the networks can be subdivided into clearly delineated groups. For this community measure, we find Orbis to be much closer than WIT. We also evaluated the evolutionary objective of WIT and the rescaling of Orbis in [32].

E. A Single Metric

We evaluate the generators quantitatively by measuring the normalized root-mean-square error (NRMSE) of their combined set of measures over time, using the RouteViews data as ground truth. For this case study, we selected fourteen simple measures: average degree, average clustering coefficient, assortativity, average distance, radius, diameter, largest eigenvalue, network conductance, average eigenvector difference, trace, rank, and average number of k-walks for $k = \{2, 3, 4\}$ and learned 26 measures from RouteViews (Section III-F). The combined set of measures over time (selected and learned) is normalized so that each measure is given the same weight. The results are shown and discussed in Fig. 6 and visualized in Fig. 7. From this, we see that the generated and rescaled topologies from Orbis are preserved with reasonable accuracy, whereas WIT is worse at tracking the properties over time.

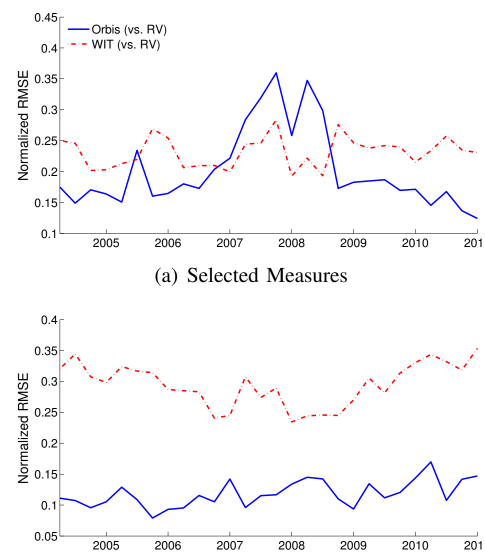
Using both the NRMSE and learning a set of measures automatically allows generators to be evaluated systematically using graph measures that are representative of the true structural features present.

VI. CONCLUSIONS

In this paper, we have proposed a multi-level framework for understanding Internet topologies, and for comparing topology generators. We used the framework to evaluate whether the recent generators Orbis and WIT preserved a wide-range of important network properties and compared their ability to preserve these characteristics as the network evolves. We identified several strengths and shortcomings of both generators. We also observed that recent Internet evolutionary characteristics significantly differ from trends assumed by many Internet topology generators.

Traditionally, topology generators have evaluated evolutionary properties using a few macro measures which often lead to misleading conclusions. For this reason, we used a multi-level approach that leverages both macro measures (graph) and micro measures (node and link measures) to more accurately compare topologies while capturing the important network characteristics warranted by researchers. Our results suggest that existing topology generators fail to accurately model the evolution of the Internet AS topology. More unexpectedly, we found that many generators fail to capture important static characteristics.

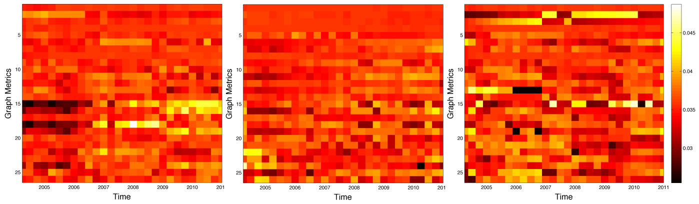
In general, we found data-driven generators, *e.g.,* Orbis, to be more accurate than the generators based on a mechanism with no estimation (such as WIT). Data-driven generators



(a) Selected Measures

(b) Learned Measures

Fig. 6. Topology generators are evaluated over time (2004-2011) by computing the NRMSE of their combined set of measures and the RouteViews data (used as ground-truth). We find that Orbis preserves the properties of the static topology with reasonable accuracy (used the 2004 RouteViews topology as input), but as this topology is rescaled to a larger size, the properties diverge more from the true distributions (shown by the increasing trend in **(a)** and **(b)**). The combined set of measures from the WIT topologies over time do not match the Internet AS (if WIT tracked the properties perfectly, then the WIT curve would be a horizontal line on zero). In all cases, the Orbis topologies are shown to be more similar (diverge less) to RouteViews.



(a) RV (Internet) (b) Orbis (c) WIT

Fig. 7. Comparison of the learned graph measures across time (2004-2011) for each topology generator. The Orbis generator is shown to be more similar to the Internet while WIT appears more random. Interestingly, there is a significant evolutionary transition in the Internet captured by the 15-19th graph measures.

preserve the properties of an input topology, but generate static topologies with low or no variance. However, for modeling the evolution of the Internet, the properties become significantly uncorrelated as the size increases. Conversely, parametric generators cannot model the Internet evolution if any key assumption is violated or the assumed characteristics change [24]. Moreover, if their models lack parameter estimation, then selecting a set of reasonable parameters becomes extremely difficult in practice.

In future work, we plan to use our framework to investigate additional topology generators. We will also develop a parameter estimation technique for WIT and analyze its behavior with the refined parameters. Another direction we plan to pursue is an in-depth study of recent Internet evolution and the causes for the changes we have observed.

References

[1] Internet routing registries. http://www.irr.net/.
[2] Oregon RouteViews project. http://www.routeviews.org/.
[3] R. Albert and A. Barabási. Topology of evolving networks: local events and universality. *Physical review letters*, 85(24):5234–5237, 2000.
[4] B. Donnet et al. Internet topology discovery: A survey. *IEEE Communications Surveys and Tutorials*, 2007.
[5] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science (New York, N.Y.)*, 286(5439):509–512, October 1999.
[6] V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008:P10008, 2008.
[7] A. Blum, T. Chan, and M. Rwebangira. A random-surfer web-graph model. In *ANALCO*, pages 238–246, 2006.
[8] T. Bu and D. Towsley. On distinguishing between Internet power law topology generators. In *IEEE INFOCOM*, pages 638–647, 2002.
[9] CAIDA. Skitter. http://caida.org/tools/measurement/skitter/.
[10] K. Calvert, M. Doar, and E. Zegura. Modeling Internet topology. *IEEE Communications Magazine*, 35(6):160–163, 1997.
[11] H. Chang, S. Jamin, and W. Willinger. Internet connectivity at the AS-level: an optimization-driven modeling approach. In *ACM SIGCOMM MoMeTools Workshop*, pages 33–46, 2003.
[12] F. Chung. *Spectral graph theory*. Number 92. AMS, 1997.
[13] A. Dhamdhere and C. Dovrolis. Ten years in the evolution of the Internet ecosystem. In *ACM SIGCOMM/USENIX IMC*, pages 183–196, 2008.
[14] A. Dhamdhere and C. Dovrolis. The Internet is flat: modeling the transition from a transit hierarchy to a peering mesh. In *Proc. of CoNEXT*, page 21, 2010.
[15] A. Fabrikant, E. Koutsoupias, and C. H. Papadimitriou. Heuristically optimized trade-offs: A new paradigm for power laws in the Internet. In *ICALP*, pages 110–122, 2002.
[16] L. Gao. On inferring autonomous system relationships in the Internet. *IEEE/ACM Trans. Netw.*, 9:733–745, December 2001.
[17] C. Gkantsidis, M. Mihail, and E. Zegura. Spectral analysis of Internet topologies. In *IEEE INFOCOM*, pages 364–374, 2003.
[18] R. Govindan and A. Reddy. An analysis of Internet inter-domain topology and route stability. In *IEEE INFOCOM*, pages 850–857, 2002.
[19] E. Gregori, L. Lenzini, and C. Orsini. k-dense communities in the internet as-level topology. In *COMSNETS*, pages 1–10, 2011.
[20] K. Henderson, B. Gallagher, L. Li, L. Akoglu, T. Eliassi-Rad, H. Tong, and C. Faloutsos. Its who you know: Graph mining using recursive structural features. In *SIGKDD*, 2011.
[21] C. Jin, Q. Chen, and S. Jamin. Inet: Internet topology generator. *Technical Report CSE-TR-433-00, University of Michigan*, 2000.
[22] S. Jin and A. Bestavros. Small-world characteristics of Internet topologies and multicast scaling. In *Proc. of IEEE/ACM MASCOTS*, 2003.
[23] C. Labovitz, A. Ahuja, R. Wattenhofer, and S. Venkatachary. The impact of Internet policy and topology on delayed routing convergence. In *IEEE INFOCOM*, pages 537–546, 2001.
[24] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. In *Proceedings of ACM SIGCOMM*, pages 75–86, 2010.
[25] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the Internet's router-level topology. *ACM SIGCOMM Comp. Comm. Rev.*, 34(4):3–14, 2004.
[26] P. Mahadevan, C. Hubble, D. Krioukov, B. Huffaker, and A. Vahdat. Orbis: rescaling degree correlations to generate annotated Internet topologies. In *ACM SIGCOMM*, pages 325–336, 2007.
[27] P. Mahadevan, D. Krioukov, M. Fomenkov, X. Dimitropoulos, et al. The Internet AS-level topology: three data sources and one definitive metric. *ACM SIGCOMM Comp. Comm. Rev.*, 36(1):17–26, 2006.
[28] A. Medina, A. Lakhina, I. Matta, and J. W. Byers. Brite: An approach to universal topology generation. In *MASCOTS*, pages 346–353, 2001.
[29] M. E. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2):026118, 2001.
[30] 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, USC, Feb 2000.
[31] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *Arxiv preprint cs/0209028*, 2002.
[32] R. Rossi, S. Fahmy, and N. Talukder. A multi-level approach for evaluating topology generators and their evolution. http://www.cs.purdue.edu/homes/fahmy/reports/topology.pdf, 2011.
[33] M. Roughan, W. Willinger, O. Maennel, D. Perouli, and R. Bush. 10 lessons from 10 years of measuring and modeling the Internet's autonomous systems. *IEEE JSAC, Special Issue on "Measurement of Internet Topologies"*, 29(9):1810–1821, October 2011.
[34] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with rocketfuel. In *ACM SIGCOMM*, volume 32, pages 133–145, 2002.
[35] H. Tangmunarunkit, R. Govindan, S. Jamin, S. Shenker, and W. Willinger. Network topology generators: degree-based vs. structural. In *ACM SIGCOMM*, pages 147–159, 2002.
[36] J. Wang, Y. Liu, and Y. Jiao. Modeling generation of the router-level topology of an ISP network. *Computing*, pages 1–16, 2010.
[37] X. Wang, X. Liu, and D. Loguinov. Modeling the evolution of degree correlation in scale-free topology generators. In *IEEE INFOCOM*, pages 1094–1102, 2007.
[38] X. Wang and D. Loguinov. Understanding and modeling the Internet topology: economics and evolution perspective. *IEEE/ACM Trans. Netw.*, 18(1):257–270, 2010.
[39] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.
[40] S. Zhou and R. J. Mondragón. Accurately modeling the Internet topology. *CoRR*, 2004.