Improving the Accuracy and Usefulness of Synthetic AS-Level Topology Models

Pedro Rodrigues
CITI and Departamento de Informática
Faculdade de Ciências e Tecnologia, FCT
Universidade Nova de Lisboa
2829-516 Caparica, Portugal

José Legatheaux Martins
CITI and Departamento de Informática
Faculdade de Ciências e Tecnologia, FCT
Universidade Nova de Lisboa
2829-516 Caparica, Portugal

***Abstract*—Scientific experiments requiring Internet models rely on topology generators to build network graphs. Good quality topology generators are available. Unfortunately, annotation of graphs nodes and edges with attributes (*e.g.*, capacities, latencies, ...) is an issue for which well founded methodologies are still lacking.**

This paper deals with the annotation of synthetic AS-level Internet graphs problem, and proposes a node layer classification methodology that supports annotation strategies with increased accuracy and usefulness. The proposal is based on well known invariant properties of the Internet, a subject also revisited in the paper, and recent theoretical results on the properties of large-scale graphs. An Internet AS-level graph made available by CAIDA is used as an anchor starting point.

***Index Terms*—AS Relationships, Internet Topology, Graph Annotations, Simulation, Modeling**

I. Introduction

Many Internet related studies are based on simulation or emulation, since testing new proposals in the Internet is often too costly, complicated, impractical, or even impossible. In a classical paper [1], Floyd and Paxson put forward not only the fundamental role of modelling and simulation in Internet research, but also how this exceedingly complex human creation is hard to model. Since they wrote their paper, Internet grew several orders of magnitude and, as they also predicted, besides mail and HTTP, new "killer applications" have emerged (P2P, video, ...) and traffic patterns continue to evolve. Therefore, these difficulties are far from being completely overcome. To deal with this state of affairs they stressed, in the same paper, the importance of *Internet invariants*, *i.e.*, properties that persist despite Internet evolution.

Around 10 years ago, empirical studies found that ASes and their interconnections form a graph whose nodes degrees follow a power law distribution [2]. Based on this finding, several Internet topology generators (e.g INET [3], BRITE [4], ...) can be used to build network models of the required size. More recently, some graph theoretical results (*e.g.*, [5]) on very large-scale graphs, with properties similar to the ones found in the Internet, are also highlighting other properties of the Internet topological structure, or can be used to generate graphs with N nodes, whose topological properties resemble the ones measured in a given input graph (*e.g.*, [6]). Using this last result, a graph with N nodes, with the same topological properties as the ones measured in the Internet AS-level graph, can be generated by *"rescaling"* [7].

In order to use a graph to build an useful network model, nodes and edges must be annotated with routers and links properties (e.g. queue sizes, capacities, latencies, packet loss ratios, ...). To our best knowledge, annotation practices often use very simple (or even ad hoc) heuristics and the resulting network model is often not calibrated at all.

The purpose of this paper is twofold. First, guided by the published literature and empirical observations, we revisit the *Internet invariants* subject. Then, guided by these invariants, we try to devise simple but meaningful heuristics, allowing the annotation of AS-level topologies generated by rescaling a recently established Internet AS-level graph, made available by CAIDA [8]. The aim being to improve the accuracy and usefulness of the available synthetic AS-level topology models so produced.

This paper is structured as follows. Section II reviews some of the most relevant invariants of the Internet topology. Section III reviews some data collection projects and tools used to study this topology. Section IV describes the ***dK-series*** graph model, which is the base of the above mentioned rescaling techniques, and presents the topology generator Orbis. Sections V and VI details our efforts to devise simple ways to meaningfully annotate AS Graphs. Section VII presents a tool to annotate AS Graphs with presented approaches, section VIII discusses related work and section IX ends the paper with some conclusions and presents future work.

II. Internet Topology Invariants

Empirical data analysis, model development and experimentation, allow the extrapolation of some important invariants of the Internet AS-level topology.

A. Hierarchical Structure

It is well known that Internet ASes form a hierarchical structure. Moreover, size and business relations among ASes can be used to classify them: `T1`, big ISPs that only peer or provide service to customers, `T2`, regional or intermediate ISPs that have relations with providers, customers and peers, and some other classes representing companies and universities, NICs, exchange points, and content provider networks (CDNs).

B. Power Law Degree Distribution and the Presence of a Core

Empirical analysis of the available AS data has shown that the ASes' degrees approximately follow a power law distribution [2] with a long tail of ASes with a very high degree[1]. The tail is the most relevant part since it encompasses the core ASes in the hierarchy.

In paper [5] it is shown that in graphs of N nodes, with a power law distribution of node degrees, there is the spontaneous emergence of a *core network* formed by the subgraph of nodes having degree $\geq t$ such that $t \in [N^{\varepsilon(N)}, \sqrt{N}\,]$, as detailed on section V. This core network possesses the property of *"soft hierarchy"*, responsible for the *robustness* (it ensures that nodes can still communicate even if a significant part of the core is lost) and the *super scalability* of the network (the diameter is in the order of $\log\log N$).

C. Constant or Slowly Growing Average Path Lenght

Empirical data concerning BGP AS paths shows that the length of these paths stayed relatively constant in the last decade [9]. This empirical result is in line with the super scalability ($\log\log$) property introduced above.

D. The Power of Geography and the Speed of Light

Internet topology reveals the presence of clusters of geographic concentration of network activity, which closely match population, wealth and the presence of big cities acting as main Internet hubs. These geographic clusters are mainly formed by `T2` ISPs and customer ASes since most `T1` ISPs have world wide reach. Studies on the geolocation of IP prefixes [10] also highlight this trend. Most prefixes are geographically constrained, but there are a number of prefixes that are not.

Nowadays, a congestion free core network would have end to end transit time dominated by the speed of light *i.e.*, latency become a dominant issue. This issue is intuitively confirmed by systems of synthetic network coordinates like [11] that use the dimensions needed to take in consideration the core, where geography and latency match, as well as a dimension where local loops and regional asymmetries are accounted for.

III. Data Collection

The construction of Internet models relies on data collected through BGP routing data, data collected from Internet registries (IRRs [12]) and path characteristics as measured by traceroute or other latency probes. Skitter [13], Archipelago [14] and RouteViews [15] are examples of projects whose tools continually collect that type of data.

Skitter's [16] goal was to compute the RTT of IP paths from a set of origin / destinations IP pairs, using different traceroute methods. Due to network dynamism and size, the probing methods did not manage to do a complete mapping of the Internet internal structure. Archipelago [17] is a next generation CAIDA's infra-structure with improvements in flexibility and coordination of probing tasks.

[1]A power law distribution of the generic degree d is $P(d \geq k) = k^{-\alpha}$. In the Internet $\alpha \in [1, 2]$.

The main goal of the RouteViews project is collecting and storing the union of BGP tables of a considerable set of ISPs. Together with the data collected by IRRs, most active ASes are identified and annotated with attributes such as the number and type of links (peer-to-peer and customer-to-provider), the number of IP prefixes and the size of the address space announced, ...

A graph of 20305 ASes computed from the above collected data, automatically annotated by a machine learning algorithm [18] with AS type information, is made available by CAIDA [19] and will be extensively used in our study. The AS types considered are: `T1` - Large ISPs (large backbone providers, mostly Tier-1 ISPs); `T2` - Small ISPs (regional providers, mostly Tier-2 ISPs); `EDU_COMP` - Customer ASes (mostly universities and companies); `IXP` - Internet exchange points; `NIC` - Network information centrs; `ABSTAINED` - Nodes not classified by the algorithm. Table I shows the class distribution found in this graph (designated as graph Ω from now on in this paper).

TABLE I
ASes per class from graph Ω

Class	Size	%
`T1`	44	0.2
`T2`	5599	27.6
`EDU_COMP`	12606	62.1
`IXP`	33	0.2
`NIC`	332	1.6
`ABSTAINED`	1691	8.3

IV. Internet AS-Level Models

Node degree average and distributions have been used as starting points to model the Internet [2]. Recently, the characterisation of the interconnectivity of neighborhoods of increasing size has been shown as being able to reproduce arbitrary interconnection metrics, the so called dK-Series model [6].

This model relies on probability distributions evolving relations of different degrees *d*, between nodes of a given original graph $\mathcal{G}$. As the value of *d* increases, more properties of graph $\mathcal{G}$ are captured relying on more complex distributions. On the limit, the most complex representation is exactly the original graph $\mathcal{G}$.

0K relation characterises a given graph by the average degree $\bar{k}$ of its nodes, with $\bar{k} = 2m/n$, being m the number of edges and n the number of nodes. With this characterization it is not possible to deduce the number of nodes $n(k)$ with degree k. ***1K relation*** includes that information, represented by the node degrees distribution $P(k) = n(k)/n$, *i.e.* the probability of having a node with k degree. Nonetheless allowing a richer characterisation than the previous one, it does not contain information about node's connectivity, such as the number of links between k and k' degrees - $m(k, k')$. ***2K relation*** comprises node's connectivity represented by the *Joint Degree Distribution* (JDD), which is defined by $P(k_1, k_2) = m(k_1, k_2)\mu(k_1, k_2)/(2m)$ where the value of $\mu(k_1, k_2)$ is 2 if $k_1 = k_2$, otherwise 1. The value of $P(k_1, k_2)$

represents the probability of having links between nodes with k_1 degree and nodes with k_2 degree. And so on. For most practical Internet modelling purposes, using *2K relations* seems sufficient.

Topology generators are used to generate graphs of size N with properties similar to the ones empirically found on the Internet. Some generators like INET [3] and BRITE [4] rely on the *1K relation* using power law degree distributions to model the AS-level Internet graph. Orbis [7] is a recent graph generation and rescaling tool relying on the *dK-Series* model. Therefore, generated topologies maintain most of the known topological properties of the original graph used as model.

From the graph Ω, approximate graphs of size N that preserve its main interconnection characteristics, can be generated, suitable to a specific simulation context, using the `scaleTopology` tool from Orbis.

V. Annotation Approaches

Given a (rescaled) AS graph, we need suitable heuristics to annotate it, based on interconnection metrics, since the `scaleTopology` tool output is a list of nodes and edges. Next we present two such annotation approaches and the conclusions we got from their usage.

A. AS Class or Type Based Approach

This approach explores relations between ASes, mapping connectivity metrics to each one of a set of classes that represent a role in the network. The following tables present the calculated values for the nodes of graph Ω, pertaining to a given class[2], and for its neighbours, respectively, which include mininium degree (Min.), maximum degree (Max.), average degree (Avg.) and standard deviation (Std.Dev.).

TABLE II
Nodes' degree metrics computed from graph Ω

Class	Size	Min.	Max.	Avg.	Std.Dev.
T1	44	115	2398	489.545	501.128
T2	5599	1	477	5.898	14.563
EDU_COMP	12606	1	29	1.909	0.972
IXP	33	1	40	7.121	9.018
NIC	332	1	140	3.036	8.740

TABLE III
Nodes' neighbours degree metrics computed from graph Ω

Class	Min.	Max.	Avg.	Std.Dev.
T1	1	2398	34.670	162.568
T2	1	2398	210.480	467.905
EDU_COMP	1	2398	582.986	785.414
IXP	1	1985	163.583	291.159
NIC	1	2398	100.818	242.403

An iterative classification process was used to classify nodes from a (rescaled) AS Graph, from the graph Ω, using for each class the identified connectivity metrics and the distribution of neighbours per class. Using a cross validation method with graph Ω, the iterative classification process did not produce a classification with low error, mainly due to the closeness of connectivity metrics for `T2` and `EDU_COMP`, as these two classes are the most representative classes in the graph Ω, representing 89.7%.

Notwithstanding the fact that this approach does not produce a classification with low error, there are some conclusions that can be made from the presented analysis:

- `T1` nodes are clearly differentiated from the others;
- most of `T2` nodes represent small region providers that have small degrees, consequently this approach tends to classify them as `EDU` or `COMPANY` (and vice-versa);
- some `T2` nodes are classified as T1 since they have high degrees, greater than some T1 degrees;
- some `COMPANIES`, such as Content Distribution Networks, may play an important role on the network, since they have a high degree;
- without other metrics, that can not be inferred from a graph generated by Orbis, it is not possible to distinguish, with low error, the majority of considered classes.

B. Layering or Core Based Approach

Being $\mathcal{G}$ a graph with N nodes whose degrees follow a power law distribution, a mathematical definition of $\mathcal{G}$'s core is presented in paper [5] as being the set of nodes whose degree is $> \tau$, with $\tau = N^{\varepsilon(N)}$. Moreover, nodes with degree $> \sqrt{N}$ form a clique, or are closely to form a clique, which we identify as the core's kernel. Thus, core can be structured as two tiers: *tier1* composed of nodes with degree $> \sqrt{N}$ forming core's kernel; *tier2* composed of nodes with degree $\in]N^{\varepsilon(N)}; \sqrt{N}]$.

In paper [5], the value of $\varepsilon(N)$ is defined as having to satisfy the following three conditions as $N \to \infty$:

$$\varepsilon(N) \to 0,\ N^{\varepsilon(N)} \to \infty,\ \frac{N^{\varepsilon(N)}}{loglogN} \to 0.$$

By the above conditions, $\varepsilon(N)$ value is slightly larger than $1/logN$. The authors of paper [5] do not define a concrete value for $\varepsilon(N)$. In order to define the value of $\varepsilon(N)$ independently of the value of N, we formalise $\varepsilon(N)$ as:

$$\varepsilon(N) = \frac{1}{logN} + \gamma \times N.$$

A shortest path heuristic was used to determine and validate a concrete value for γ. Nonetheless the fact that routing between ASes does not rely on shortest paths but on commercial relationships, in what concerns connectivity, *betweeness* [3] is a good metric to validate the chosen value for γ.

The validation process is as follows: for each node of graph Ω it was computed its betweeness value, using Dijkstra algorithm; by an iterative process, ranging γ in $]0; 9.75 \times 10^{-6}[$, the concrete value of γ was obtained according to the following conditions:

- considering ζ as the set of nodes pertaining to the resulting core for a specific value of γ, with size n;

[2]Class `ABSTAINED` is not considered since it will not be used to classify ASes.

[3]Betweeness is the number of shortest paths that pass through a given node.

- considering β as the set of n nodes with the highest values of betweeness;
- considering μ as the size of $\zeta \cap \beta$;
- maximise $\phi = \mu/n$.

The value of γ was ranged to an upper bound of 9.75×10^{-6} in order to abstain cores too closer to core's kernel. The ϕ maximum value was approximately 0.84, corresponding to $\gamma = 5 \times 10^{-6}$, consequently $\varepsilon(N) \simeq 0.33$ and $\tau \simeq 27.37$. Whereas, considering a core composed of nodes with degree $\geq \sqrt{N} \Rightarrow \gamma \simeq 1.32 \times 10^{-5}$ *i.e.*, core's kernel, a higher ϕ value was achieved (0.87). In fact, as the threshold increases it is possible to define very small cores, *e.g.* $\gamma = 2.1 \times 10^{-5}$ corresponding to a core with 6 `T1` nodes, in which all nodes are the n nodes with the highest betweeness values, thus achieving $\phi = 1$. Hence, limiting the value of γ is essential to obtain a reasonable core, otherwise with the above conditions a core with few nodes would be selected.

Table IV presents core tier1 and tier2 of the graph Ω, conforming with presented core's definitions.

TABLE IV
ASES PER CLASS PERTAINING TO THE CORE TIER1 (LEFT)
ASES PER CLASS PERTAINING TO THE CORE TIER2 (RIGHT)

Class	Count	Class	Count
T1	39	T1	5
T2	5	T2	207
EDU_COMP	0	EDU_COMP	2
IXP	0	IXP	2
NIC	0	NIC	5

These results confirm some of the conclusions made on the previous subsection, such as:

- `T1` nodes are rather different from other nodes as most of them are part of core's kernel;
- Some of the `T2` nodes were classified as `T1` as a result of being part of core's kernel;
- `EDU_COMP` nodes, like CDN, play a relevant part in the AS structure as they are one of core's members;

C. Periphery Tier Identification

For the non-core nodes, a common used approach consists in identifying as periphery ASes (stub-ASes), nodes whose degree is ≤ 2.

TABLE V
ASES PER CLASS WITH DEGREE ≤ 2, FROM THE GRAPH Ω

Class	Count	%
T1	0	0
T2	3046	54.4
EDU_COMP	10919	86.6
IXP	13	39.3
NIC	247	74.3

Since `EDU_COMP` do not provide service to others, the majority of those ASes should be identified as stub-ASes. We identify as periphery ASes, nodes whose degree is ≤ 3, in order to include more `EDU_COMP` ASes, as it can be seen in the following table.

TABLE VI
ASES PER CLASS WITH DEGREE ≤ 3, FROM THE GRAPH Ω

Class	Count	%
T1	0	0
T2	3575	63.9
EDU_COMP	12132	96.2
IXP	16	48.4
NIC	287	86.4

D. Validation

Using the presented layering approach, a graph that preserves the characteristics of graph Ω can be structured in four tiers: core tier1, core tier2, intermediate tier and periphery tier. In order to validate this approach, we computed 20 graphs from graph Ω, using `scaleTopology` tool from Orbis, 10 with about 10000 nodes and 10 with about 5000 nodes[4].

In the following tables we present the node distribution per tier for the graph Ω and the average of node distributions for rescaled graphs with about 10000 and 5000 nodes.

TABLE VII
ASES PER TIER FROM THE GRAPH Ω

Tier	%
Kernel	0.2
Core	1.1
Intermediate	12.8
Periphery	85.9

TABLE VIII
ASES PER TIER FROM RESCALED GRAPHS WITH ABOUT 10000 NODES (LEFT) ASES PER TIER FROM RESCALED GRAPHS WITH ABOUT 5000 NODES (RIGHT)

Tier	%	Tier	%
Kernel	0.4	Kernel	0.6
Core	2.3	Core	2.8
Intermediate	12.9	Intermediate	12.6
Periphery	84.4	Periphery	84.0

The node distributions per tier are very similar. Therefore, applying the presented layering approach on rescaled graphs preserve the hierarchical structure invariant of the original graph Ω. Accordingly, this approach can be used to structure AS level graphs, rescaled from the graph Ω, in four layers (core tier1, core tier2, intermediate tier and periphery tier), which induce a classification of ASes based on the layer that each AS pertains.

Additionally, from the graphs mentioned above we computed the distribution of link type: links within the same layer are considered as peer-to-peer (`p-p`), whereas links connecting different layers are considered as client-provider (`c-p` and `p-c`). In the following tables we present the distribution of link type per tier, as a percentage of all links from a tier[5].

[4]It is not known the factor to which a given graph can be rescaled and still preserve its main original characteristics.

[5]We merged the two tiers of the core in one for a more adequate analysis.

TABLE IX
PEERING, PROVIDER-CUSTOMER AND CUSTOMER-PROVIDER LINKS PER TIER FROM THE GRAPH Ω

Tier	p-p (%)	p-c (%)	c-p (%)
Core	15.4	84.6	-
Intermediate	16.6	38.1	45.3
Periphery	6.2	-	93.8

TABLE X
AVERAGE PEERING, PROVIDER-CUSTOMER AND CUSTOMER-PROVIDER LINKS PER TIER FROM RESCALED GRAPHS WITH ABOUT 10000 NODES

Tier	p-p (%)	p-c (%)	c-p (%)
Core	25.1	74.9	-
Intermediate	15.6	39.2	45.2
Periphery	10.3	-	89.7

TABLE XI
AVERAGE PEERING, PROVIDER-CUSTOMER AND CUSTOMER-PROVIDER LINKS PER TIER FROM RESCALED GRAPHS WITH ABOUT 5000 NODES

Tier	p-p (%)	p-c (%)	c-p (%)
Core	20.9	78.9	-
Intermediate	13.9	36.9	49.2
Periphery	10.1	-	89.9

Rescaled graphs, from the graph Ω, approximately preserve the distribution of `p-p`, `c-p` and `p-c` links per tier of the graph Ω. Consequently, a classification of links based on connectivity between the identified layers can be used to define link annotations.

VI. MODELS SUPPORTED BY THE LAYERING APPROACH

The layering approach defined in the previous section can be used to introduce the following models.

1) Link Capacity Model: A pragmatic capacity model relying on the identified layers can be defined as follows: core tier1, core tier2, intermediate and periphery intra tier links with, respectively, Υ, Ψ, Φ and ω of capacity; links between the two core's tiers having φ capacity; links between core tier 1 and intermediate nodes with θ capacity; links between core tier 1 and periphery nodes with ϑ capacity; links between core tier 2 and intermediate nodes with δ capacity; links between core tier 2 and periphery nodes with η capacity; links between intermediate and periphery nodes with ρ capacity. The concrete values are conditioned by simulation characteristics.

For instance, in what concerns simulation of congestion control or queue management, it is necessary to assign low capacity values, in the order of hundreds kbps, due to the limitation of simulators to generate sufficient traffic to cause congestion. Therefore, by providing the distribution of capacities (Υ, Ψ, Φ, ω, $\varphi, \theta, \vartheta, \delta, \eta, \rho$), one can adapt the capacity model to a specific simulation purpose.

2) End-node Attachment Model: Depending on the simulation or emulation purpose, we distinguish two situations: a situation where application end-nodes are only attached to non-core ASes and a situation where application end-nodes are also attached to core ASes. In what concerns the former, application end-nodes can be randomly assigned to ASes which pertain to the periphery layer. Since the domain of some applications is not limited to one of the considered layers, for the second situation identified, one can choose how many nodes are attached to each layer. Therefore, the attachment model can be adapted to specific simulation environment characteristics.

3) Link Type Model: The classification of client-provider, provider-client and peer-to-peer, as identified on section V-D, is applied to the links of a layered AS graph. Additionally, links between application end-nodes and periphery ASes are classified as stub links.

VII. IMPLEMENTED TOOL

We developed a tool to annotate rescaled AS graphs, that preserve graph Ω properties, which uses the previous defined models. The annotation process can be described as follows:

Input:

- size of the rescaled graph (n);
- distribution of capacity of intra-tier and inter-tier links;
- number of application end-nodes to attach on each layer and the capacity of attachment links (optional).

Process:

- Rescale graph Ω to a graph with size n using `scaleTopology` tool from Orbis;
- Distribute nodes to each layer as defined in the layering approach;
- Attach application end-nodes to each layer according to input distribution (if specified);
- Annotate nodes with layer information and links with capacity values conforming to distribution of link capacities;

Output:

- List of n nodes annotated with layer type (coreTier1, coreTier2, intermediate, periphery and application) along with an ID;
- List of links with IDs of connected nodes and annotated with link type information;
- List of link types with properties of each type.

The resulting graphs can be used as input to simulators or emulators (*e.g.*, Modelnet) in studies regarding to Internet, improving simulation/emulation results towards graphs annotated with common ad-hoc heuristics. Next we present a summarised example produced by the tool:

```
Nodes :
     nodeID = 0 nodeType = periphery
(...)

Links :
     src = 10366 dst = 10806 linkType = coreTier1-coreTier1
(...)

Link Type Configuration :
     linkType = coreTier1-coreTier1 capacity = 125 Mbps
(...)
```

VIII. RELATED WORK

Tools to generate an annotate AS-level Internet topology are currently available. For example, the GHITLE [20] topology generator creates a small number of `T1` ASes, a certain number

of other intermediate tier ASes, and several stub ASes. Peering edges among ASes of the same level, as well as provider-customer edges among ASes of different levels, are then added. Distributions of ASes and links are user provided. This style of approach is quite common (e.g. [9]). It is also frequent to classify ASes in just two layers, stub and transit tiers, where stub ASes are randomly chosen or chosen as the ASes with lower degree.

With our approach, nodes and edges are generated by soundly rescaling graph Ω with the Orbis tools, and the identification of the proposed four layers is based on tested properties of the original graph, that essentially still hold in rescaled graphs.

Authors of paper [21] describe an approach to generate Internet AS-level topologies with business relations annotations using a generalisation of the rescaling techniques behind the Orbis generator [7]. Likely, their proposal could also be used to generate the extra annotations required in many simulation experiments (capacities, latencies, ...) provided these were publicly available in the original AS databases. This is not the case since most providers do not disclose them and monitoring all the links among the different ASes is an insurmountable task.

IX. Conclusions and Future Work

Current state-of-the-art to build Internet models rely on topology generators able to produce graphs of different sizes, that mimic quite reasonably most interconnection metrics found in real-world Internet graphs. This is specially true for AS-level Internet graphs since the empirical data made available by CAIDA gives access to a very reasonably accurate AS graph. Although these tools can be used to generate rescaled graphs from an original graph, they generally do not produce annotated graphs with properties like link and node types, capacities or latencies. Producing them is a major requirement in the current scientific practice related to networking and distributed systems studies. However, current state-of-the-art concerning Internet model graphs annotation, is generally performed using intuition-based (or even ad hoc) heuristics.

In this paper we have introduced annotation strategies, suitable for each experiment's specific needs, built on an AS classification by tiers: core tier 1 or tier 2, which union forms the core tier, intermediate and periphery tiers. As topology properties drive our nodes' tier determination, they can be used in the original graph, as well as in the rescaled graphs.

The classification methodology is based on heuristics derived from several invariant Internet properties and theoretical results concerning large-scale graphs, with properties similar to the Internet, which have node degree power law distributions, and possess a core. For the validation of the proposed approach, we performed several experiments and analysis of an original AS graph, provided by CAIDA, as well as of several rescaled graphs built using it as model.

The main contributions of this paper are a fresher view of the Internet invariants issue, as well as a new improved layering approach, to annotate AS Graphs. Although the presented strategy results on richer and more realistic AS Graphs, as most available annotation strategies, it does not take into consideration geographic issues to drive latency annotation. Since most ASes are geographically clustered, at least intra and inter geography clusters links latencies should be differentiated. One possible approach we intend to pursue is to explore the relations of topology and geography to improve latency annotations.

Acknowledgment

The authors want to thank the anonymous reviewers by their helpful comments.

References

[1] S. Floyd and V. Paxson, "Difficulties in simulating the internet," *IEEE/ACM Transactions on Networking*, vol. 9, pp. 392–403, 2001.

[2] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *In SIGCOMM*, 1999, pp. 251–262.

[3] C. Jin, C. J. Qian, and S. Jamin, "Inet: Internet topology generator," 2000.

[4] A. Medina, A. Lakhina, I. Matta, and J. Byers, "Brite: An approach to universal topology generation," *Proceedings of MASCOTS*, vol. 1, 2001.

[5] I. Norros and H. Reittu, "Network models with a 'soft hierarchy': a random graph construction with loglog scalability," *Network, IEEE*, vol. 22, no. 2, pp. 40–46, 2008. [Online]. Available: http://dx.doi.org/10.1109/MNET.2008.4476070

[6] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat, "Systematic topology analysis and generation using degree correlations," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 4, pp. 135–146, 2006.

[7] P. Mahadevan, C. Hubble, D. Krioukov, B. Huffaker, and A. Vahdat, "Orbis: rescaling degree correlations to generate annotated internet topologies," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 325–336, 2007.

[8] CAIDA, "The cooperative association for internet data analysis," http://www.caida.org.

[9] A. Elmokashfi, A. Kvalbein, and C. Dovrolis, "On the scalability of bgp: the roles of topology growth and update rate-limiting," in *CoNext 2008*, ACM, Ed. ACM, 2008.

[10] M. Freedman, M. Vutukuru, N. Feamster, and H. Balakrishnan, "Geographic locality of ip prefixes," *Proceedings of the Internet Measurement Conference 2005 on Internet Measurement Conference table of contents*, pp. 13–13, 2005.

[11] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: a decentralized network coordinate system," in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM, 2004, pp. 15–26.

[12] I. R. Registries, http://www.irr.net.

[13] CAIDA, "Skitter," http://www.caida.org/tools/measurement/skitter/.

[14] ——, "Archipelago measurement infrastructure," http://www.caida.org/projects/ark/.

[15] U. of Oregon, "Route views project," http://www.routeviews.org.

[16] B. Huffak, D. Plummer, D. Moore, and K. Claffy, "Topology discovery by active probing," in *SAINT-W '02: Proceedings of the 2002 Symposium on Applications and the Internet (SAINT) Workshops*. Washington, DC, USA: IEEE Computer Society, 2002, p. 90.

[17] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, "Internet mapping: from art to science," CAIDA.

[18] X. Dimitropoulos, D. Krioukov, G. Riley, and K. Claffy, "Revealing the autonomous system taxonomy: The machine learning approach," in *In Passive and Active Measurement (PAM) Workshop*, 2006.

[19] X. Dimitropoulos, "Autonomous system taxonomy," http://www.caida.org/data/active/as_taxonomy/.

[20] C. de Launois, "Ghitler," http://ghitle.info.ucl.ac.be/.

[21] X. A. Dimitropoulos, D. V. Krioukov, A. Vahdat, and G. F. Riley, "Graph annotations in modeling complex network topologies," *CoRR*, vol. abs/0708.3879, p. 28, 2007.