Towards a bipartite graph modeling of the internet topology

Fabien Tarissan [a,*], Bruno Quoitin [b], Pascal Mérindol [c], Benoit Donnet [d], Jean-Jacques Pansiot [c], Matthieu Latapy [a]

[a] Laboratoire d'Informatique de Paris 6 (LIP6), Université Pierre et Marie Curie (UPMC), Centre National de la Recherche Scientifique (CNRS), France
[b] Département d'Informatique, Faculté des Sciences, Université de Mons, Belgium
[c] Laboratoire des sciences de l'Ingénieur, de l'Informatique et de l'Imagerie (ICube), Université de Strasbourg, France
[d] Institut Montefiore, Université de Liège, Belgium

ARTICLE INFO

ABSTRACT

Modeling the properties of the Internet topology aims at generating large scale artificial IP networks that mimic properties of real ones for simulation purposes. Current models typically consider the Internet as a simple graph where edges are point-to-point connections between routers. This approach does not take into account point-to-multipoint connections that exist at lower layers in the network, e.g. layer-2 clouds, such as Ethernet switches or MPLS networks. Instead, such physical point-to-multipoint connections are modeled as several logical IP level point-to-point connections.

In this paper, we rely on recent developments in topology discovery based on IGMP probing that allows for revealing part of the network's layer-2 structure. We take advantage of this additional knowledge for proposing an Internet model based on bipartite graphs considering both point-to-point and point-to-multipoint connections. Our model remains simple: it only takes as input the node degree sequence for both layer-2 and layer-3 nodes, randomly generates a bipartite graph respecting those distributions, and then derives the corresponding layer-3 topology. We show that, despite the simplicity of our model, realistic network properties, such as high local density, emerge naturally. This is in contrast with the now common belief that such properties can only appear with more intricate models or if explicitly injected in random models. Besides, we also provide evidences of how the analysis performed at the bipartite level might shed light on important properties of the real network structure. Finally, we propose and evaluate a bipartite graph generator based on our model that only takes two synthetic node degree distributions as input.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Improving our understanding of the Internet topology structure is extremely important. It has much impact on the ability to provision and manage IP networks and enhance their reliability and efficiency. It also allows for designing effective network protocols matching the specific requirements of a large panel of applications. Assessing the quality of a network or protocol design involves theoretical studies and simulations conducted on artificial graphs obtained from models of the Internet topology. Many efforts have been made in modeling Internet [1,2], from very simple models [3] to more complex ones based on latest developments in Internet topology discovery and modeling [4–6]. However, modeling the Internet remains a challenging task because of its heterogeneity and dynamics [7,8].

* Corresponding author. Address: LIP6, UMPC, 4 place Jussieu, 75005 Paris, France. Tel.: +33 144278810; fax: +33 144277495.
E-mail addresses: fabien.tarissan@lip6.fr (F. Tarissan), bruno.quoitin@umons.ac.be (B. Quoitin), merindol@unistra.fr (P. Mérindol), benoit.donnet@ulg.ac.be (B. Donnet), pansiot@unistra.fr (J.-J. Pansiot), matthieu.latapy@lip6.fr (M. Latapy).

Usually, the Internet is depicted as a simple graph where vertices represent, depending on the Internet topology view, IP interfaces, routers, or autonomous systems (ASes) and edges stand for direct connections between those vertices. In particular, when considering the router level view of the Internet, edges usually represent point-to-point links between routers, i.e., IP hops. However, the Internet is actually made of fundamentally different kinds of nodes at layer-2 (L2), which induce its layer-3 (L3) structure: routers might be connected through L2 devices such as Ethernet switches, IXPs, sub-networks, etc. and a L3 link between two routers goes therefore through a L2 device. Multiple routers connected through a single L2 device will appear as a possibly large clique of separate point-to-point links in a layer-3 view. The underlying point-to-multipoint connections are generally invisible because they are challenging to discover when using ordinary active topology discovery techniques based on traceroute [9].

Being able to map Internet topologies exhibiting the two layers of connection would open new perspectives in Internet modeling and topology generation. Indeed, one could then model the Internet topology as a *bipartite graph*, i.e., a graph in which vertices can be divided into two disjoint sets, ⊤ (e.g., Ethernet switches) and ⊥ (e.g., routers), such that every edge connects a vertex in ⊤ to one in ⊥. Bipartite graphs are a fundamental object in computer science and, as such, are widely studied [10–12]. A key operation over bipartite graphs is the *projection* that transforms the bipartite structure into a simple graph where a link between two routers in ⊥ exists if they are linked to a same L2 device in ⊤. Interestingly, the projection fits exactly the inference of the Internet L3 topology from its L2 topology. This makes bipartite graphs (and their projection) an appealing approach for Internet topology modeling with a L2/L3 view.

Fortunately, a recent advance in Internet topology discovery through IGMP probing [13] has offered an opportunity to better characterize the nature of IP connections (point-to-point or point-to-multipoint). With a single IGMP probe, one can obtain all local multicast interfaces and neighbors of a multicast router, as well as its multicast connections through L2 multi-access networks. This latter feature provides point-to-multipoint connections between L3 devices. Considering a map resulting from IGMP probing [6], we are able to construct a bipartite graph where vertices belong to one of these two types.

It is worth noticing that IGMP data allows one to easily discover the *actual* bipartite shape of the Internet induced by the interactions between L2 and L3 devices. Generally, such a bipartite structure is artificially generated to capture some clustering properties in flat network ground measurements [14]. This is not our approach here since we stick to the *existing* bipartite structure detected by the measurement tool. Interestingly enough, some IP network protocols are aware of such a physical distinction. An example is the OSPF routing protocol [15]. OSPF routers describe the network topology using *Link State Advertisements* (LSAs). Separate LSAs are used to represent different kinds of links: Router LSAs are dedicated to outgoing links of L3 nodes while Network LSAs represent outgoing links of transit L2 networks. IGMP probing offers an abstraction of the same bipartite view as OSPF: when several IP neighbors are seen through the same IP interface, one may deduce the presence of a transit L2 device.

Such a view is necessary to understand the actual (e.g., physical) nature of IP networks. If one wants to determine the actual physical degree of routers [6] or measure the physical resiliency of a network [13], it is mandatory to avoid the confusion between logical and physical connections. For example, a clique between routers appears much more resilient than a star inter-connection. Common graph properties such as the clustering coefficient are also impacted according to the nature of the network view [16]. Further, recent works have provided evidences that real network topologies are far from random and are often due to strong constraints [4]. A bipartite vision of the network would allow for identifying which aspects of the real network might stem from random processes and which ones are due to strong designed patterns.

In this paper, we step into the breach opened by the L2 devices inference and describe the first bipartite model of the Internet topology. Our model has the strong advantage of being "simple", i.e., it is a random-based model that does not require injecting several constraints. As input, we only consider the node degree distribution of both L2 and L3 devices for generating the random bipartite graph and, then, project this structure into a simple graph. To this regard, our model can be seen as an extension of the standard *configuration model* [17–19] using two distinct degree sequences instead of one.

Note that our model does not aim at identifying network construction mechanisms as would do a structural model [20,21], a preferential attachment model [22,23], or an engineer-oriented model [4]. If such approaches may bring interesting knowledge on the networks emergence (although they are often criticized, see [16]), they are usually not well suited for formal analysis and they tend to enforce specific properties in the generated graphs. This is why we rather follow the tradition of random models [3,17,24].

In order to assess the relevance of our model, we perform two different analyses. First, we demonstrate that the resulting projected graphs have behaviors similar to those of actual ones, specially regarding metrics that were not injected in the model, such as the local density or the degree correlations for instance. Indeed, it is worth noticing that standard models usually only reproduce properties they focus on but are unable to cope with all other features. On the contrary, our model is able to capture a range of characteristics that goes beyond the simple degree distribution it relies on. In order to emphasize this point and better evaluate the improvements brought by our bipartite model, we confront our results with random graphs directly generated with the configuration model from which our model derives. It results that, as expected, the configuration model is unable to cope with other properties than the degree distribution. Second, we evaluate different metrics on the bipartite structure itself and give evidences of the relationship between the observed bipartite properties and the projected ones. Our analyses show that, although not perfectly reproducing the real data, our model succeeds in capturing most of its properties
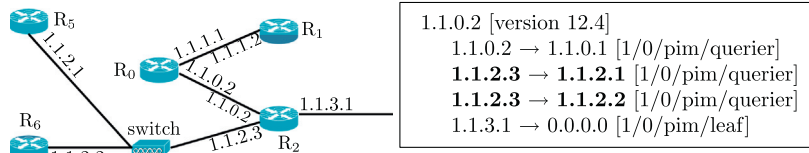
Fig. 1. mrinfo example.

and provides mathematical tools for explaining properties of the L3 structure from the analysis of the bipartite structure.

Analyzing the limitations derived from our first study, we also explore possible extensions to our model. First, we investigate the interest of taking into account the correlation between point-to-point and point-to-multipoint connections of the routers. Second, using statistical metrics defined at the bipartite level, we identify strong redundant patterns and propose a way to cope with such a structural property in the model. Indeed the overlapping between L2–L3 connections is frequent in real IP networks since redundancy is a key feature to increase the reachability between networking devices. Such resiliency patterns cannot be accurately revealed using a L3 view only.

We finally present and evaluate a network topology generator that is based on our bipartite model. The generator goes beyond the model in that it does not rely on prescribed degree sequences but rather random ones. The current version of the generator draws the bipartite nodes degree from two distinct power-law distributions. We evaluate our generator and show that even with synthetic degree sequences, the underlying bipartite model still produces projected graphs exhibiting realistic properties. Our generator is freely available at https://code.google.com/p/py-bipartite/.

The remainder of this paper is organized as follows: Section 2 discusses the background required throughout this paper; Section 3 presents the methodology we followed when modeling the network as a bipartite graph and evaluates our model; Section 4 discusses the limits of our model and investigates which additional constraints could lead to better results; Section 5 describes and evaluates our topology generator; finally, Section 6 concludes this paper and lays some foundations for future works.

2. Background

In this section, we introduce the required background for the remainder of the paper. First, we focus on mrinfo (Section 2.1), a tool allowing for silently revealing all multicast IP addresses of a router, as well as its connections towards other routers and L2 devices. Second, we discuss bipartite graphs (Section 2.2) that are used for modeling topology data collected with mrinfo.

2.1. IGMP probing

mrinfo messages use the Internet Group Management Protocol (IGMP [25]). IGMP was initially designed to allow hosts to report their active multicast groups to a multicast router on their LAN. However, the Distance Vector Multicast Routing Protocol, DVMRP, has defined two special types of IGMP messages that can be used to monitor routers [26]. Although current IPv4 multicast routers do not use DVMRP anymore, they still support these special IGMP messages. Upon reception of an IGMP ASK_NEIGHBORS message, an IPv4 multicast router replies by sending an IGMP NEIGHBORS_REPLY message that lists all its multicast enabled adjacencies. Fig. 1 shows an example of the usage of mrinfo to query the router R_2 (1.1.0.2 is the replying interface of R_2). mrinfo reports that this router is directly connected to R_0 (through interface 1.1.0.1) via a layer-3 (L3) point-to-point link. One can also notice that R_2 is connected to routers R_5 and R_6 through a layer-2 (L2) network (labeled "switch" in Fig. 1) because interface 1.1.2.3 appears twice in the mrinfo reply (see bold text in Fig. 1). Finally, mrinfo reports that interface 1.1.3.1 has no multicast router neighbor (the right IP address is equal to 0.0.0.0). All this topological information is obtained by sending a single IGMP message. mrinfo provides information similar to a show command dedicated to the multicast routing plan.

In the analysis provided in this paper, the inference of *L2 networks* is critical. In our context, by L2 network, we mean a technology allowing a router to transmit IP packets to several other IP routers through the same interface, i.e., a multi-access network. One often distinguishes between *Non Broadcast Multiple Access* (NBMA) networks (e.g., ATM, Frame Relay, X25), and *broadcast networks* (BN) such as most LAN networks (e.g., Ethernet, token ring, FDDI).

These two kinds of networks behave differently as far as IP multicast is concerned. In particular, when using *Protocol Independent Multicast* (PIM) as a routing protocol in a BN, only one of the PIM IP neighbors is elected as the *querier* [27]. Moreover, in common BN such as L2 Ethernet switches, the IP view around the L2 device should exhibit symmetric properties and reveals that IP interfaces involved in this symmetric point-to-multipoint connection are allocated within a tight subnet prefix. In NBMA networks, IP packets are usually transmitted via circuits that behave as a collection of point-to-point or point-to-multipoint connections. Such properties can be easily revealed within the mrinfo range: in this paper, we focus on most common BN such as L2 Ethernet switches. These represent the vast majority of multi-access networks in the mrinfo dataset that we use.

Since May 1st, 2004, we collect the mrinfo data from a host located at the University of Strasbourg, France. In this paper, we consider the data collected until October 31st, 2008. The collection script maintains a list of known multicast routers. Each day, it sends an IGMP query to each of these routers to collect their list of interfaces. If a new router is discovered in a received answer (in the list of
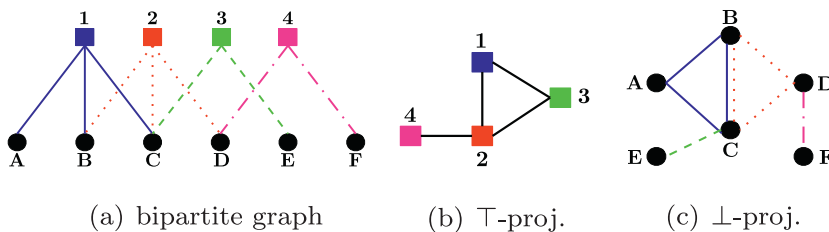
Fig. 2. Example of bipartite graph and its {⊤, ⊥}-projections.

outgoing interfaces of an already known router), this router is also queried in turn. These recursive queries stop at unresponsive routers or when all known routers have been queried. Each router is queried at most once per day. Additional information about the collection script and the raw `mrinfo` dataset may be found in [28].

On average, `mrinfo` was able to daily discover roughly 10,000 different routers while scanning 100,000 interfaces, and 1000 ASBRs belonging to 200 ASes. The largest connected component consists in 7000–8000 routers on average. We remove interfaces with nonpublicly routable IP addresses (see RFC 3330 [29]), loopback address block 127.0.0.0/8 and the 0.0.0.0 address. We also remove all multicast tunnel interfaces to focus on actual physical links. On average, more than 25% of the interfaces collected by `mrinfo` fall within those categories.

Finally, we extracted L2 nodes based on the three following rules [6]:

- *Symmetry rule.* All routers attached to the potential L2 network should have the same view. In Fig. 1, router R_2 is connected to R_5 and R_6 through a L2 device. When probing R_5 and R_6 with `mrinfo`, R_2 must also appear in their `mrinfo` output.
- *Querier rule.* In a normal case, only one router per L2 network must be tagged as the IGMP "querier" (i.e., it won the querier election on the subnet [27]: it has the greatest IP address on the subnet). For instance, in Fig. 1, as interface `1.1.2.3` of R2 is tagged as "querier", interfaces `1.1.2.1` of R_5 and `1.1.2.2` of R_6 should not be tagged as such.
- *Subnet mask rule.* The validity of the minimum mask covering all IP addresses in the subnet is verified.

It is worth noticing that obtaining L2 and L3 topologies is also possible using Gunes and Sarac's subnet inference technique [30]. We believe that the framework provided in this paper can also be applied on Gunes and Sarac's dataset.

2.2. Bipartite graphs

A *bipartite graph* is a triplet $G = (\top, \bot, E)$, where $\top$ is the set of *top* nodes, $\bot$ the set of *bottom* nodes, and $E \subseteq \top \times \bot$ the set of links. Compared to standard graphs, nodes in a bipartite graph are in two disjoint sets, and the links are always between a node in one set and a node in the other set. An example of bipartite graph is given in Fig. 2a, where $\top$ nodes are depicted by squares and $\bot$ nodes by circles.

The ⊥-projection of G is the graph $G_\bot = (\bot, E_\bot)$ where two nodes (of ⊥) are linked together if they have at least one neighbor in common (in ⊤) in G: $E_\bot = \{(u, v), \exists x \in \top: (u, x) \in E$ and $(v, x) \in E\}$. The ⊤-projection is defined dually. Both projections are illustrated in Fig. 2b and c.

Note that bipartite graphs are very close to *hypergraphs* which also model one-to-many relationships by representing hyper-links as sets of nodes instead of using a new kind of nodes. As such, hypergraphs are in complete bijection with bipartite graphs, the only difference being that bipartite graphs allow to explicitly exhibit the hyper-links between the nodes. The directed hypergraphs are good models for one-way one-to-many links such as radio links (see for instance [31,32]) but in our case we are interested in a model of the (wired) core of the Internet where the connections are bidirectional. This is why we do not need directed hypergraphs. Besides, the bipartite model follows more closely the actual physical architecture than an hypergraph model: Firstly, a Layer-2 node (typically, a switch) is a single point of failure as much as a Layer-3 node (router) and it is very interesting to have the same representation of these 2 types of nodes. Secondly, a link connecting a Layer-2 and a Layer-3 node has a physical existence which allows to study the impact of its failure, since it is represented by a single edge in the bipartite graph. This type of link is not directly represented in an hypergraph model. For all those reasons, we decide to rely on the bipartite formalism instead of hypergraph in the rest of the paper.

2.2.1. Classical analysis over projections

In order to analyze this bipartite structure, it is natural to transform a bipartite graph into one of its projection in order to compute standard metrics defined for graphs. Let us recall briefly here those metrics and the usual properties shared by real-world networks [10].

Let $G = (V, E)$ be the (projected) graph. We denote by $N(v)$ the set of neighbors of $v \in V$: $N(v) = \{u \in V, (u, v) \in E\}$ and by $d(v)$ its degree: $d(v) = |N(v)|$.

The usual statistics used to characterize such a graph involve its size ($n = |V|$), its number of links ($m = |E|$), its highest degree ($d^+ = \max_v d(v)$), and its average degree ($k = \frac{2 \cdot m}{n}$). Over those notions, one can also study the density $\delta = \frac{2 \cdot m}{n \cdot (n-1)}$ that is usually small as real networks happen to be very sparse. Indeed the probability that a link exists between two randomly selected nodes is generally very small.

On the contrary, two nodes sharing a common neighbor have usually a high probability to be linked. This property is often referred to as the *local density* and is generally captured by the *clustering coefficient* and the *transitivity ratio* [10,33,34]. The first one computes, for every node $v \in V$, the probability that two of its neighbors are linked together. This is denoted by $\mathtt{cc}(v) = \frac{\Delta(v)}{\vee(v)}$ where $\Delta(v)$ is the number of triangles (sets of three nodes with three links) to which v belongs and $\vee(v) = \frac{d(v) \cdot (d(v)-1)}{2}$ the number of pairs of neighbors of v. The clustering coefficient of the graph is the average value $\mathtt{cc} = \frac{\sum_v \mathtt{cc}(v)}{n}$.

The second coefficient, the transitivity ratio, provides a more direct computation of the property over the whole graph. Let $\Delta = \sum_v \Delta(v)$ and $\vee = \sum_v \vee(v)$, then $\mathtt{tr} = \frac{3 \cdot \Delta}{\vee}$ is defined as the transitivity ratio of G.

A classical observation is that those two quantities are high, at least compared to the density. In other words, if one selects a random pair of links with an extremity in common (transitivity ratio) or a random node and two of its neighbors (clustering coefficient), then the probability that the third possible link exists is high.

2.2.2. Specific metrics for bipartite graphs

The metrics defined in Section 2.2.1 have the advantage to be well understood and allow for immediate analysis of the flat topology. On the other hand, the required projection leads to a loss of information. It is thus necessary to define extensions of those metrics on the bipartite structure itself.

From a bipartite graph $G = (\top, \perp, E)$ and for each top node $v \in \top$, we denote by $N_\top(v)$ the set of bottom neighbors of v: $N_\top(v) = \{u \in \perp, (u, v) \in E\}$ and by $N_\perp N_\top(v)$ the set of top neighbors of v: $N_\perp N_\top(v) = \{u \neq v \in \top, \exists x \in \perp : (u, x) \in E$ and $(v, x) \in E\}$. We use similar notations for the bottom nodes $N_\perp(v)$ and $N_\top N_\perp(v)$. For instance, on Fig. 2a, $N_\top(1) = \{A, B, C\}$ and $N_\perp N_\top(1) = \{2,3\}$. Similarly, $N_\perp(C) = \{1,2,3\}$ and $N_\top N_\perp(C) = \{A, B, D, E\}$.

Let $n_\top$ (respectively $n_\perp$) be the number of $\top$ (respectively $\perp$) nodes and $m_{\mathtt{bip}}$ be the number of bipartite links. We denote by $k_\top$ (respectively $k_\perp$) the average degree of $\top$ (respectively $\perp$) nodes and $\delta_{\mathtt{bip}} = \frac{m_{\mathtt{bip}}}{n_\top \cdot n_\perp}$ the density of the bipartite graph. In Fig. 2a, $n_\top = 4$, $n_\perp = 6$, $k_\top = 2.5$, $k_\perp = 1.6$, $m_{\mathtt{bip}} = 10$, and $\delta_{\mathtt{bip}} = 0.42$.

Those statistics are natural extensions of graph metrics. However, for the *local density*, there is no standard variant since, by definition, there is no triangle in a bipartite graph. As suggested by Latapy et al. [35], we will rely on the following coefficient that tends to capture the overlapping between the neighborhood of two nodes of $\top$

$$\mathtt{cc}_\top(u, v) = \frac{|N_\top(u) \cap N_\top(v)|}{|N_\top(u) \cup N_\top(v)|}. \tag{1}$$

This coefficient is interesting as it captures the relative overlap between neighborhoods of top nodes, i.e., $\mathtt{cc}_\top(u, v)$ is equal to 1 if the neighborhood of u and v intersects exactly, to 0 if they do not share any neighbor. If we apply the overlapping coefficient on nodes 1 and 2 in Fig. 2a, we have $\mathtt{cc}_\top(1, 2) = \frac{|\{A,B,C\} \cap \{B,C,D\}|}{|\{A,B,C\} \cup \{B,C,D\}|} = 0.5$.

From this coefficient, it becomes natural to define the clustering coefficient related to a specific $\top$ node v. This is given by

$$\mathtt{cc}_\top(v) = \frac{\sum_{u \in N_\perp N_\top(v)} \mathtt{cc}_\top(u, v)}{|N_\perp N_\top(v)|}. \tag{2}$$

Applied on node 1 of Fig. 2a, it gives $\mathtt{cc}_\top(1) = 0.375$. This coefficient enables to study the distribution of this property over the top nodes as well as its correlation with the degree or other properties. Then, one can naturally compute the *bipartite top clustering coefficient* $\mathtt{cc}_\top$ of G as the average value of $\mathtt{cc}_\top(v)$ over all the nodes v of $\top$. More formally

$$\mathtt{cc}_\top(G) = \frac{1}{|\top|} \sum_{v \in \top} \mathtt{cc}_\top(v). \tag{3}$$

Following those definitions, we can derive the dual $\mathtt{cc}_\perp(G)$ bottom clustering coefficient of G which finally leads to the global *clustering coefficient* of G defined by:

$$\mathtt{cc}_{\mathtt{bip}}(G) = \frac{n_\top \mathtt{cc}_\top(G) + n_\perp \mathtt{cc}_\perp(G)}{n_\top + n_\perp}. \tag{4}$$

3. Model

In this section, we explain how we model a subset of the router level topology as a bipartite graph. We explain our approach (Section 3.1) before evaluating it (Sections 3.2 and 3.3).

3.1. Methodology

Our methodology is sketched in Figs. 3 and 4. We first start by removing loops[1] in our dataset ("cleaning step" in Fig. 4). We then have a dataset that contains a set of L3 devices (routers) and L2 devices (typically switches) with links between them and among routers (see Fig. 3b). Because of these point-to-point links between routers (between R_2 and R_3 in this example), this is not a pure bipartite graph, as defined in Section 2.2. However, there is no difference between a point-to-point link and a pair of routers connected through a L2 device[2] that itself is connected only to these two routers. In Fig. 3b, the direct link between R_2 and R_3 can be replaced by a L2 device of degree 2 linking only R_2 and R_3 (see S_2 in Fig. 3c) without any loss or addition of relevant information. Indeed, if we $\perp$-project Fig. 3c, we get back the direct link between R_2 and R_3. In addition, there is no such L2 devices with degree 2 in the raw data as IGMP probing can only detect L2 devices connecting, at least, three routers. Consequently, we replace each point-to-point link between two routers by a new L2 device linking them without any loss of information. This results in a bipartite graph, that we call `BipReal`, that encodes exactly the same information as the raw data in the sense that we can retrieve at any time the original format of the data by only mapping the L2 nodes with degree 2 into a direct link between two
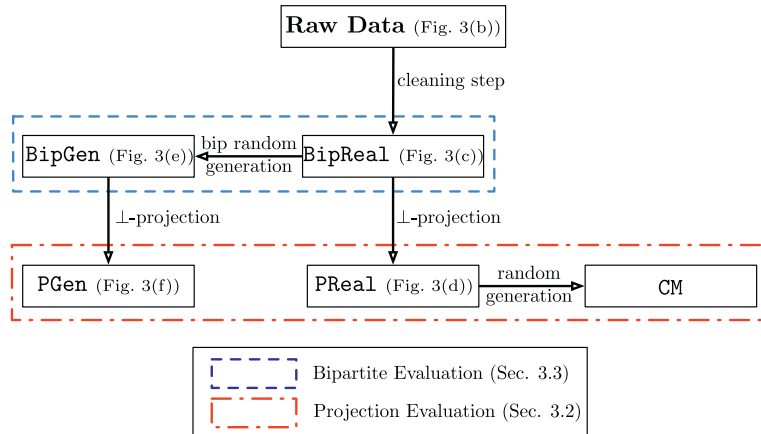
---

[1] It affects less than 0.3% of the links on average.

[2] The point-to-point link may actually be seen as a L2 device as it can be the case using OSPF.
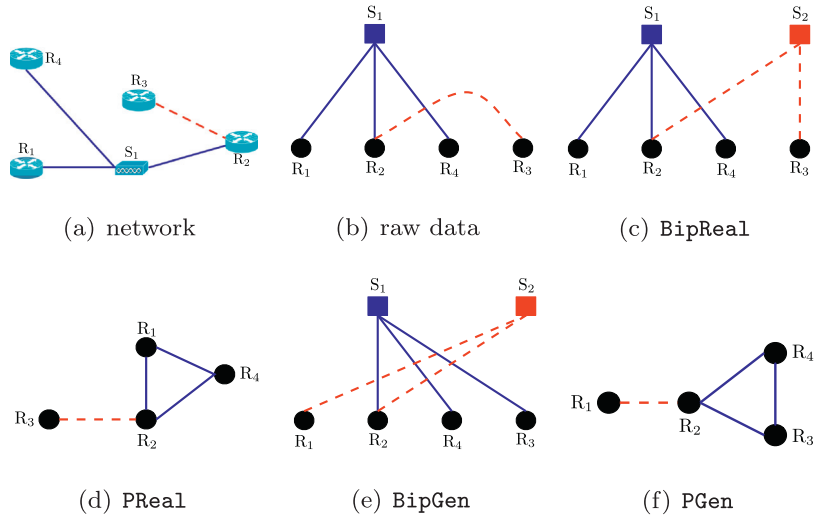
Fig. 3. Example of transformation between network, raw data, `BipReal`, `PReal`, `BipGen`, and `PGen`. A plain line refers to a point-to-multipoint link, while a dashed line refers to a point-to-point link.

Fig. 4. Model setup.

routers. Although this step has no impact regarding the projection, it concerns an important fraction of links in the real data since point-to-point connections represent 58% of all the links on average.

It is worth emphasizing here that the interest of considering point-to-multipoint connections in the modeling of Internet stands precisely in the fact that this new level enables to link several nodes together in a way that point-to-point connections could not do. Thus a degree-2 L2 device would be nothing more than a simple extension cable between two routers. So the fact that no L2 devices with degree 2 are present in the raw data is not really due to the measurement procedure but is more related to the fact that, by definition, a degree-2 device, if it exists, has to be considered as a direct link.

The classical modeling approach consists in computing the ⊥-projection `PReal` of `BipReal` (see Fig. 3d), and, then, in modeling it with a random graph `CM` obtained with

the *Configuration Model* [17–19]. This model produces a random graph with the node degree sequence given in input of the model, the one of `PReal` here. We claim that this approach is not satisfying as it does not allow to capture other properties than the one being part of the model. For instance, if one wants to also capture local density properties, one has to look for another model such as the one introduced by Newman [36]. However, this would lead to the same observation, i.e., no other properties than the ones injected will be captured and one has to look for another model if additional properties are desired.

Instead, we propose a model that relies directly on the real bipartite structure in order to generate graphs that will reproduce several aspects of the actual data. Our model consists in using the degree sequences of L2 and L3 devices in the bipartite graph `BipReal` in order to generate a random bipartite graph `BipGen` (see Fig. 3e) and project it into a standard graph `PGen` (see Fig. 3f). Following the

tradition of random models, the `BipGen` graph is obtained by shuffling the links between L2 and L3 nodes while maintaining the node degree distribution at both levels. Our expectation is that this bipartite representation of the data will produce a graph `PGen` close to the actual one `PReal`, closer than the `CM` one, in particular regarding other metrics than node degree distribution (e.g., the local density, degree correlations, etc.).

For the evaluation, we use the dataset provided by `mrinfo`, as described in Section 2.1 and [13]. From the 4 year daily dataset, we arbitrarily select, each month, the largest output file, leading thus to 56 global topologies (generally they exhibit a large connected component having more than 7000 nodes). From this subset, we more specifically focus on the largest topology, corresponding to the data collected by `mrinfo` on 2006/09/07.[3] We infer the presence of L2 devices following methodology discussed by Mérindol et al. [6, Section 2.3].

Note that, although the data obtained by `mrinfo` could be partial (as they only capture a subset of the Internet topology relying on multicast-enabled interfaces) and/or biased, the present work is independent from the quality of the data itself. The problem of improving the measurement tools is different from the one of identifying relevant properties able to exploit the features observed in the data, which is what we focus on in this paper. This means that when we further show the ability of the proposed model to reproduce the characteristics of real data, it has to be understood as the ability of reproducing the characteristics as *observed* in the data, with their flaws.

The rest of the section is devoted to the comparison of core statistics in order to assess the quality of the models. Section 3.2 focuses on statistics on the projection while Section 3.3 studies the statistics related to the bipartite level. Our purpose is to check whether this simple process provides good results, in particular regarding metrics that were not injected in the model. Recall that, during the whole transformation process, we only relied on the L2 and L3 node degree distributions. Connections between the two layer devices are then simply randomized without injecting any other structural relationship.

3.2. Projection evaluation

Here, we evaluate the projection by considering general statistics (Section 3.2.1) before going into details (Section 3.2.2).

3.2.1. General statistics

The first statistics we focus on concern some basic properties observed in most real-world networks [10], formally presented in Section 2.2.1. For each metric, the right part of Table 1 (labeled as "Ratios") positions the data used in this paper (the column labeled as "2006") with respect to the set of 56 IGMP topologies (the column labeled as "Avg case"), each `BipGen` topology being generated 10 times, thus leading to 10 corresponding `PGen` projections. The left

Table 1
Global statistics for projection evaluation.

	Raw data			Ratios	
	PReal	PGen	CM	2006	Avg case
n	9740	9749	9740	1.00	1.00
m	35,567	48,877	35,470	1.37	1.32
δ	7.5	10.3	7.5	1.37	1.32
k	7.3	10.0	7.3	1.37	1.32
d^+	58	234	58	4.03	2.93
tr	0.88	0.53	0.01	0.60	0.72
cc	0.58	0.42	0.00	0.72	0.73

part of Table 1 (labeled as "Raw Data") provides absolute values for the `PGen` and `PReal` graphs according to the largest topology used over the paper.

From Table 1, one can see that the number of links, m, is significantly higher for `PGen` than in the actual graphs (around 37%). It follows naturally that the density ($\times 10^{-4}$ in Table 1) and the average degree are also higher for `PGen`. As explained later in this paper, it comes mainly from the fact that there exists overlaps and significant correlations between the two levels of nodes that are not necessarily preserved during the randomization process. On the other hand, the `CM` graph is particularly close to actual values regarding the same properties. This is not surprising since this model focuses precisely and only on the degree sequence of the projection. Looking at the transitivity ratio and the clustering coefficient, Table 1 reveals that the `CM` model is unable to take into account the local density captured by those coefficients. The `PGen` model seems, on the contrary, able to capture it (although the values are quite different) in the sense that the local density is relatively very high compared to the global density, the key point for this property.

Finally, it is worth noticing that, for several properties, Table 1 reveals that the selected topology positions itself in a worst case scenario compared to the averaged results over the 56 topologies. This is particularly obvious for the highest degree. This indicates that the conclusions drawn from the analysis of this particular case would also be relevant for the other dataset.

3.2.2. A deeper analysis

In order to refine the general statistics provided in Section 3.2.1, Fig. 5 presents the distribution of the degrees for the real data, `PReal`, and the random graphs generated by the two methods, `PGen` and `CM`. The horizontal axis, in log-scale, is the degree of the nodes, while the vertical axis, also in log-scale, presents the inverse cumulative mass. As expected, the `CM` model is very efficient (it is superimposed on `PReal` in Fig. 5) as its process is precisely to mimic the degree sequence given in input, i.e., the one of `PReal`.

The slight differences observed stem from the cleaning steps (removing multiple-links, loops, etc.) made during the generation. Regarding the `PGen` method, one can see that it is less efficient but it shows a similar distribution. One might notice that the main differences are located in the higher degrees. This is also corroborated by Table 1.

Fig. 5. Inverse cumulative degree distribution.

Fig. 6. Clustering coefficient inverse cumulative distribution.
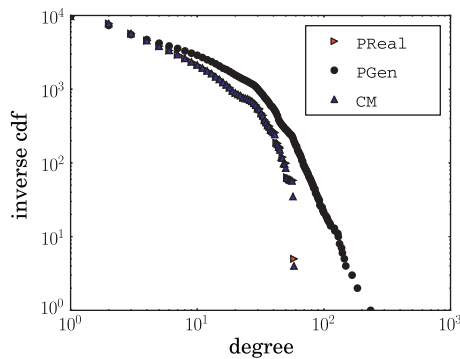
The highest degree is significantly higher for the `PGen` graph than the real one (234 instead of 53). It partially comes from the fact that, although the generated bipartite graph respects the degree distribution of the routers and the L2 devices, it does not ensure that the overlapping of the L2 devices is preserved, thus increasing the degrees of L3 nodes in the projection. This potential overlapping and other possible correlations between the two layers of node will be investigated more precisely in Sections 3.3 and 4.2.

Fig. 6 presents the inverse cumulative distribution of the clustering coefficient for the real data, `PReal`, and graphs generated by the two methods, `PGen` and `CM`. Note that the plots are normalized over the number of nodes with degree $\geqslant 2$ in order to avoid side effects from the nodes of degree 1, for which the notion of clustering coefficient is inadequate. Fig. 6 clearly shows that the `CM` model is unable to provide a correct representation of such a distribution. This is corroborated by Table 1 as the clustering coefficient as well as the transitivity ratio are close to 0. This is due to the fact that the model does not consider the local density and that the number of triangles is very low (only 1299 triangles while the actual graph has over 203,608 ones). On the other hand, the `PGen` graph provides a similar progression, but with a significant shift of the values.

Fig. 7 shows the correlation between the node degree and the average clustering coefficient, i.e., a (x, y) dot means that the average clustering coefficient for the nodes having degree x is y. Fig. 7 confirms the analysis made above. Whatever the degree of a node in the `CM` model, its clustering coefficient remains close to zero. The `PGen` graph, on the other hand, is able to present a similar scatter plot shape, although the values are significantly different. More interestingly, one can see that high clustering coefficients are related to nodes having a similar degree on both figures.

The main difference concerns small degree nodes. For instance, nodes with degree 2 in `PGen` graphs have an average clustering coefficient of 0.1 while actual ones are close to 0.4. This indicates an interesting characteristic of the two bipartite structures. Whereas in the actual bipartite, it seems that, when a router is connected to two others routers, they tend to share L2 devices. This is absolutely
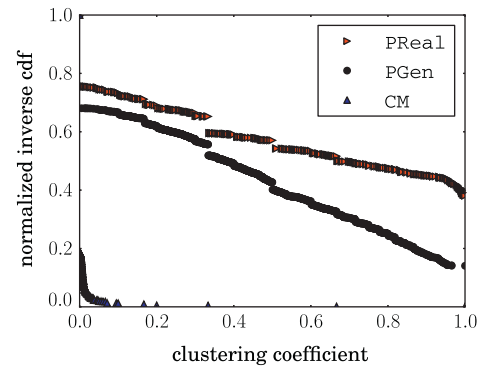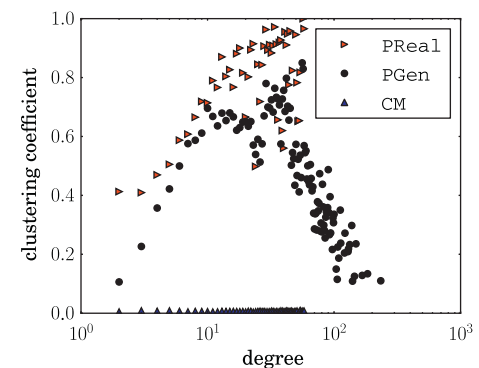

Fig. 7. Average clustering coefficient associated to a given degree.

not the case for the `PGen` graph. This particular difference concerning degree-2 nodes can be explained by the L2 devices added during the first step of the `PGen` generation (see Section 3.1). A deeper study of the degree correlations in the bipartite structure will confirm this statement (see Section 4.1).

This first analysis made on the projected graphs confirms the relevance of using bipartite structure to model the data as it succeeds in reproducing *globally* the characteristics of the real network. In particular, it is able to capture metrics that are not part of the model. This is a significant improvement in itself since the usual way to obtain properties is to encode them directly in the generation process, which we claim is not satisfying in a long term perspective.

3.3. Bipartite evaluation

This section intends to better characterize the differences observed between real (i.e., `BipReal`) and L2L3 (i.e., `BipGen`) projections from the point of view of the bipartite structure. Following notations presented in Section 2.2.2, we compare standard properties of bipartite graphs.
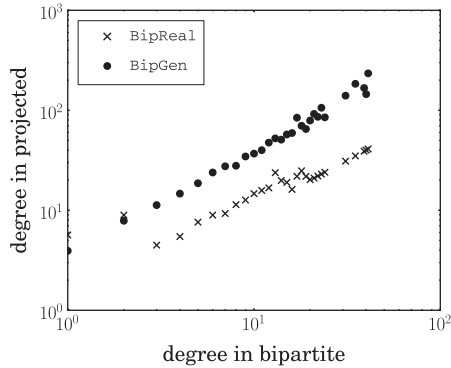
3.3.1. General statistics

Table 2 gathers the statistics presented in Section 2.2.2 for the real and the random bipartite graphs, where ⊤

Table 2
Global statistics for bipartite evaluation.

	BipReal	BipGen
n_{L_2}	10,224	10,224
n_{L_3}	9758	9758
m_{bip}	25,422	25,415
k_{L_2}	2.5	2.5
k_{L_3}	2.6	2.6
δ_{bip}	0.00025	0.00025
cc_{bip}	0.37	0.27

Fig. 8. Correlation between degrees in the bipartite and in the projection.

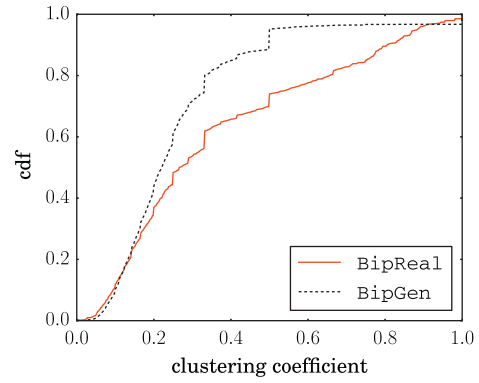**Fig. 9.** Clustering coefficient distribution.

Fig. 10. Degree correlation.

refs to L2 nodes (L_2) and ⊥ to L3 nodes (L_3). It shows that all the simple properties are respected by the random bipartite graph, except for the bipartite clustering coefficient for which a slight shift is observed. Note that we do not present the ratios given in Table 1 here since they are all equal to 1 (either for this specific case or the average ones), except for the bipartite clustering coefficient for which our case ratio (0.73) is slightly worst than the average value (0.78).

Those observations show that our model succeeds in preserving the global characteristics of the real bipartite structure but do not provide insight on why the projections differ. This is why we turn now to a more refined analysis over those notions.
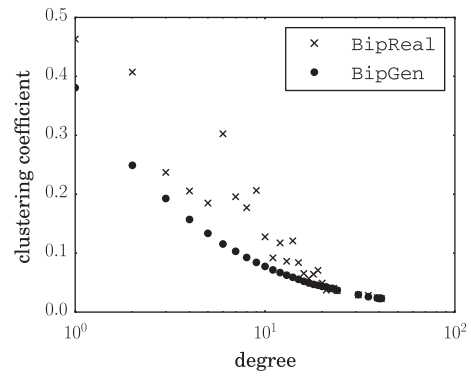
3.3.2. A deeper analysis

First, Fig. 8 presents the correlation between the degree of L3 nodes in the bipartite graph and their average degree in the projection, i.e., a (x, y) dot means that the nodes having degree x in the bipartite structure have an average degree y in the projection.

Fig. 8 shows that the behavior is similar in both cases. In particular, they both follow a straight line in the log–log scale for x values ⩾3 with a similar slope. But two important differences are noticed. First, the values are significantly lower for the actual bipartite. This indicates some redundancies in the bipartite structure, meaning that many neighbors of nodes in the projection share actually several common L2 devices in the bipartite. This overlapping pattern induces the lowering of their degree in the projection.

From the BipGen points in Fig. 8, one can conclude that this redundancy over the L2 nodes is seemingly lost when shuffling the links in the bipartite graph. Note that this is true in particular for high degree nodes, suggesting that the difference observed for the highest degree in the projection might be due to this redundancy. Another difference can be pointed out for low degree nodes for which the remark stated above does not stand. Degree-1 nodes in particular present the opposite situation: in real bipartite, the single L2 device to which they are connected happens to have a relatively high degree (close to 6 on average). This differs both from the tendency observed for nodes with degree ⩾3 and from the random case for which the correlation is consistent for all degrees.

Figs. 9 and 10 focus on the bipartite clustering coefficient as defined in Section 2.2.2. Fig. 9 presents the cumulative bipartite clustering coefficient of L2 nodes for real and random bipartite graphs, while Fig. 10 shows the correlation between degree of L2 nodes and their average bipartite clustering coefficient (i.e., a (x, y) dot means that the average bipartite clustering coefficient for L2 nodes having degree x is y).

Both figures show that the two bipartite graphs have a similar behavior regarding this coefficient although a non-negligible fraction of nodes in the real bipartite has a higher clustering coefficient than in our model. This is particularly true for low degree nodes (Fig. 10). This means that low degree L2 nodes tend to share their neighbors
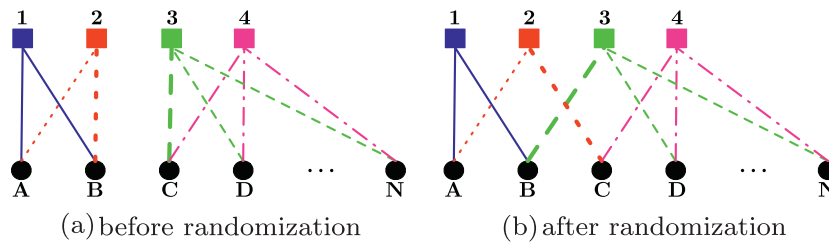
Fig. 11. Effects of bipartite random generation (`BipReal` to `BipGen`) on L3 degree.

with other L2 nodes. This phenomenon explains the gap observed in Fig. 9 for high clustering coefficients and corroborates the difference observed for the global cc_{bip} statistics in Table 2. It strengthens also our former remark on the redundancy that seems to be more important in the real bipartite topology than in the random one and that explains the differences observed on high degree nodes in the projection.

In order to test this hypothesis, we compute two more refined properties. The bipartite clustering coefficient, although dealing with overlapping of L2 nodes, is defined for pairs of nodes. We might want to use a more direct notion defined for a single L2 node. One possible solution is to use the *redundancy coefficient* [35] defined for all L2 nodes v as the fraction of pairs (u, w) of L3 neighbors of v that are connected to a common L2 node other than v. When such a case occurs, then (u, w) is linked in the projection whether v exists or not. Thus, we might consider v as redundant as far as u and w are concerned. Our analysis shows that the two topologies behave very differently regarding this coefficient: on average, 27% of L3 nodes pairs connected to a L2 device in a real case would not be affected by the removing of this device in term of their link in the projection. This proportion drops to 0.3% in the random case.

The notion above focuses on the L2 nodes that are redundant for the projection. One might similarly define a notion of redundancy over the links, i.e., the links that would not modify the projection if they were removed from the bipartite graph. Let us call *internal link* such a link [37]. Our analysis shows that 13.7% of links in the real bipartite graph are internal links, while this proportion is only 0.2% for the random case.

These two last properties are clearly in relation with the notion of degree in the projection and, as such, explain partially the differences observed in Section 3.2. A deeper analysis is left for further works but Section 4.3 already provides interesting directions to improve our model.

All the properties explored in Section 3 show the benefit one can gain from modeling such L2–L3 data with bipartite graphs. While it offers support for generating flat graphs that are able to reproduce *qualitatively* several and independent properties of the original data (see Section 3.2), it also proposes new mathematical tools to analyze its structure from the point of view of the bipartite graph itself. In particular, it allows for identifying which aspects of the real network might stem from random processes and which ones are due to strong designed patterns.

4. Discussion

In order to better understand the limitations of our model illustrated in Sections 3.2 and 3.3, we investigate here two interesting properties: (i) we evaluate the effects of the bipartite random generation on L3 degrees and (ii) we study the redundancy between L2 devices (i.e., we analyze the cases in which removing a L2 node or a L2–L3 link would affect or not the L3 projection).

While the first property (i), detailed in Section 4.1, allows us to emphasize an interesting correlation between the L3 degree and the L2–L3 degree, the second property (ii), discussed in Section 4.2, allows us to exhibit strong patterns and to explain how point-to-multipoint connections "behave" in real networks. Finally, we envision two possible extensions to improve our model in Section 4.3.

4.1. Correlation analysis

Although our model respects the degree distribution of both L2 and L3 devices, there remains one important difference between the raw data and the proposed bipartite structure. As explained in Section 3.1, due to the definition of bipartite graphs, we first replace any point-to-point connections between routers by a virtual L2 device connecting them. Although this modeling is strictly equivalent to the point-to-point connection for the projection perspective, it might have an impact on the structure during the randomization process.

Indeed, as shown in Fig. 11, a simple rewriting in the bipartite graph may induce an important modification for the degree of the nodes in the projected graph. This is the case for node *B* in this virtual example. Before the randomization, it is connected to a unique L3 node (both via node 1 and 2, see Fig. 11a), thus having degree one in the projection. But simply switching the extremities of links (2, *B*) and (3, *C*) leads to a new bipartite graph (see Fig. 11b) in which *B* is now connected to every nodes (to *A* via 1 and to all others via 3), thus increasing drastically its degree in the projection. Obviously, this example is an extreme case but it illustrates how the randomization process at the bipartite level may affect the degree properties of the projections.

In order to study how such a randomization may impact the generated graphs, we investigate how routers are connected to L2 and L3 devices in the raw data and in random graphs (considering here only "actual" L2 devices). Fig. 12 shows the correlation between L3 degrees and the number of point-to-multipoint connections both for real data and
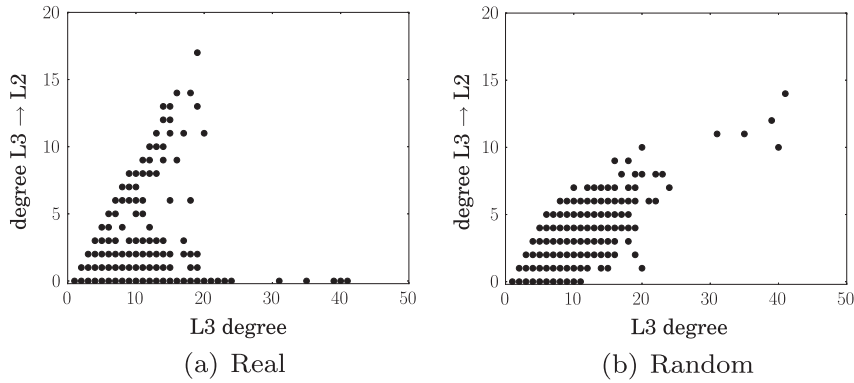
Fig. 12. Degree versus point-to-multipoint degree.

random bipartite graphs. On each plot, a (x, y) dot stands for a router having x links in the bipartite graph, y of them being with a L2 device with degree strictly higher than 2 (recall that a L2 node with degree 2 stands precisely for a point-to-point connection).

Fig. 12a shows a striking fact: routers with degree higher than 20 have no connection to real L2 devices but only to point-to-point connections. As explained above in the example, our randomization process does not verify such a strong characteristic, as it can be noticed in Fig. 12b. This means that, in our model, it is likely that routers having a high L3 degree (i.e., routers having only point-to-point links) will be connected to actual L2 devices (whose degree is strictly greater than 2), thus increasing, and potentially significantly, their degree in the projection.

This observation on real data leads us to explore a more constrained model able to preserve the correlation between the number of point-to-point and point-to-multipoint connections of routers collected in the ground data. It relies on splitting the L3 degree into two disjoint values: one for the number of point-to-point links and one for the number of point-to-multipoint ones. Once such a couple of degrees for each L3 node has been defined, it is easy to adapt our former model to cope with this distinction.

Surprisingly, this approach does not improve significantly the properties observed in the generated projections (compared to the ones observed in the real projections). More precisely, most of the characteristics of the generated hybrid projections are close to those of the random projections (see Table 1). The only improvement we observe is in the highest degree which is a bit lower than in the basic model but still higher than in the dataset (we obtain a ratio of 3.29 instead of 4.03). But the overall degree distribution and correlations studied in Section 3 are not improved. This weak improvement is partially due to the low number of large degree routers in the raw data set (less then 1% of routers have a degree higher than 20), limiting so their impact on global properties.

In order to better understand the inadequacy of the hybrid approach, we also look in detail at the properties of the random hybrid bipartite structure. It turns out that they match exactly those of the random bipartite graph (shown in Table 2). Besides, as for the basic model, redundancy patterns are lost in the randomization process. For instance, the redundancy coefficient is still very low (0.16%) compared to the one observed in the real bipartite (27%). We pointed out previously that this is a plausible explanation to the model limitations. The redundancy coefficient is clearly in relation with the degree properties of the projections and thus with the quality of such properties in the model. This last observation explains why this hybrid approach fails to enhance the quality of the model accuracy. In the next section, we study in more depth the property related to the redundancy coefficient.

4.2. Redundant networking patterns

At the end of Section 3.3, we identified an interesting property using the redundancy coefficient and internal links. Indeed, the data considered in this paper exhibits many redundant Point-of-Presence (PoP) patterns.

Fig. 13 illustrates such redundant patterns between L2 devices observed in the raw data. In both Fig. 13a and b (both figures come from the Level3 London PoP observed in 2011 and 2007), one can guess that the redundancy coefficient and the number of internal links are high. Such network structures, generally required for physical/logical redundancy and/or load balancing, are not random. Thus, these structures, favoring the network robustness, imply that the degree in the projected graph will be lower for the projection of the real network than for the projection of the random bipartite graph.

In Fig. 13a, we can observe that the two L2 devices generate two cliques of $i + 1$ routers that only differ on EBR1 and EBR2 (while these two routers are connected through multiple parallel point-to-point links). As a result of the projection, i links will disappear in the projection of the real graph while it is likely that, in the random bipartite graph, those links will be distributed over all the network: the projection will then have a higher average and maximal degree.

The example given in Fig. 13b exacerbates this observation: here, while two of the six L2 devices[4] interconnects

---

[4] Note that it is possible that such a symmetry involves some VLAN configurations leading so to two physical L2 devices having three VLANs. Furthermore, such an evolution between 2007 and 2011 suggests an improvement in the architecture capacity.
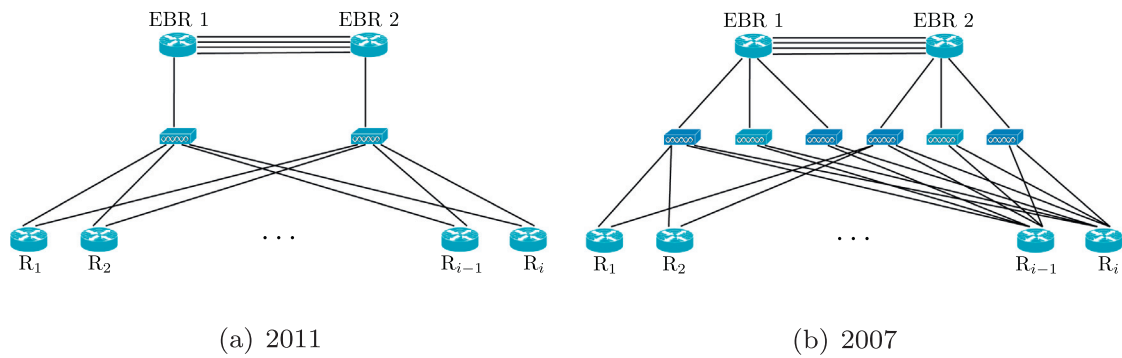
(a) 2011   (b) 2007

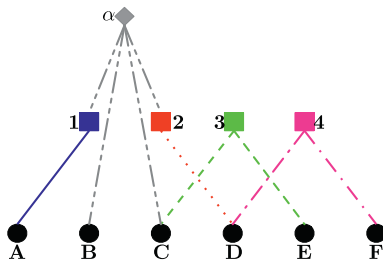**Fig. 13.** PoP configuration in London for Level3.

Fig. 14. A Tripartite view of IP networks.

the i routers, $i − k$ other routers are connected again through the four others L2 devices. This kind of configuration is not that rare and can, at least, partially explain our random model limitations on metrics previously highlighted.

4.3. Next steps

Observations made in Sections 4.1 and 4.2 open the way to the design of improved models, potentially fixing lacks of the former one. The two improvements we envision belong to two distinct families of models: *random* and *structural*.

On the one hand, we could rely on the strong redundancy patterns highlighted in Section 4.2. Based on such observations, it becomes natural to attempt to capture the redundant PoP patterns illustrated in Fig. 13. One possible way would be to encode the overlapping among L2 devices in the model itself. To do so, one can extend the bipartite structure into a *tripartite* one using a third level to integrate such a redundancy.

In practice, one can encode any overlapping among L2 devices by the addition of a new node at a third level (L1), connecting both the L2 and L3 devices they are covering (see [38]). Applying this procedure on the bipartite graph of Fig. 2a, for example, would result in the tripartite structure presented in Fig. 14. Indeed, the nodes 1 and 2 are both connected to nodes *B* and *C* in the bipartite graph. This overlapping is then encoded in the tripartite graph by the addition of a new third-level node α connecting *B*, *C*, 1, and 2.

Although such L1 nodes are artificially introduced, they intend to encode real networking logical patterns. As explained previously, an L1 inter-connection may result from some L2 logical sub-layer as VLAN. It can be used for traffic engineering purpose such as fast-rerouting or load balanc-

ing to circumvent capacity limitations. We envision to capture that kind of (frequent) redundant patterns by encoding them in a third layer of inter-connection: L1 nodes are logical (on the contrary to L2 ones that are physical) and represent the subset of routers that are connected to a given set of L2 nodes.

Once such a tripartite structure has been defined, one can easily apply similar randomization processes than the one proposed in this paper. This process would shuffle independently L1–L3 links and L1–L2 links but preserve the structure defined by the new third level. This would result in generating a new tripartite graph presenting the same redundancy patterns than the original ones that could eventually be projected into a bipartite graph.

On the other hand, we could follow the path opened by the observations in Section 4.1: there are evidences of a significant correlation between a router degree and the degrees of L2 devices connected to it. We observed, for instance, that very small degree routers are connected to high degree L2 devices (on average). More generally, backbone routers (in an AS core) have a large degree mostly composed of L3–L3 links, while access routers (providing Internet services to clients) exhibit a lower degree and, generally, mainly composed of L3–L2 connections. However, although this kind of design features can bring improvements in the ability of the model to reproduce this specific property, it also comes with a loss of generality in generated graph properties. As mentioned in the introduction, we believe that random models, such as our tripartite proposal, are more suited for formal analysis.

The two directions suggested here may take several forms. They would require to investigate different mapping as well as to define new extensions of metrics proposed for the bipartite structure analysis. We leave these promising directions for future work.

5. Topology generation

This section presents a first implementation of a probabilist topology generator relying on an intermediate bipartite graph. Our generator is freely available from http://code.google.com/p/py-bipartite/. It generates bipartite and projected graphs that can be saved to most common graph formats.

The difference between the generator and the bipartite model presented in Section 3 is that the generator does not use a prescribed degree sequence obtained from real

bipartite graph but, rather, uses probability distributions. This makes possible the synthesis of a larger number of graphs by varying the sizes of the L2 and L3 sets or the parameters of the probability distributions.

5.1. Methodology

The generator applies a few simple steps. First, L2 and L3 degrees are drawn from a probability distribution. Second, a bipartite graph is generated by randomly connecting L2 vertices to L3 vertices. Finally, a flat graph is produced by projecting the bipartite graph on to the L3 set. The remaining of this section describes each of these steps in further details.

To draw L2 and L3 degrees, the current version of the generator is based on a discrete power-law distribution. For several practical reasons, we opted for the *Zipf distribution* [39], a discrete power-law distribution. The support of the Zipf is a finite interval of natural numbers $[x_m, N]$, $x_m \leqslant N$. Outside of this interval, the probability mass function equals 0. This property is convenient as it allows us to define a limit on the maximum degrees that can be drawn. Recall that large degrees in the bipartite graph can lead to extremely large degrees in the final projected graph (as already noted in Section 3). Thus, this feature of Zipf distributions is of the highest interest compared to an unbounded Zeta discrete distribution or a standard Pareto continuous distribution.

The probability mass function of the Zipf distribution is given by

$$p(k) = \frac{k^{-\alpha}}{C(N, \alpha, x_m)}, \tag{5}$$

where $k, x_m, N \in \mathbb{N}, N \geqslant k \geqslant x_m$, and $\alpha \in \mathbb{R}, \alpha > 1$. The term at the denominator, C, is a normalization factor that ensures the sum of the probability mass function values over the support interval equals to 1. The value of $C(N, \alpha, x_m)$ is given by $\sum_{i=x_m}^{N} i^{-s}$.

To generate a bipartite graph, two distinct Zipf random variables are used: one for the L2 degrees, the other for the L3 degrees. These degrees must satisfy the following basic property of bipartite graph: the sum of the L2 degrees must match the sum of the L3 degrees. To ensure this property, one can check *a priori* that their expectations will match, or more formally $E(Z_{L2}) \cdot n_{L2} = E(Z_{L3}) \cdot n_{L3}$, where n_{Li} is the number of the Li vertices and $E(Z_i)$ is the expectation of the random variable Z_{Li} from which Li degrees are drawn.

The expectation of the Zipf distribution is given by

$$E(Z) = \frac{\sum_{i=x_m}^{N} i^{-s+1}}{C(N, \alpha, x_m)}. \tag{6}$$

Even if the expectations for the sums of degrees match, it is possible that the degrees actually drawn from the random variables do not have matching sums. In this case, we follow the suggestion by Newman et al. [18]: we randomly pick one L2 node and L3 node, discard their prescribed degrees, and draw new degrees. This process might need to be repeated several times but remains practically feasible. Most of all, it ensures that there is no bias in the redrawing step.

Finally, the bipartite graph is produced by considering that each node has a number of half-edges equal to its prescribed degree. The half-edges of L2 nodes are then randomly connected to half-edges of L3 nodes. This process is equivalent to the random generation of `BipGen` from the degree sequences of `BipReal` described in Section 3.1.

5.2. Evaluation

This section provides a first evaluation of our generation methodology. We aim at quantifying the accuracy of our probabilistic redrawing process to mimic real degree sequences as well as verifying that basic properties observed in bipartite graph projections are still coherent with our model results. We rely on the `mrinfo` dataset described in Section 2.1. For clarity reasons, as in Section 3, our discussion will focus on the graph collected on 2006-09-07. We consider the `BipReal` bipartite graphs built from the raw `mrinfo` data by converting any L3–L3 edge into a pair of L3–L2 edges and an artificial L2 device, as explained in Section 3.1.

First, the L2 and L3 degrees of the `BipReal` graphs are modeled by two Zipf distributions. The exponent of the Zipf distribution is estimated using the method described by Clauset et al. [40]. Power-law distributed integers are approximated as continuous reals rounded to the nearest integer and the maximum likelihood estimator $\hat{\alpha}$ for the exponent is computed. Unfortunately, the degree distributions in the `BipReal` graphs do not show a scaling behavior along all the degrees. For L2 (respectively L3), the scaling behavior is only observed for degrees equal to and above $x_m = 3$ (respectively $x_m = 2$). One of the reasons is the large number of artificial L2 devices added to model the L3–L3 edges. Another reason is a limitation in the `mrinfo` measurement itself: the raw data does not contain nonresponding IP addresses (the ones seen as neighbors by some responding router). The estimated exponents for the L2 and L3 degree distributions are computed based on their respective minimum values x_m.

Table 3 summarizes the `BipReal` statistics and metrics computed for the 2006-09-07 instance. $\hat{\alpha}$ denotes the power-law exponent estimate while $E(Z)$ is the expectation of the power-law distribution.

Fig. 15a and b shows the frequency distribution of L2 and L3 degrees above or equal to x_m for the selected instance using a log–log scale. Those two figures also show a plot of the probability mass function of the random variables using the estimated exponents. The match obtained using the maximum likelihood estimator to compute the Zipf exponent seems quite satisfying for degrees above or

Table 3
General statistics and parameter estimates for `BipReal`.

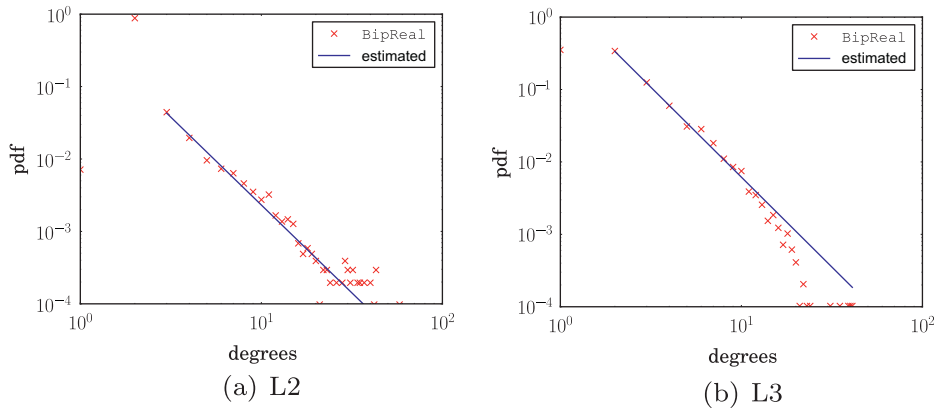|	L2	L3
n	10,227	9758
d^+	58	41
k	2.49	2.61
x_m	3	2
$\hat{\alpha}$	2.43	2.48
$E(Z)$	1.85	1.73

Fig. 15. Frequency distribution of degrees and PDF of estimated distribution, for degrees above or equal to x_m.
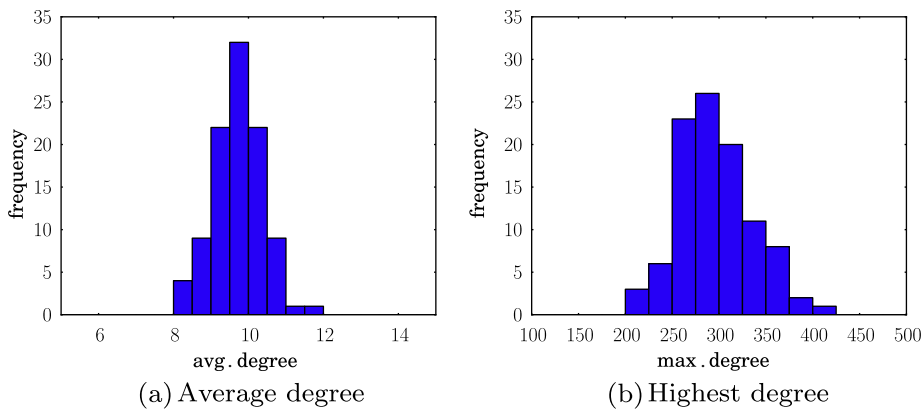

Fig. 16. Distribution of the average and highest degrees among the synthetic instances.

equal to x_m. For the low L2 degrees (under x_m) the Zipf model over-represents the degree 1 vertices while it under-represents the degree 2 vertices. For the low L3 degrees, the degree 1 vertices are over-represented. This has an impact on the L2 and L3 degree expectations $E(Z)$ which are smaller than the respective average degrees k. Although the remaining of this section will show that this does not prevent the projected graphs to exhibit realistic properties, it can be a motivation to use another probability distribution than the Zipf one. However, we believe that, as mentioned previously in Section 4.3, splitting the L3 degree distribution in its two sub-distributions, L3 → L2 degrees (pure-bipartite) and L3 → L3 degrees (pure-L3), could already bring significant improvements. Indeed, in the pure-bipartite graph, the degrees under x_m are more marginal and the scaling region captured by the Zipf represents a large fraction (88%) of the L2 degrees. Since our generator intends to be as simple as possible in term of the number of input parameters, we leave this hybrid redrawing approach for further works.

To generate graphs according to the above model, the generator is invoked with a set of eight parameters: three parameters (α, x_m, N) for both the L2 and L3 Zipf random variables and two for the cardinals of the L2 and L3 sets.

To ensure the expectations of the sum of degrees match, the size of the L2 or L3 set can be left undefined and the generator will compute the right size.

To further illustrate the generation process, we generated 100 bipartite graphs with the parameters estimated from `BipReal`. As our objective is to produce projected graphs of the same size as `PReal`, we fixed the size of the L3 set to 9758. We let the size of the L2 set undefined and the generator computed a size of 9149 L2 vertices. Note that the L2 size is slightly lower than in `PReal` (10,227). This can be explained by the expected L2 degree that is slightly larger than the expected L3 degree while in `PReal`, the average L2 degree was slightly lower than the L3 degree.

The expected sums of degrees equal to 16961.86 for L2 nodes and to 16960.97 for L3 ones.[5] During the generation of the 100 synthetic instances, the number of redraws required was limited to 60,739 in the worst case and 8789 on average. The minimum number of redraws was 23.

We now focus on the projected graphs. Fig. 16a and b shows the distribution of, respectively, the average and

---

[5] The difference comes from rounding during the computation of the size of the L2 set.

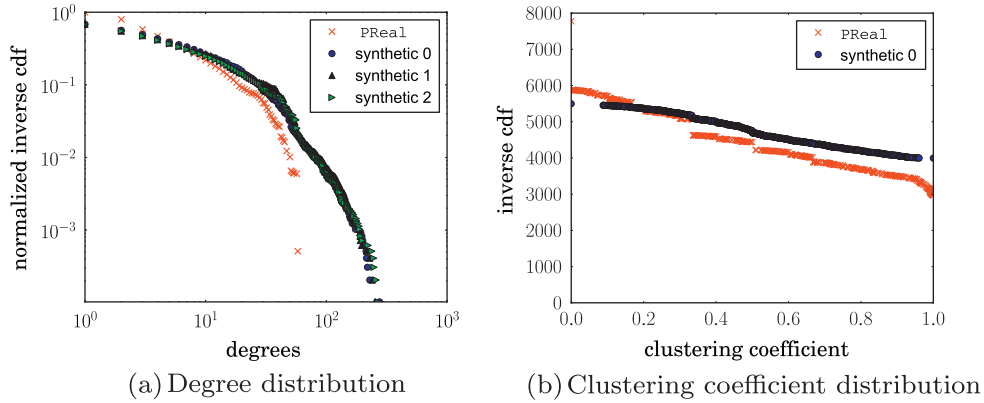(a) Degree distribution   (b) Clustering coefficient distribution

Fig. 17. Comparison of `PReal` with synthetic instances.

maximum degrees among the 100 instances generated. We observe that the maximum degree can go as high as 408 but is less than 290 in 50% of the instances. The average degree is 9.73 on average and tops at 11.85 while the average degree of the `PReal` graph is 7.29. As expected, we can notice that differences observed here are in the same order than the ones revealed with our model using prescribed sequences.

Fig. 17a shows the degree distribution in `PReal` and in the first synthetic instances. The main difference lies in the number of unconnected nodes: `PReal` contains only 18 while the first synthetic instance contains 3058. The reason for this difference is the larger amount of degree-1 L2 vertices in the synthetic bipartite graphs. Both the `PReal` and synthetic distributions seem to show a scaling region up to a certain degree, but the tail seems to be cut off. Note that this behavior is achieved for the synthetic instances thanks to the highest degree bound of the Zipf distributions used to generate the bipartite graphs.

Fig. 17b shows the clustering coefficient distribution in `PReal` and in the first synthetic instance. This instance is representative of the others as their respective curves would be hardly distinguishable on the figure. Note that the clustering coefficient is only defined for nodes with a degree higher than 1. This amounts for 5496 nodes (57%) in the synthetic instances while it concerns 7777 nodes (80%) in `PReal`. The reason for such a difference is in the frequency of very low degree nodes. This explains why the top left point of the inverse CDF does not match for both curves. In the remaining of the paragraph, we only consider nodes for which the `cc` is defined. In `PReal`, a significant fraction of the nodes (1900, 24%) have a clustering coefficient equal to 0, another large fraction (2961, 38%) have a `cc` equal to 1, and the remaining nodes have a `cc` relatively well distributed between 0 and 1. In the synthetic instances, 3995 nodes (72.3%) have a `cc` equal to 1 and the remaining nodes a `cc` between 0.08 and 1. The synthetic instances seem to not very well capture the low `cc` nodes, but have a behavior similar to `PReal` for the remaining nodes.

Those results show that our proposed generator is able to provide results similar to the former model, at least qualitatively. But a careful reader might notice that the

Table 4
Global statistics for the bipartite evaluation of the generator.

	BipReal	synthetic 0
n_{L_2}	10,224	9149
n_{L_3}	9758	9758
m_{bip}	25,422	17,098
k_{L_2}	2.5	1.8
k_{L_3}	2.6	1.9
δ_{bip}	0.00025	0.00019
cc_{bip}	0.37	0.31

behavior of the clustering coefficient for the generated topology are different from the model presented in Section 3. Indeed, in Fig. 17b, the curve for the synthetic instance is above the one of `PReal` while in Section 3.2 it was the opposite. This difference can be explained by the generation of the synthetic degree sequence. Indeed, as mentioned earlier, in order to ensure that the expectations of the sum of degrees match, we had to release the constraint on the number of L2 nodes. This, in turn, happened to have a major impact on the number of degree-2 nodes which are very few in the synthetic instances compared to the real data. Although studying the impact of generating synthetic degree sequences on the model seems very interesting and should help understanding better the relation between the parameters, we claim that such a formal investigation is beyond the scope of the present work and let such a study for further work.

Finally, in order to assess the quality of the generated bipartite structure itself, we performed the same analyses as in Section 3.3 by computing the standard metrics defined at the bipartite level. Those are shown in Table 4. We observe that, despite some differences, the statistics are quite well reproduced by the generator. As explained above for the clustering coefficient distribution, it turned out that the differences are due to the generation process of synthetic degree distributions. Beside those statistics, we also computed the redundancy coefficient and found that it is very different for the generated bipartites (0.5%) and the real one (27%). We obtained a similar result with the model presented in Section 3 where the observed redundancy coefficient was only 0.3%.

To summarize, except some limitations that are partially due to the difficulty to synthesize the raw data, our generator is able to provide almost equivalent results than the former model. The tradeoff between our generator accuracy and simplicity (and generality) seems very interesting to reproduce main real network characteristics.

6. Conclusion

Recent developments in Internet topology discovery have opened new perspectives in modeling IP networks. Indeed, when considering IGMP probing instead of standard traceroute probing, one can distinguish two layers of networking devices in the collected dataset: point-to-point connections between layer-3 (L3) devices (such as routers) and point-to-multipoint routers through layer-2 (L2) devices (such as switches). Prior to this work, to the best of our knowledge, those two layers have not been integrated into topology models.

In this paper, we proposed a bipartite model of the Internet topology, i.e., an Internet model relying on graphs in which vertices are divided into two disjoint sets (L2 and L3 nodes) such that every edge connects a vertex in one set to one in the other. Our model has the advantage of being "simple" in the sense that it does not require more than two metrics as inputs: we only consider the L2 and L3 node degree distributions. In addition, by confronting this model with measurement data obtained by `mrinfo`, we proved that it provides interesting behaviors: it succeeds in reproducing qualitatively different and independent properties of the original data. This result is particularly promising regarding the metrics that were not directly injected in the model (such as the clustering coefficient and some degree correlations) which is in sharp contrast with usual results in the domain.

In a second step, using metrics defined for bipartite graphs, we provided evidence of the relation between the properties of the bipartite structure and the ones of its projection. We showed in particular how strong redundant patterns observed in real data might be detected using the notion of internal link and redundancy coefficient defined at the bipartite level. This offers a second interesting perspective for using the bipartite vision.

Further, based on our bipartite model, we proposed and evaluated an Internet topology generator. It relies on bounded discrete power law distributions allowing to reproduce sequences of degree in order to generate a large diversity of graphs. Our dataset is freely available (http://svnet.u-strasbg.fr/merlin) as well as our topology generator (https://code.google.com/p/py-bipartite/).

Finally, we also envisioned possible future research directions able to capture redundant patterns and correlations in L3 and L2–L3 degrees. Those directions seem promising and future work should reveal how they can improve the contributions presented in this paper.

Acknowledgement

References

[1] H. Haddadi, G. Iannaccone, A. Moore, R. Mortier, M. Rio, Network topologies: inference, modeling and generation, IEEE Communications Surveys and Tutorials 10 (2) (2008) 48–69.

[2] R. Pastor-Satorras, A. Vespignani, Evolution and Structure of the Internet: A Statistical Physics Approach, Cambridge University Press, 2004.

[3] P. Erdos, A. Renyi, On random graphs, Publicationes Mathematicae Debrecen 6 (1959) 290–297.

[4] D. Alderson, L. Li, W. Willinger, J.C. Doyle, Understanding Internet topology: principles, models and validation, IEEE/ACM Transactions on Networking 13 (6) (2005) 1205–1218.

[5] X. Wang, D. Loguinov, Understanding and modeling the Internet topology: economics and evolution perspective, IEEE/ACM Transactions on Networking 18 (1) (2010) 257–270.

[6] P. Mérindol, B. Donnet, O. Bonaventure, J.-J. Pansiot, On the impact of layer-2 on node degree distribution, in: Proc. ACM/USENIX Internet Measurement Conference (IMC), 2010.

[7] C. Magnien, F. Ouedraogo, G. Valadon, M. Latapy, Fast dynamics in Internet topology: observations and first explanations, in: Proc. 4th International Conference on Internet Monitoring and Protection (ICIMP), 2009.

[8] H. Haddadi, S. Uhlig, A. Moore, R. Mortier, M. Rio, Modeling Internet topology dynamics, ACM SIGCOMM Computer Communication Review 38 (2) (2008) 65–68.

[9] B. Donnet, T. Friedman, Internet topology discovery: a survey, IEEE Communications Surveys and Tutorials 9 (4) (2007) 2–15.

[10] D.J. Watts, S.H.S.H. Strogatz, Collective dynamics of small-world networks, Nature 393 (1998) 440–442.

[11] M. Newman, Scientific collaboration networks. network construction and fundamental results, Physical Review E 64 (1) (2001) 016131.

[12] A. Iamnitchi, R. Matei, I. Foster, Small world file-sharing communities, in: Proc. IEEE INFOCOM, 2004.

[13] P. Mérindol, V. Van den Schriek, B. Donnet, O. Bonaventure, J.-J. Pansiot, Quantifying ASes multiconnectivity using multicast information, in: Proc. ACM/USENIX Internet Measurement Conference (IMC), 2009.

[14] J.-L. Guillaume, M. Latapy, Bipartite graphs as models of complex networks, Physica A 371 (2) (2006) 795–813.

[15] J. Moy, OSPF version 2, RFC 2328, Internet Engineering Task Force, April 1998.

[16] W. Willinger, D. Alderson, J.C. Doyle, Mathematics and the Internet: a source of enormous confusion and great potential, Notices of the American Mathematical Society 56 (5) (2009) 586–599.

[17] W. Aiello, F. Chung, L. Lu, A random graph model for massive graphs, in: Proc. ACM Symposium on Theory of Computing (STOC), 2000.

[18] M.E.J. Newman, D.J. Watts, S.H. Strogatz, Random graphs with arbitrary degree distribution and their applications, Physical Review E. 64 (2001) 026118.

[19] X. Xu, F. Liu, A novel configuration model for random graphs with given degree sequence, Chinese Physics 16 (2) (2007) 283–287.

[20] E.W. Zegura, K.L. Calvert, S. Bhattacharjee, How to model an internetwork, in: Proc. IEEE INFOCOM, 1996.

[21] E. Zegura, K. Calvert, M. Donahoo, A quantitative comparison of graph-based models for internetworks, IEEE/ACM Transactions on Networking 5 (6) (1997) 770–783.

[22] A.L. Barabási, R. Albert, Emergence of scaling in random networks, Science 286 (5439) (1999) 509–512.

[23] R. Albert, A.L. Barabàsi, Topology of evolving networks: local events and universality, Physical Review Letters 85 (24) (2000) 5234–5237.

[24] B.M. Waxman, Routing of multipoint connections, IEEE Journal on Selected Areas in Communications 6 (9) (1988) 1617–1622.

[25] S. Deering, Host Extensions for IP Multicasting, RFC 1112, Internet Engineering Task Force, August 1989.

[26] T. Pusateri, Distance Vector Multicast Routing Protocol Version 3 (DVMRP), Internet Draft (Work in Progress) draft-ietf-idmr-dvmrp-v3-11, Internet Engineering Task Force, October 2003.

[27] W. Fenner, Internet Group Management Protocol (IGMP), Version 2, RFC 2236, Internet Engineering Task Force, November 1997.

[28] J.-J. Pansiot, `mrinfo` Dataset. <http://svnet.u-strasbg.fr/mrinfo/>.

[29] IANA, Special-use IPv4 Addresses, RFC 3330, Internet Engineering Task Force, September 2002.

[30] M. Gunes, K. Sarac, Inferring subnets in router-level topology collection studies, in: Proc. ACM/USENIX Internet Measurement Conference (IMC), 2007.

[31] O. Sokolova, A. Yurgenson, Using graph, hypergraph, and hypernet models for network analysis problems, in: Proc. 7th International Forum on Strategic Technology (IFOST), 2012.

[32] D. Lun, M. Médard, R. Koetter, Efficient operation of wireless packet networks using network coding, in: Proc. International Workshop on Convergent Technologies, 2005.

[33] T. Schank, D. Wagner, Approximating clustering coefficient and transitivity, Journal of Graph Algorithms and Applications (JGAA) 9 (2) (2005) 265–275.

[34] T. Schank, D. Wagner, Finding, counting and listing all triangles in large graphs, an experimental study, in: Proc. International Workshop on Experimental and Efficient Algorithms (WEA), 2005.

[35] M. Latapy, C. Magnien, N. Del Vecchio, Basic notions for the analysis of large two-mode networks, Social Networks 30 (1) (2008) 31–48.

[36] M.E.J. Newman, Random graphs with clustering, Physical Review Letters 103 (5) (2009) 058701.

[37] O. Allali, L. Tabourier, C. Magnien, M. Latapy, Internal links and pairs as a new tool for the analysis of bipartite complex networks, Social Network Analysis and Mining 3 (1) (2013) 85–91.

[38] M. Latapy, T. Phan, C. Crespelle, T. Nguyen, Termination of multipartite graph series arising from complex network modeling, in: Proc. International Conference on Combinatorial Optimization and Applications (COCOA), 2010.

[39] G.K. Zipf, Human Behavior and the Principle of Least-Effort: An Introduction to Human Ecology, Addison-Wesley, Cambridge, MA, 1949.

[40] A. Clauset, C.R. Shalizi, M.E.J. Newman, Power-law distributions in empirical data, SIAM Review 51 (4) (2009) 661–703.

Fabien Tarissan is a permanent associate professor at University Pierre et Marie Curie (Paris 6) since 2009. He obtained his Ph.D. in computer science from University Denis Diderot (Paris 7) in 2006, after which he spent a year as a post-doc in Italy (Bologna) and then two years in a post-doc position at the ISC (Institut des Systémes Complexes, École Polytechnique) in France. He joined the LIP6 in September 2009 and is now one of the permanent researcher of the ComplexNetworks team. His work focuses on complex systems with a strong expertise in the modelling and analysis of complex networks. He worked in particular on inverse problems arising in measuring large scale networks. This led to the development of methods enable to deduce general properties from the partial knowledge of the networks dynamics.

Bruno Quoitin is an assistant professor at UMons in Belgium. He obtained his M.S. degree in computer science from University of Namur in 1999. He then worked in the industry in the field of industrial control networks until 2002. He obtained his Ph.D. degree in applied sciences from UCLouvain in 2006. His main research interests are interdomain routing, network management, interdomain traffic engineering, network modeling and simulation.
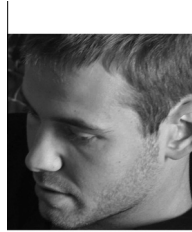
Pascal Mérindol received his Ph.D. degree from the University of Strasbourg (France) in 2008. Then, he spent two years in Belgium at the Université catholique de Louvain as a post-doc researcher. He is now Associate Professor in the Network Research Group of the ICube laboratory (http://icube.unistra.fr/) in Strasbourg. His main research topics are routing and Internet topology discovery.

Benoit Donnet is currently Assistant Professor at the Electrical Engineering and Computer Science Department of the Université de Liège (Liège – Belgium). He received his Master Degree in Computer Science from the Université de Namur (Namur – Belgium) in 2003 and his Ph.D. in Computer Science from the Université Pierre & Marie Curie (Paris – France) in 2006. His research interests are in Internet measurements (measurements scalability, Internet topology discovery), network modeling, new Internet architectures (LISP), Bloom filters, and machine learning applied to Internet measurements.

J.-J. Pansiot received a M.Sc. in Computer Science from Nancy University (France, 1972), a Ph.D. in Computer Science from Cornell University (USA, 1976) and a "Doctorat d'Etat" from the University of Strasbourg (France, 1983). He joined the Department of Computer Science from the same University where he was successively appointed as assistant professor, associate professor and full professor (1984). His research interests include routing, multicasting, traffic engineering and Internet cartography.

Matthieu Latapy is a CNRS senior researcher, head of the Complex Systems department of the LIP6 laboratory, hosted by University Pierre and Marie Curie, Paris, France. He works on all kinds of questions related to complex networks like the internet, the web, peer-to-peer exchanges, social networks, etc. He is particularly interested in the identification of research questions transversal to many of these objects (including metrology, analysis, modeling, and algorithms), and their resolution as well as practical applications. He published numerous papers on these topics in high-quality international journals and conferences in computer science, physics, mathematics and sociology.