institute IMDEA networks

technical report

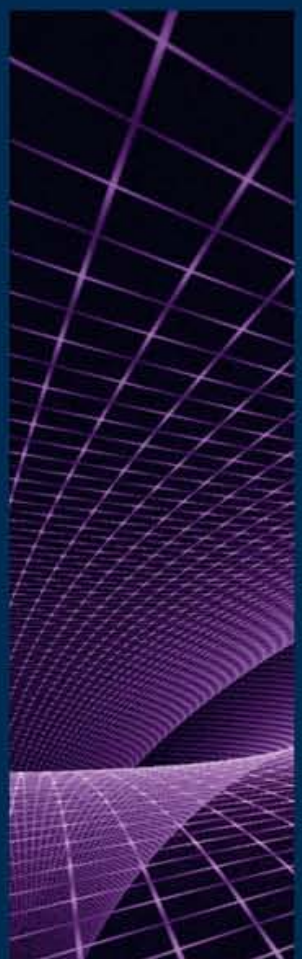
TR-IMDEA-Networks-2011-3

Obscure Giants: Detecting the Provider-Free ASes

Syed Hasan

Sergey Gorinsky

December 2011



Obscure Giants: Detecting the Provider-Free ASes

Syed Hasan* and Sergey Gorinsky

Institute IMDEA Networks
Avenida del Mar Mediterraneo, 22
Leganes, Madrid, 28918, Spain
syed.anwar@imdea.org, sergey.gorinsky@imdea.org

***Abstract*—Internet routing depends on economic relationships between ASes (Autonomous Systems). Despite extensive prior research of these relationships, their characterization remains imprecise. In this paper, we focus on provider-free ASes that reach the entire Internet without paying anyone for the traffic delivery. While the ground truth about PFS (set of the provider-free ASes) lies outside the public domain, we use trustworthy non-verifiable sources as a baseline for result validation. Straightforward extraction of PFS from public datasets of inter-AS economic relationships yields poor results. Then, we develop a more sophisticated Temporal Cone (TC) algorithm that relies on topological statistics (customer cones of ASes) and exploits the temporal diversity of the datasets. Our evaluation shows that the TC algorithm infers PFS from the same public datasets with a significantly higher accuracy. We also assess the sensitivity of the TC algorithm to its parameters.**

I. Introduction

Economic relationships between ASes (Autonomous Systems) matter for Internet routing. For example, it is financially more attractive for an AS to route traffic through a peering link than a transit connection of the AS to its provider. Despite a trend towards flattening [1], the Internet routing ecosystem is essentially hierarchical [1]–[4]. A vast majority of ASes are relatively small and route traffic either as customers of transit links or by peering with local ASes of a similar stature. There exists only a handful of *provider-free ASes* that reach the entire Internet without paying anyone for the traffic delivery. While a tier-1 network is a more common name for a provider-free AS, our paper uses the latter term because prior attempts to redefine AS tiers make network tiering an ambiguous notion. The *set of the provider-free ASes*, to which we refer as *PFS*, contains only large networks. Nevertheless, the real difference between them and another large network can be subtle. For example, if a network is not a provider-free AS because it pays for less than 1% of its inter-domain traffic, the lack of the provider-free status can be obscure to outsiders, especially if the disqualifying payments are for a paid peering relationship which is subject to a non-disclosure agreement.

Due to the general reluctance of ASes to disclose their business agreements, researchers infer the inter-AS economic relationships from BGP (Border Gateway Protocol) [5] route advertisements or actual IP (Internet Protocol) [6] forwarding routes. Such inferences are imperfect, e.g., a router misconfiguration can trigger an inference of an invalid relationship. Also, the inference algorithms are heuristic and can cause additional deviations from the reality. Finally, the economic relationships are dynamic: while it takes time to collect a comprehensive set of measurements, changes in the relationships can decrease the inference accuracy. Over the past decade, numerous research efforts have tried but failed to infer inter-AS economic relationships precisely. In this paper, we explore this failure in the specific context of provider-free ASes.

Our interest in PFS arises due to a number of reasons. First, the provider-free ASes clearly play a key role as the transit core of the Internet ecosystem. By delivering a significant portion of Internet traffic, PFS is highly relevant to the overall resilience of the Internet to accidental failures and intentional disruptions. In particular, economic disputes between provider-free ASes can endanger the universal connectivity of Internet users. Second, while humans prefer to think in discrete categories, designation of an autonomous system as provider-free can have tangible marketplace implications. Third, some algorithms for inter-AS relationship inference use PFS as an input [4], [7] and hence need to know PFS accurately.

This paper contributes by developing an algorithm that detects PFS from public datasets of inter-AS economic relationships. We show that straightforward extraction of PFS from the public datasets yields poor results. Our alternative algorithm utilizes topological statistics (customer cones of ASes) and temporal dataset diversity. The more sophisticated algorithm infers PFS with a significantly higher accuracy. Although a lot of related studies deal with the more general problem of inter-AS relationship inference, our algorithm succeeds by focusing on the more specific problem of PFS detection. Another group of related work redefines tier-1 networks as per a new classification of Internet ASes, e.g., based on their graph-theoretic topological properties. In contrast, our study detects provider-free ASes in accordance to the traditional tier-1 definition. The two main contributions of our paper are in deriving:

- *PFS insights from mostly trustworthy but non-verifiable sources*; we carefully filter out occasional spurious answers;
- *TC (Temporal Cone) algorithm that detects PFS based on public datasets of inter-AS economic relationships*; the derived TC algorithm is useful because it enables accurate

*Syed Hasan is also a doctoral student at Carlos III University of Madrid

inferences of PFS in the future even if PFS insights from the non-verifiable sources become unavailable.

While the knowledge of PFS is highly valuable, validation of PFS inference results constitutes a major challenge because the ground truth lies outside the public domain. To tackle the validation challenge, we utilize trustworthy but non-verifiable sources such as Wikipedia [8]. These sources do not disclose their data and methods. Thus, their conclusions are not purely scientific. Nevertheless, our conversations with network operators indicate that the non-verifiable sources reflect the reality accurately. Whereas it seems practically impossible to obtain the complete ground truth from network operators, the non-verifiable source insights form the best available baseline for result validation in this important domain. As a midpoint between traditional science and citizen science [9], [10], our PFS detection method expands the scope of knowledge but softens the benchmark for validation.

We structure the rest of the paper as follows. Section II reports PFS insights from the non-verifiable sources. Section III describes the public datasets in our study. Section IV considers a straightforward PFS detection method. After analyzing the failures of this straightforward method, Section V develops the more sophisticated TC algorithm. Section VI evaluates the TC algorithm. Section VII comments on related work. Section VIII concludes the paper by summing up its contributions.

II. Non-verifiable sources

While the obscure inter-AS economic relationships do not reveal the ground truth about PFS, a number of non-verifiable sources offer insights into this set. We consider three such non-verifiable sources: Wikipedia, Renesys, and Hurricane Electric.

Wikipedia maintains an article about provider-free ASes [8]. Our primary interest is in the Wikipedia perspectives throughout 2009 because the development of our TC algorithm relies on public datasets collected during that year. According to Wikipedia, PFS consisted of 8 members on 1/1/2009: AT&T, Global Crossing, Level 3, NTT, Qwest, Sprint, Verizon, and Savvis [11]. The article has seen frequent revisions and expanded its PFS with Telia on 28/1/2009 [12]. The addition of Tata on 25/3/2009 resulted in the following PFS [13]:

$W_1 = \{$AT&T, Global Crossing, Level 3, NTT, Qwest, Sprint, Verizon, Savvis, Telia, Tata$\}$.

Except for few incidents in June and October when spurious modifications disappeared shortly after being made, PFS preserved this 10-member composition until the end of 2009. In 2010 and 2011, Wikipedia continued the trend of the PFS expansion and typically recognized Tinet as the 11th member of the PFS, e.g., in the 10/2/2011 revision [14]:

$W_2 = \{$AT&T, Global Crossing, Level 3, NTT, Qwest, Sprint, Verizon, Savvis, Telia, Tata, Tinet$\}$.

Whereas Wikipedia is an online encyclopedia that anyone may edit, some short-lived revisions of this particular article certainly distorted the reality [15]. Nevertheless, experts think that on the whole the Wikipedia perspective reflects PFS accurately [4].

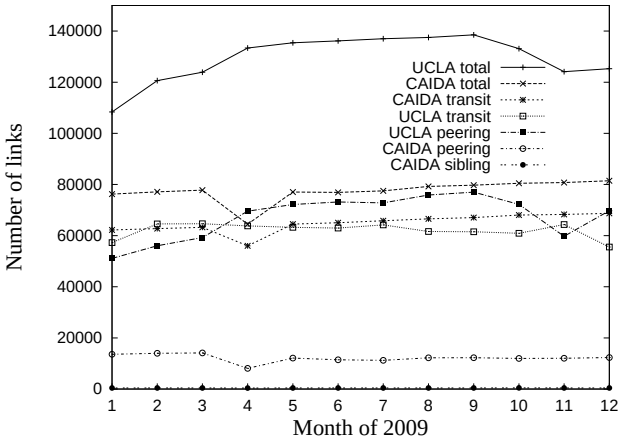


Fig. 1: Inter-AS economic relationships in the UCLA and CAIDA datasets during 2009.

Renesys is a private company that sells Internet business information. In 1/2009, Renesys announced a 12-member set of commercial default-free ASes [16], i.e., ASes that can route traffic to any Internet destination without relying on a default route. Default-free ASes are either provider-free or reaching the entire Internet by paying for peering but not for transit. The Renesys set subsumes W_1 and includes two more ASes: XO and AboveNet. Interestingly, the Wikipedia article explicitly stated in all its revisions that XO and AboveNet were not provider-free due to paid peering [11]–[15]. Thus, the Renesys perspective is consistent with limiting PFS to W_1.

Hurricane Electric is an Internet service provider that offers an online tool for ranking the peers of an autonomous system [17]. The specific criteria for the ranking are not clear but seem to rely on the number of active BGP connections for the AS or the percentage of BGP paths transiting the AS. For each AS in W_1, all the other ASes in W_1 are among highly ranked peers of this AS according to the Hurricane Electric tool. Thus, the ASes of W_1 do form a close-knit peering community as expected for provider-free ASes.

Based on the above considerations, our paper subsequently treats W_1 as the primary PFS answer from the non-verifiable sources for 2009.

III. Public datasets

PFS insights in Section II came from the non-verifiable sources that did not disclose their data and methods. The rest of our study explores datasets from two public sources: UCLA (University of California, Los Angeles) [18] and CAIDA (Cooperative Association for Internet Data Analysis) [19]. The datasets from both public sources characterize the economic relationships between Internet ASes. UCLA classifies inter-AS links as transit or peering. CAIDA uses an additional category for sibling relationships: a sibling link connects two ASes belonging to the same Internet service provider.

Month of 2009	1	2	3	4	5	6	7	8	9	10	11	12
UCLA	8 (1)	6 (0)	7 (0)	17 (9)	16 (9)	17 (9)	15 (9)	19 (9)	19 (9)	16 (10)	17 (9)	18 (9)
CAIDA	23 (6)	26 (6)	26 (6)	29 (7)	30 (7)	27 (7)	28 (7)	29 (7)	25 (6)	26 (6)	27 (6)	27 (6)

TABLE I: PFS size according to the straightforward method for the UCLA and CAIDA datasets and (in parentheses) number of ASes from W_1 in this PFS.

Both UCLA and CAIDA leverage BGP measurements but employ different methods to infer the economic relationships from the BGP data. The UCLA method utilizes the Route Views [20] and RIPE RIS (Réseaux IP Européens Routing Information Service) [21] measurement infrastructures where route collectors engage via BGP with routers in strategic Internet locations to collect AS-level path announcements. The routers that supply the announcements are called BGP monitors. UCLA collects the announcements from BGP monitors located in provider-free autonomous systems (as identified by Wikipedia). The UCLA method categorizes the collected inter-AS relationships into peering and transit based on the valley-free routing conditions [22] and depending on how consistent the views from different monitors are.

CAIDA looks up the IRR (Internet Routing Registry) database [23] to detect sibling links: if two linked ASes belong to the same organization as per the database, the CAIDA method classifies the relationship between the autonomous systems as a sibling link. To infer peering and transit links, CAIDA relies on BGP measurements from Route Views. The CAIDA heuristic for identifying and directing the transit links strikes a balance between maximizing the following two metrics: (1) number of BGP paths that are valid according to the valley-free routing rules and (2) number of links where the provider node of the link has a higher degree than the customer node of the link (the degree of a node refers to the number of links between this node and other ASes). With the CAIDA method, peering relationships are links between nodes with similar degrees.

While the UCLA datasets are available starting from 10/2008, CAIDA reports its datasets infrequently for 2009 and only twice after 2009. During the development of our PFS detection algorithm, it would be desirable to have similar time series for the two sources. Hence, our Sections IV and V focus on the 12 months of 2009. Guided by the CAIDA dataset availability and picking one day per month, we select the following days for both sources: 22/1, 20/2, 11/3, 29/4, 20/5, 15/6, 20/7, 30/8, 20/9, 20/10, 20/11, and 15/12. June is the only exception: because the number of links in the UCLA 15/6 dataset is extremely low, we use 16/6 instead for UCLA.

Figure 1 depicts the inter-AS economic relationships in the UCLA and CAIDA datasets during 2009. For either source, the total number of links tends to grow with time, and the few down-and-up swings are most likely due to imperfect measurements rather than actual fluctuations in the number of economic relationships. The sibling relationships in the CAIDA datasets constitute a negligible fraction of the overall link population. While the number of peering links is much higher for UCLA than for CAIDA, the number of transit links is rather similar for the two sources. The transit-link profiles are mostly consistent but do have some aberrations such as the dip for CAIDA in 4/2009. The numbers of transit links for the two sources remain most stable and close to each other between 5/2009 and 7/2009.

When evaluating our TC algorithm in Section VI, we utilize the UCLA datasets for all 32 months of their availability from 10/2008 to 5/2011. We select the 20th day for all 20 additional months except 5/2011, for which we use 10/5/2011 as the last day of our data gathering.

IV. Straightforward inference

Given a dataset of inter-AS economic relationships, one might hope to infer PFS using the following *straightforward method: compose PFS from all such ASes in the dataset that have no transit provider*. We apply this straightforward method to the UCLA and CAIDA datasets of Section III. Table I sums up the generally disappointing results for all 12 months of 2009. Throughout the year, the straightforward method includes into its PFS up to 23 non-W_1 ASes and excludes up to all 10 ASes of W_1. For the UCLA and CAIDA datasets from 6/2009 (when the numbers of transit links for the two sources remain most stable and close to each other), PFS contains respectively 17 and 27 ASes, with respectively 9 and 7 of these ASes belonging to W_1.

For the UCLA 6/2009 dataset, the straightforward method excludes Tata from PFS because NTT and GIT Telecom (a Cypriot AS) are transit providers for this missing member of W_1 according to the dataset. Among the 8 non-W_1 members of PFS, Sunkist Growers (a not-for-profit cooperative of citrus growers in California and Arizona), Open Peering Initiative (a public peering IXP in Amsterdam), and Siemens seem highly unlikely to be genuine provider-free ASes. These 3 ASes do have providers in the CAIDA dataset from the same month.

For the CAIDA 6/2009 dataset, the straightforward method omits NTT, Savvis, and Tata from PFS because these 3 members of W_1 have transit providers. Specifically, NTT has 3 providers: Verizon, Telia, and Easynet. Savvis has 5 providers: Telia, Tata, Tinet, XO, and Deutsche Telekom. Although Tata is a transit provider for Savvis, the straightforward method does not recognize Tata as a provider-free AS either: Tata appears as a customer of NTT, Telia, and Tinet. On the other hand, PFS of the straightforward method includes 20 non-W_1 ASes such as the University of Texas System, NASA, and New Zealand Research Network, which do have providers in the UCLA 6/2009 dataset.

Link misclassification in the datasets is the most common source of errors for the straightforward method. The UCLA

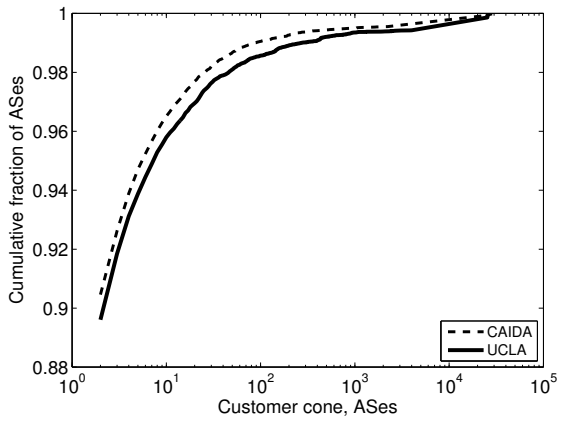


Fig. 2: 6/2009 distributions of the AS customer cones.

Rank	AS name (AS number)	Customer cone, ASes	In W_1?
1	Sprint (1239)	28478	✓(1)
2	Level3 (3356)	28168	✓(2)
3	NTT (2914)	27650	✓(3)
4	AT&T (7018)	27613	✓(4)
5	Global Crossing (3549)	27236	✓(5)
6	Verizon (701)	27121	✓(6)
7	Telia (1299)	26833	✓(7)
8	Qwest (209)	26764	✓(8)
9	Deutsche Telekom (3320)	26263	–
10	Ipercast (34763)	26127	–
11	Savvis (3561)	26082	✓(9)
12	GIT Telecom (38925)	26015	–
13	Tata (6453)	26014	✓(10)

TABLE II: UCLA customer-cone ranks of ASes for 6/2009.

and CAIDA datasets are typical in this regard. We applied the straightforward method to another dataset inferred with Gao's algorithm [24], and the respective results suffer from the link misclassification as well.

V. TC ALGORITHM

Section IV demonstrates that the straightforward inference yields disappointing PFS results with respect to both false positives and false negatives. Two factors undermine the straightforward method. First, while the UCLA and CAIDA datasets do not classify the inter-AS links fully and correctly, even a single error in the input dataset can mislead the straightforward method. The method can exclude a genuine provider-free AS (e.g., Tata in the UCLA 6/2009 dataset) from PFS because the dataset mistakenly reports a provider for this AS. Also, the method can wrongly include an AS (e.g., Sunkist Growers) into PFS because the dataset misses the transit link between this AS and its provider. Second, the straightforward method implicitly assumes that having no provider implies the ability to reach the entire Internet. In reality, some ASes in the Internet ecosystem do not strive for the universal reachability. For example, the main goal of an IXP (Internet eXchange Point) [1], [25] is to serve as a peering infrastructure that enables other ASes to exchange their local traffic. The straightforward method can incorrectly classify an IXP (e.g., Open Peering Initiative) as a provider-free AS.

Thus, we develop a more sophisticated TC (Temporal Cone) algorithm for detecting PFS. Sections V-A, V-B, and V-C discuss the three important components of our algorithm: its use of topological statistics to deal with the noisy data, setting the PFS size, and exploiting the temporal diversity of the datasets to improve the accuracy of the PFS detection further.

A. Customer-cone ranking

Topological statistics represent a promising basis for accurate PFS detection because of their potential resilience to individual errors caused by the link misclassification. While the datasets of inferred inter-AS relationships clearly contain numerous errors, our approach relies on the premise that the datasets are also rich in correct information and that looking at the datasets from a right perspective can reveal PFS accurately.

After examining a number of options, we choose the *customer cone* as the topological parameter for the TC algorithm: the customer cone of an AS includes the AS itself as well as all direct and indirect customers of the AS, i.e., every customer reachable from the AS through a sequence of provider-to-customer transit links [26]. We expect the customer cones of the provider-free ASes to be among the largest because the customer cone of an AS is strictly larger than the customer cone of any of its customers. This expectation is certainly a heuristic (in principle, a provider-free AS can have a smaller customer cone than a network that lies outside this customer cone and has a provider) but our results confirm its effectiveness. Due to multihoming [27] which is common throughout the Internet ecosystem, the customer cones of two ASes can overlap. We compute the customer cone of each AS using a recursive algorithm that takes the overlaps of the customer cones into account.

To illustrate the potential of the customer cone for PFS detection, let us revisit the false negatives and false positives of the straightforward method for the 6/2009 datasets in Section IV. For the UCLA 6/2009 dataset, the straightforward method computes the PFS that incorrectly excludes Tata and wrongly includes Sunkist Growers, Open Peering Initiative, and Siemens. The customer cones of Tata, Sunkist Growers, Open Peering Initiative, and Siemens are 26014, 69, 75, and 8 ASes respectively. While the customer cone of 26014 ASes is the 13th largest among all networks in the dataset, the customer-cone perspective leaves Tata as a plausible candidate for PFS. On the other hand, the small customer cones of Sunkist Growers, Open Peering Initiative, and Siemens clearly suggest that these 3 networks are not provider-free ASes. Similarly, for the CAIDA 6/2009 dataset, the 3 false negatives of the straightforward method are NTT, Savvis, and Tata which have very large customer cones of 24473, 23769, and 23788

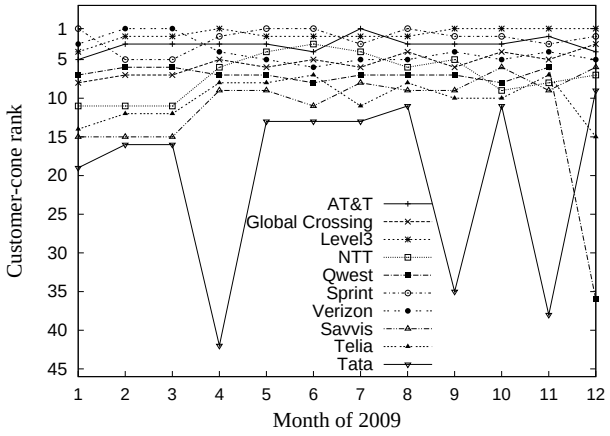


Fig. 3: UCLA customer-cone ranks of the ASes in W_1.

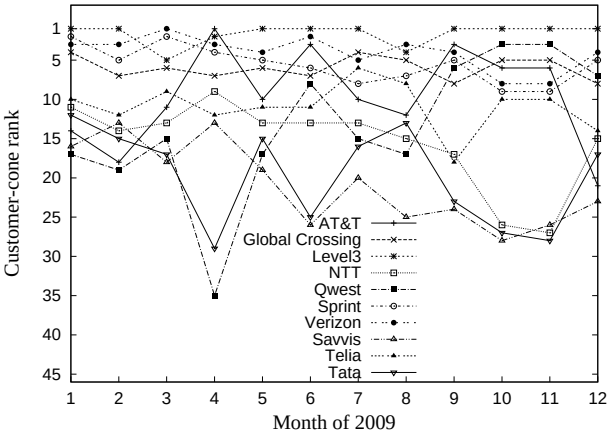


Fig. 4: CAIDA customer-cone ranks of the ASes in W_1.

ASes respectively. The University of Texas System, NASA, and New Zealand Research Network are false positives of the straightforward method, and their small corresponding customer cones of 19, 11, and 232 ASes strongly indicate that these 3 networks are not provider-free. The above examples confirm that the customer-cone metric is more robust to the link misclassification than the simple inspection of the link types as with the straightforward method.

Among alternative topological parameters that we considered as a basis for the TC algorithm, the customer count of an AS is easier to compute than the customer cone and refers to the number of direct customers of the AS. A very large value of the customer count has some correlation with the provider-free status. However, the correlation is weaker than for the customer cone: even if a network does not belong to PFS due to being a direct customer of a provider-free AS, this network can have a very large number of own direct customers.

While the PFS members peer with each other, another potential approach to detecting PFS is to search for close-knit peering communities, e.g., to examine the number of peering links of each AS. However, our preliminary analyses

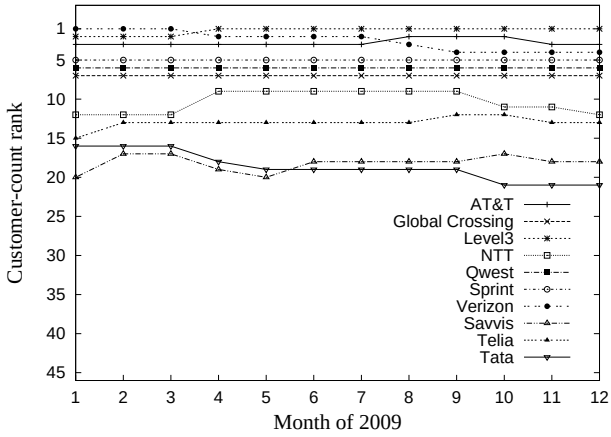


Fig. 5: UCLA customer-count ranks of the ASes in W_1.

for peering-based and other alternative parameters did not yield encouraging results. Consequently, the customer cone serves as the topological basis for our PFS detection algorithm.

Figure 2 plots the distributions of the AS customer cones in the UCLA and CAIDA datasets for 6/2009 and shows that only a tiny fraction of all ASes have a really large customer cone. Table II zooms in on the tail of the UCLA 6/2009 distribution. The tail covers set W_1 quite tightly: all 10 members of W_1 appear among the top 13 ASes ranked by the customer cone; this is an improvement over the straightforward method which includes only 9 members of W_1 into its 17-member PFS for 6/2009.

Figure 3 tracks the UCLA customer-cone ranks of all ASes in W_1 throughout 2009. The ranks remain close to the top 10 with few exceptions such as three dramatic dips for Tata. While Figure 3 corroborates the promising potential of the customer-cone statistics for PFS detection, the results also suggest that our algorithm needs additional features for overcoming the noise in the datasets.

Figure 4 depicts the CAIDA customer-cone ranks of all ASes in W_1 during 2009. In agreement with Table I, the customer-cone results in Figures 3 and 4 imply that the UCLA datasets are less noisy and thus more suitable for PFS detection than the CAIDA datasets.

Figure 5 presents the 2009 customer-count ranks of all ASes in W_1 for the UCLA datasets. Compared to Figure 3 for the customer-cone ranks, Figure 5 shows that the customer-count ranks are less effective in capturing W_1.

B. PFS size

To detect PFS, the TC algorithm has to size this set. Whereas the Internet is growing, our hypothesis is that the set of provider-free ASes scales up proportionally with the overall population of Internet ASes. More specifically, we set size S_m of PFS at time m to:

$$S_m = \lfloor k \cdot P_m \rfloor \quad (1)$$

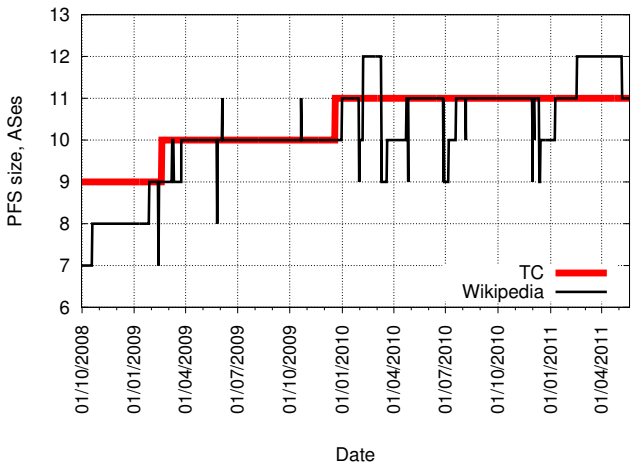


Fig. 6: PFS size according to Wikipedia and TC algorithm for the UCLA datasets.

where P_m represents the total number of Internet ASes at time m, and k is a fixed factor.

To validate the hypothesis and select the value of k, we explore how PFS evolved from 10/2008 to 5/2011 according to the Wikipedia perspective. During this time interval, the article has been revised on 113 days, and multiple revisions on a single day were common. Figure 6 depicts the PFS size according to Wikipedia, with short-lived spikes representing spurious revisions. For every day throughout the 32-month interval, Figure 6 also plots the PFS size as per Equation 1 with the value of k set to 0.00032, which corresponds to 1 in about 3000 Internet ASes being provider-free. This equation-based prediction is aligned well with the PFS growth trend in the depicted Wikipedia data. Whereas the amount of the available data is too limited to recommend strongly the specific value of k or even to defend confidently the proportionality of PFS to the overall population of Internet ASes, the available data do suggest that Equation 1 offers a reasonable approximation for the PFS size.

C. Temporal dimension

With the PFS size selected, the algorithm still needs to identify the ASes of the set. We utilize the temporal dimension of the datasets to tackle the noise remaining in the customer-cone statistics. Our intuition is that the membership of an AS in PFS is relatively stable. While a new AS can join PFS and subsequently lose the provider-free status again, such transitions are infrequent, caused by rare mergers/acquisitions and guarded against by long-term business contracts. Therefore, to decide whether an AS is provider-free for month m, our algorithm looks w months back and ahead from month m and includes the AS into PFS for month m only if the AS belongs to the set according to the customer-cone ranks for at least n out of these $2w + 1$ months. For an input with M months in the time series, our algorithm outputs PFS for each month except for the first w and last w months, i.e., the algorithm computes PFS for the $M - 2w$ middle months.

Notation	Semantics
m or i	month
M	number of months in the time series
C_m	list of the Internet ASes ordered by their customer-cone ranks for month m
L_m	ordered list of PFS candidates for month m
S_m	size of PFS for month m
w	lookback/lookahead window
F_m	PFS for month m
a	AS
b_a	counter of months when AS a belongs to PFS as per the customer-cone rankings
$r_{a,m}$	rank of a in L_m
n	PFS membership threshold

TABLE III: Notation for our algorithm in Figure 7.

for $m = 1, \ldots, M$
  compute C_m;
  $L_m \leftarrow C_m$;
  calculate S_m according to Equation 1;
for $m = M - w, \ldots, w + 1$
  $F_m \leftarrow \emptyset$;
  $a \leftarrow$ first AS in L_m;
  while $|F_m| < S_m$ and $a \neq$ null
    $b_a \leftarrow 0$;
    for $i = m - w, \ldots, m + w$
      if $r_{a,i} \leq S_i$
        then $b_a \leftarrow b_a + 1$;
    if $b_a \geq n$
      then $F_m \leftarrow F_m \cup \{a\}$
      else remove a from L_m; $r_{a,m} \leftarrow \infty$;
    $a \leftarrow$ next AS in L_m

Fig. 7: TC (Temporal Cone) algorithm for PFS detection.

While one-year contracts between ASes are common, we recommend $w = 6$ months and $n = 5$ months as default values for the w and n parameters of the algorithm, i.e., inclusion of an AS into PFS requires from the customer-cone ranks to endorse the AS for at least 5 out of 13 months. These settings enable our algorithm to recognize a genuine one-year PFS membership in spite of multiple months of erroneous disqualifications by the customer-cone ranks. These settings also allow the algorithm to exclude a non-provider-free AS from PFS despite multiple months of mistaken customer-cone endorsements. In Section VI, we study sensitivity of the TC algorithm to the w and n parameters and show that $w = 6$ months and $n = 5$ months are reasonable settings.

We refer to the developed PFS detection algorithm as TC (Temporal Cone). Table III explains the notation used in Figure 7 that presents our TC algorithm in detail.

VI. Evaluation

According to Sections III through V, the datasets from UCLA are available for more months and less noisy than

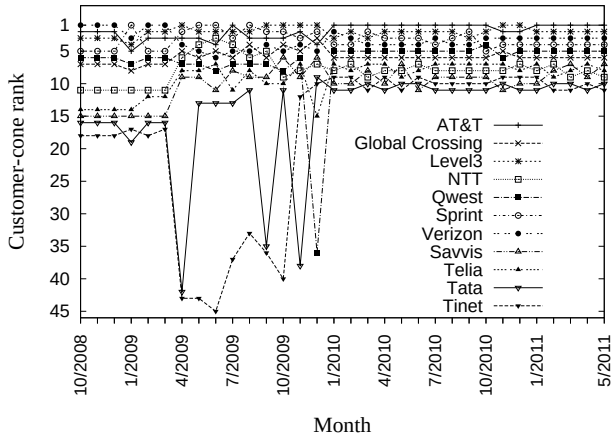


Fig. 8: UCLA customer-cone ranks of the ASes in W_2 (i.e., the W_1 members and Tinet) from 10/2008 to 5/2011.

the CAIDA datasets. To evaluate the developed TC algorithm, Section VI-A relies on the UCLA datasets for the 32 months from 10/2008 to 5/2011 and – following the recommendations from the previous section – sets the PFS sizing factor, lookback/lookahead window, and PFS membership threshold to $k = 0.00032$, $w = 6$ months, and $n = 5$ months respectively. Then, Section VI-B examines the sensitivity of the TC algorithm performance to the w and n parameters.

A. TC results

During its first iterative stage, the TC algorithm determines the AS customer-cone ranks and PFS sizes for all $M = 32$ months. Figure 8 plots the customer-cone ranks of the ASes in set W_2, i.e., all W_1 members and Tinet which became the 11th member of PFS according to Wikipedia after 2009. All 11 members of W_2 consistently appear among the top 11 ASes ranked by the customer cone in 2010 and 2011, indicating a higher accuracy of the more recent UCLA datasets.

As shown in Figure 6, the TC algorithm sizes PFS to 9 ASes between 10/2008 and 1/2009, 10 ASes between 2/2009 and 12/2009, and 11 ASes from 1/2010 to 5/2011. This expansion is consistent with the PFS insights from the trustworthy but non-verifiable sources.

With $w = 6$ months to look back and ahead, the TC algorithm executes its second stage to compute PFS for the $M - 2w = 20$ middle months from 4/2009 to 11/2010. Among the 9 months of 2009 (when the PFS size is 10 ASes), PFS perfectly matches W_1 for one month, omits only Qwest for another month, and excludes only Tata for the other 7 months. For all 11 months of 2010 (when the PFS size is equal to 11 ASes), PFS matches W_2 exactly.

Table IV sums up the performance of the TC algorithm. A quick comparison of these results with Table I reveals that the TC algorithm detects PFS significantly better than the straightforward method.

While the TC algorithm agrees with the Wikipedia perspective on the PFS size, the false positives of the algorithm are

Year	2009		2010
Month	4-8, 10-12	9	1-11
UCLA	10 (9)	10 (10)	11 (11)

TABLE IV: Size of PFS according to the TC algorithm for the UCLA datasets and (in parentheses) number of ASes in this PFS that match the Wikipedia insights (W_1 for 2009 and W_2 for 2010).

equal in number to its false negatives. Hence, we further quantify the performance of the TC algorithm with the following 2 metrics:

- *Accuracy* A_m of the PFS detection for month m is the fraction of ASes in the computed PFS that are provider-free during month m according to Wikipedia;
- *Average accuracy* of the PFS detection is the average of monthly accuracies A_m over all the $M - 2w$ middle months in the input time series.

For the TC results in Table IV, the accuracy of the PFS detection is 90% for 8 months and perfect 100% for the other 12 months. Thus, the corresponding average accuracy of the PFS detection is 96%.

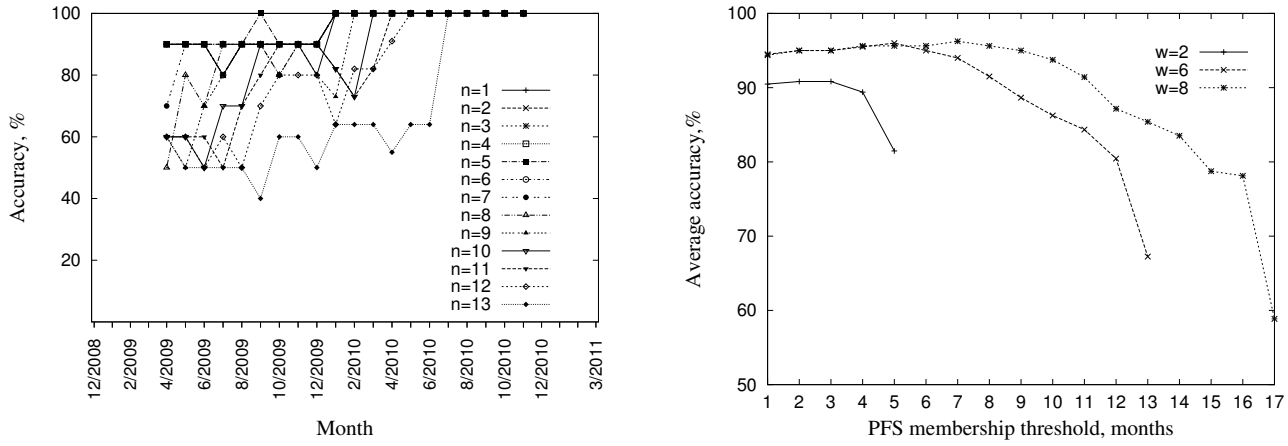
B. Parameter sensitivity

Whereas our TC algorithm relies on parameters w and n, this section studies the sensitivity of the algorithm performance to these 2 parameters. We conduct such study for not only UCLA but also CAIDA. Throughout the study, we use $k = 0.00032$ as discussed in Section V-B.

For the UCLA datasets, Figure 9a depicts the sensitivity of the TC algorithm accuracy to PFS membership threshold n with $w = 6$ lookback/lookahead months. Any of the examined n values delivers 100% accuracy for the last few months. For earlier months, the accuracy is lower and varies from one value of n to another. With $n = 5$ months, the accuracy is most stable and remains at least 90%. With $n = 13$ months, the accuracy is only 40% for 9/2009. In general, the results indicate that values of n in the lower portion of its range are more attractive than values in the upper portion.

By averaging the accuracy over the individual months, Figure 9b exposes more clearly the trend revealed in Figure 9a. With $w = 6$ months, the average accuracy of the TC algorithm declines steadily and dramatically as PFS membership threshold n grows beyond 5 months. When n decreases from 5 months to 1 month, the average accuracy declines slightly. Hence, for $w = 6$ months, the average accuracy attains its peak of 96% when n is set to 5 months. Figure 9b also plots the average accuracy for $w = 2$ months and $w = 8$ months, with the profile of the accuracy sensitivity to n remaining qualitatively the same. The average accuracy is stable for smaller values of the PFS membership threshold but decreases consistently and significantly after n grows beyond a tipping point.

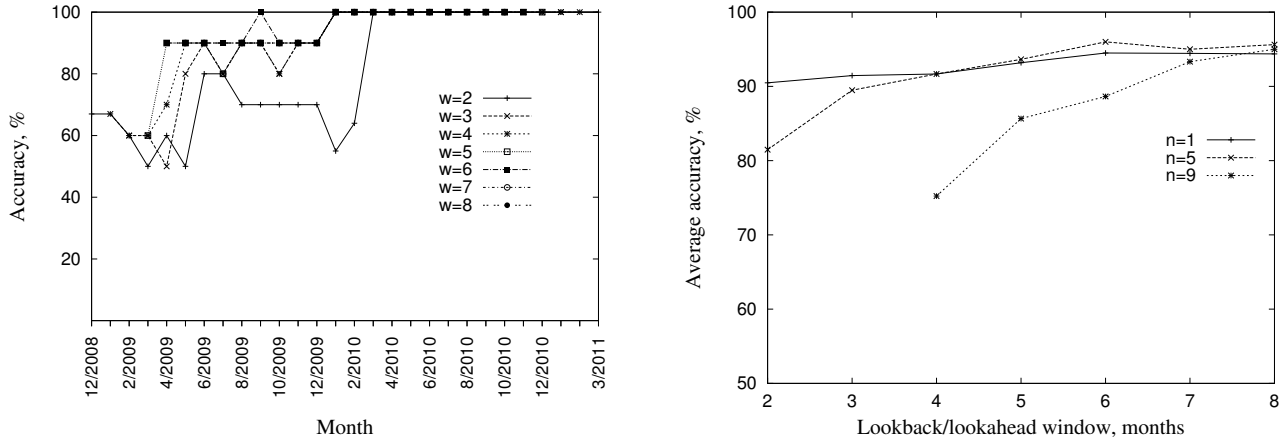
Figure 10 shows the sensitivity of the TC algorithm accuracy to lookback/lookahead window w for the UCLA datasets. For $n = 5$ months, Figure 10a presents the accuracy for



(a) Accuracy with $w = 6$ months

(b) Average accuracy

Fig. 9: Sensitivity of the TC algorithm accuracy to PFS membership threshold n for the UCLA datasets.



(a) Accuracy with $n = 5$ months

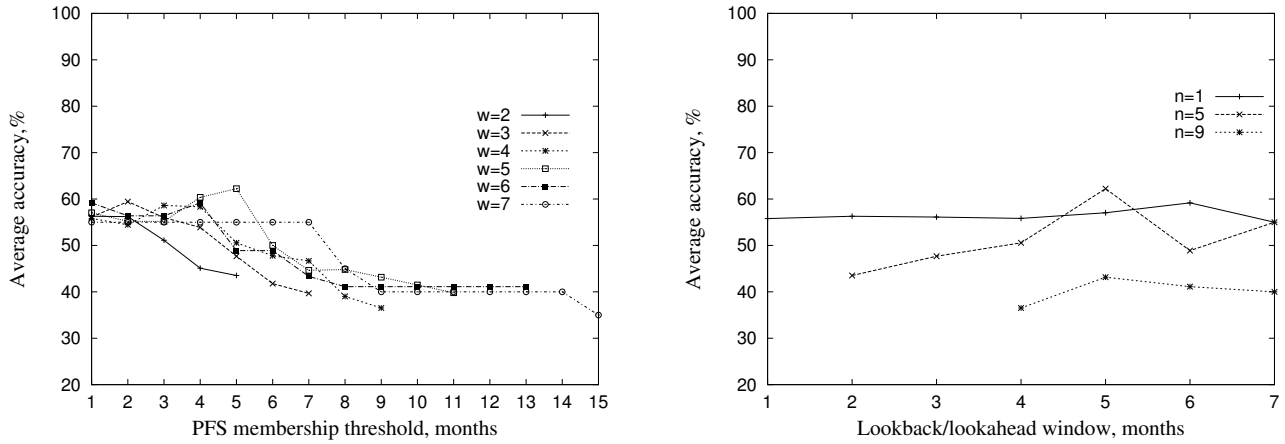
(b) Average accuracy

Fig. 10: Sensitivity of the TC algorithm accuracy to lookback/lookahead window w for the UCLA datasets.

individual months and suggests that larger values of w are generally beneficial. Figure 10b reveals this dependence more clearly. As w grows, the average accuracy increases first but then tends to flatten out. With $n = 5$ months, the average accuracy reaches the maximum of 96% when w is set to 6 months. $w = 7$ months and $w = 8$ months yield similarly high values of the average accuracy. Based on the above observations, we conclude that $w = 6$ months and $n = 5$ months constitute reasonable settings of the two parameters for the UCLA datasets.

For the CAIDA datasets, we conduct a similar sensitivity study and report the results in Figure 11. The study relies on data for the 16 months from 10/2008 to 1/2010. The shorter duration of the CAIDA time series reduces the meaningful value range for w to be up to 7 months. Figure 11a plots the average accuracy of the PFS detection as a function of n. The dependence is qualitatively the same as with the UCLA datasets. When the the PFS membership threshold increases, the average accuracy remains rather stable first but then declines steadily and substantially after a tipping point. The average accuracy peaks at 62% with $w = 5$ months and $n = 5$ months. The qualitative profile for the sensitivity of the TC algorithm accuracy to lookback/lookahead window w is also similar to the pattern observed for UCLA. As w increases, the average accuracy improves first but then stays mostly stable. Note that the best CAIDA settings of $w = 5$ months and $n = 5$ months are close to the settings recommended above for the UCLA data source.

Although the sensitivity of the PFS detection accuracy to the w and n parameters has a qualitatively similar profile for the UCLA and CAIDA datasets, quantitatively the TC algorithm performs very differently with the 2 sources. In particular, the average accuracy peaks at 96% and 62% for UCLA and CAIDA respectively. The performance differences could be partly attributed to the differences in the UCLA and CAIDA inference methodologies. For example, UCLA might be yielding the more accurate results because of considering the RIPE RIS BGP measurements in addition to the Route



(a) Sensitivity to PFS membership threshold n

(b) Sensitivity to lookback/lookahead window w

Fig. 11: Sensitivity of the TC algorithm accuracy to PFS membership threshold n and lookback/lookahead window w for the CAIDA datasets.

Views BGP measurements. Whereas the CAIDA method accounts for node degrees, the UCLA methods disregards them to focus on valley-free routing. Our results suggest that PFS detection might benefit from disregarding the node degrees. Finally, while UCLA collects its data from BGP monitors located in provider-free autonomous systems as identified by Wikipedia, CAIDA gathers its data from a more diverse group of BGP monitors. The higher precision of the UCLA datasets might be due to utilizing, at least indirectly, the provider-free AS knowledge taken from the non-verifiable source.

VII. Related work

The TC algorithm derives PFS from inter-AS economic relationships. Since the pioneering work by Gao [2], the problem of inter-AS relationship inference has attracted a variety of other heuristic solutions [4], [7], [26], [28]–[33]. While our paper is the first to focus on detecting PFS, previous works used PFS as an input to their inter-AS relationship inference algorithms [4], [7]. PFS also served as a basis for studies of backbone networks and resilience of routing to failures [34]–[36].

Derivation of PFS from public inter-AS relationship datasets is challenging because missing or misclassified links make the datasets noisy. Addressing the problem of hidden links [37]–[41] has a potential for making the results of our TC algorithm even better.

While the TC algorithm exploits the temporal diversity of the inter-AS relationship datasets, prior works explored the temporal dimension for studying other problems such as network graph evolution [42], [43].

In general, Internet AS-level graphs have been studied from numerous perspectives. For example, [44], [45] studied the structure of the AS-level graphs using the k-dense and k-clique community detection algorithms.

The work by Subramanian et al. [29] is the closest in spirit to ours. Among its other contributions, that paper proposed a new hierarchical taxonomy for Internet ASes and developed an algorithm that uses AS customer counts to detect the top-tier ASes of the newly proposed hierarchy. While similar in spirit, our work is very different in its specific goals and methods. In particular, we strive to detect PFS in accordance to the traditional definition of provider-free ASes.

VIII. Conclusion

PFS, or the set of provider-free ASes, is important for the Internet resilience and economics. Albeit the ground truth about PFS is not publicly available, there is a significant interest in knowing PFS. For example, the Wikipedia article on provider-free ASes has been viewed about half a million times during the previous three years. In this paper, we sought to supplement the non-verifiable sources, such as the Wikipedia article, with scientific insights from public datasets of inferred inter-AS economic relationships. In particular, we developed the TC algorithm that sized PFS to a fraction of the overall AS population and determined the PFS members by means of AS customer-cone ranking and temporal dataset diversity. In comparison to the straightforward method for extracting PFS, our TC algorithm detected PFS with a substantially higher precision. We also assessed the sensitivity of the TC algorithm to its parameters. The derived TC algorithm is useful because it enables accurate inferences of PFS in the future even if PFS insights from the non-verifiable sources become unavailable.

Whereas the current insights from the non-verifiable sources appeared trustworthy and were corroborated through conversations with network operators, we used the Wikipedia insights to validate the accuracy of our TC algorithm. Although clearly imperfect, this validation method seemed the best option available currently for scientific studies of PFS. One could see our work as a middle point between traditional science and citizen science: our PFS detection method expanded the scope of knowledge but softened the benchmark for validation. Choosing such trade-off is not a novel feature of our methodology: even the discussed UCLA relationship inference method exhibits this property because of its reliance on the

insights from Wikipedia. In spite of utilizing the non-verifiable information, this trade-off is useful for networking practice due to the scientific component that rises the knowledge above the state-of-the-art level of pure beliefs.

REFERENCES

[1] A. Dhamdhere and C. Dovrolis, "The Internet is Flat: Modeling the Transition from a Transit Hierarchy to a Peering Mesh," in *Proceedings of ACM CoNEXT 2010*.

[2] L. Gao, "On Inferring Autonomous System Relationships in the Internet," *IEEE/ACM Trans. Netw.*, 2001.

[3] R. T. B. Ma, D. M. Chiu, J. C. S. Lui, V. Misra, and D. Rubenstein, "On Cooperative Settlement Between Content, Transit and Eyeball Internet Service Providers," in *Proceedings of ACM CoNEXT 2008*.

[4] E. Gregori, A. Improta, L. Lenzini, L. Rossi, and L. Sani, "BGP and Inter-AS Economic Relationships," in *Proceedings of IFIP Networking 2011*.

[5] Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)," RFC 1771, 1995.

[6] J. Postel, "Internet Protocol DARPA Internet Program Protocol Specification," RFC 791, 1981.

[7] J. Xia and L. Gao, "On the Evaluation of AS Relationship Inferences," in *Proceedings of IEEE GLOBECOM 2004*.

[8] Wikipedia, "Tier 1 Network," http://en.wikipedia.org/wiki/Tier_1_network.

[9] M. Stevens and E. D'Hondt, "Crowdsourcing of Pollution Data using Smartphones," in *Proceedings of Ubiquitous Crowdsourcing 2010.*

[10] N. Maisonneuve, M. Stevens, M. Niessen, P. Hanappe, and L. Steels, "Citizen Noise Pollution Monitoring," in *Proceedings of DG.O 2009*.

[11] Wikipedia, "Tier 1 Network, 1/1/2009 revision," en.wikipedia.org/w/index.php?&oldid=261328396.

[12] Wikipedia, "Tier 1 Network, 28/1/2009 revision," en.wikipedia.org/w/index.php?&oldid=267026426.

[13] Wikipedia, "Tier 1 Network, 25/3/2009 revision," en.wikipedia.org/w/index.php?&oldid=279646779.

[14] Wikipedia, "Tier 1 Network, 10/2/2011 revision," en.wikipedia.org/w/index.php?&oldid=413097463.

[15] Wikipedia, "Tier 1 Network, 5/6/2009 revision," en.wikipedia.org/w/index.php?&oldid=294566542.

[16] M. Brown, C. Hepner, and A. Popescu, "Internet Captivity and the De-peering Menace," www.renesys.com/tech/presentations/pdf/nanog-45-Internet-Peering.pdf, NANOG 45.

[17] Hurricane Electric, "Hurricane Electric BGP Toolkit," http://bgp.he.net.

[18] University of California Los Angeles, "Internet Topology Collection," irl.cs.ucla.edu/topology.

[19] Cooperative Association for Internet Data Analysis, "AS Relationships," www.caida.org/data/active/as-relationships.

[20] University of Oregon Route Views Project, http://www.routeviews.org/.

[21] The RIPE Routing Information Service, http://www.ripe.net/data-tools/stats/ris/routing-information-service.

[22] L. Gao and F. Wang, "The Extent of AS Path Inflation by Routing Policies," in *Proceedings of Globecom 2002*.

[23] Internet Routing Registry, http://www.irr.net/.

[24] "AS Topology Data," http://www.cc.gatech.edu/~amogh/topology.html, Georgia Institute of Technology.

[25] B. Augustin, B. Krishnamurthy, and W. Willinger, "IXPs: Mapped?" in *Proceedings of ACM SIGCOMM 2009*.

[26] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, k. claffy, and G. Riley, "AS Relationships: Inference and Validation," *ACM SIGCOMM CCR, 2007*.

[27] A. Akella, B. Maggs, S. Seshan, A. Shaikh, and R. Sitaraman, "A Measurement-based Analysis of Multihoming," in *Proceedings of ACM SIGCOMM 2003*.

[28] Z. Ge, D. Figueiredo, S. Jaiswal, and L. Gao, "Hierarchical Structure of the Logical Internet Graph," in *Proceedings of SPIE ITCOM 2001*.

[29] L. Subramanian, S. Agarwal, J. Rexford, and R. Katz, "Characterizing the Internet Hierarchy from Multiple Vantage Points," in *Proceedings of IEEE INFOCOM 2002*.

[30] Z. M. Mao, L. Qiu, J. Wang, and Y. Zhang, "On AS-level Path Inference," *Proceedings of ACM SIGMETRICS 2005*.

[31] G. Di Battista, T. Erlebach, A. Hall, M. Patrignani, M. Pizzonia, and T. Schank, "Computing the Types of the Relationships between Autonomous Systems," *IEEE/ACM Trans. Netw.*, 2007.

[32] H. Asai and H. Esaki, "Estimating AS relationships for Application-layer Traffic Optimization," in *Proceedings of ETM 2010*.

[33] H. Asai, H. Esaki, and T. Momose, "A Solution Approach for AS Relationships-aware Overlay Routing," http://tools.ietf.org/html/draft-asai-cross-domain-overlay-01, IETF Internet draft (Informational).

[34] R. Mahajan, M. Zhang, L. Poole, and V. Pai, "Uncovering Performance Differences among Backbone ISPs with Netdiff," in *Proceedings of NSDI 2008*.

[35] J. Wu, Y. Zhang, Z. M. Mao, and K. G. Shin, "Internet Routing Resilience to Failures: Analysis and Implications," in *Proceedings of ACM CoNEXT 2007*.

[36] W. Deng, P. Zhu, N. Xiong, Y. Xiao, and X. Hu, "How Resilient are Individual ASes against AS-level Link Failures?" in *Proceedings of SCNC 2011*.

[37] R. Oliveira, D. Pei, W. Willinger, B. Zhang, and L. Zhang, "The (in)Completeness of the Observed Internet AS-level Structure," *IEEE/ACM Trans. Netw.*, 2010.

[38] M. Roughan, S. J. Tuke, and O. Maennel, "Bigfoot, Sasquatch, the Yeti and other Missing Links: What We Don't Know about the AS Graph," in *Proceedings of ACM IMC 2008*.

[39] K. Chen, D. R. Choffnes, R. Potharaju, Y. Chen, F. E. Bustamante, D. Pei, and Y. Zhao, "Where the Sidewalk Ends: Extending the Internet AS Graph using Traceroutes from P2P users," in *Proceedings of ACM CoNEXT 2009*.

[40] B. Zhang, R. Liu, D. Massey, and L. Zhang, "Collecting the Internet AS-level Topology," *ACM SIGCOMM CCR, 2005*.

[41] Y. He, G. Siganos, M. Faloutsos, and S. Krishnamurthy, "A Systematic Framework for Unearthing the Missing Links: Measurements and Impact," in *Proceedings of NSDI 2007*.

[42] A. Dhamdhere and C. Dovrolis, "Ten Years in the Evolution of the Internet Ecosystem," in *Proceedings of ACM IMC 2008*.

[43] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations," in *Proceedings of ACM SIGKDD 2005*.

[44] E. Gregori, L. Lenzini, and C. Orsini, "K-dense Communities in the Internet AS-level Topology," in *Proceedings of COMSNETS 2011*.

[45] E. Gregori, L. Lenzini, and C. Orsini, "K-clique Communities in the Internet AS-level Topology Graph," in *Proceedings of SIMPLEX 2011*.