Obscure Giants: Detecting the Provider-Free ASes

Syed Hasan, Sergey Gorinsky

Obscure Giants: Detecting the Provider-Free ASes

Syed Hasan[1,2] and Sergey Gorinsky[1]

[1] Institute IMDEA Networks, Spain
[2] Carlos III University of Madrid, 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.

Keywords: Internet economics, transit, provider-free ASes

1 Introduction

Economic relationships between ASes (Autonomous Systems) are relevant 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 [6], the Internet routing ecosystem is essentially hierarchical [6, 8, 11, 14]. 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 it is more common to call a provider-free AS a tier-1 network, our paper uses the former 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 measurements of routing and forwarding. Such inferences are imperfect, as this paper demonstrates for 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 [11, 24] and hence need to know PFS accurately.

This paper contributes 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 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 according to 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 non-verifiable sources* via careful removal of 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 continued detection of PFS even if the trustworthy non-verifiable sources stop supplying data about PFS in the future.

We structure the rest of the paper as follows. Section 2 comments on related work. Section 3 reports PFS insights from the non-verifiable sources. Section 4 describes the public datasets in our study. Section 5 considers a straightforward PFS detection method. After analyzing the failures of this straightforward method, section 6 develops the more sophisticated TC algorithm. Section 7 evaluates the TC algorithm. Section 8 concludes the paper by summing up its contributions.

2 Related work

The TC algorithm derives PFS from inter-AS economic relationships. Since the pioneering work by Gao [8], the problem of inter-AS relationship inference has

attracted a variety of other heuristic solutions [7,9,11,17,24]. 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 [11,24]. PFS also served as a basis for studies of backbone networks and resilience of routing to failures [15,23].

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 [3,16,25] 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 [5,13].

The work by Subramanian et al. [17] 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.

3 Non-verifiable sources

Although 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. *Wikipedia* maintains an article about provider-free ASes [19]. According to Wikipedia, PFS consisted of 9 members on 28/1/2009. The 25/3/2009 revision expanded this PFS to the following set [20]:

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

Except for few incidents in 6/2009 and 10/2009 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 [22]:

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 [21]. Nevertheless, experts think that on the whole the Wikipedia perspective reflects PFS accurately [11]. The data from Renesys and Hurricane Electric also support the Wikipedia perspective [12]. Based on the above considerations, the primary PFS answers from the non-verifiable sources in our paper are W_1 for 2009 and W_2 for 2010.

4 Public datasets

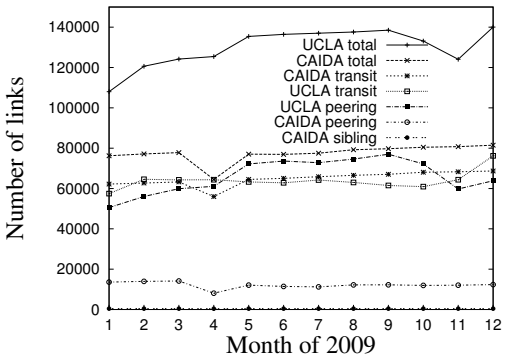


Fig. 1: Inter-AS economic relationships in the UCLA and CAIDA datasets during 2009.

PFS insights in section 3 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) [4]. 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.

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 in sections 5 and 6, we focus on the 12 months of 2009 to have similar time series for the two sources. Figure 1 depicts the inter-AS economic relationships in the UCLA and CAIDA datasets during 2009. When evaluating our TC algorithm in section 7, we utilize the UCLA datasets for all 32 months of their availability from 10/2008 to 5/2011. Our technical report [12] elaborates on the UCLA and CAIDA datasets as well as inference methodology for the datasets.

5 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 4. Table 1 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

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 1: 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.

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.

The most common source of errors for the straightforward method is link misclassification in the UCLA and CAIDA datasets. We also applied the straightforward method to another dataset inferred with Gao's algorithm [10], and the respective results suffer from the link misclassification as well.

6 TC algorithm

Section 5 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) [2, 6] 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 6.1, 6.2, and 6.3 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.

6.1 Customer-cone ranking

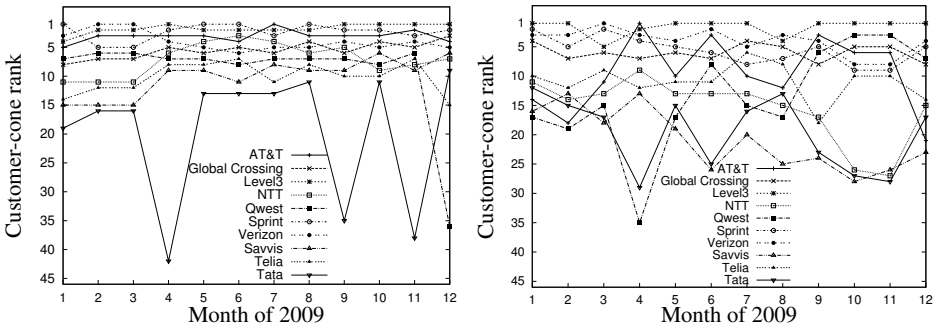
Topological statistics represent a promising basis for accurate PFS detection because of their potential resilience to individual errors caused by the link misclassification. While the datasets of inferred inter-AS relationships clearly contain numerous errors, our approach relies on the premise that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

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

After examining a number of options [12], 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 [7]. 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 [1] 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 5. 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 ASes respectively. The University of Texas System, NASA, and New



(a) UCLA datasets (b) CAIDA datasets

Fig. 2: Customer-cone ranks of the ASes in W_1.

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 refers to the number of direct customers of the AS and is easier to compute than the customer cone. Also, 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 for peering-based, customer-count, and other alternative parameters did not yield encouraging results [12]. Consequently, the customer cone serves as the topological basis for our PFS detection algorithm.

In the distributions of the customer cones in the UCLA and CAIDA datasets, only a tiny fraction of all ASes have a large customer cone [12]. Table 2 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 2(a) 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 2(a) 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 2(b) depicts the CAIDA customer-cone ranks of all ASes in W_1 during 2009. In agreement with table 1, the customer-cone results in figures 2(a) and 2(b) imply that the UCLA datasets are less noisy and thus more suitable for PFS detection than the CAIDA datasets.

6.2 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:

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

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

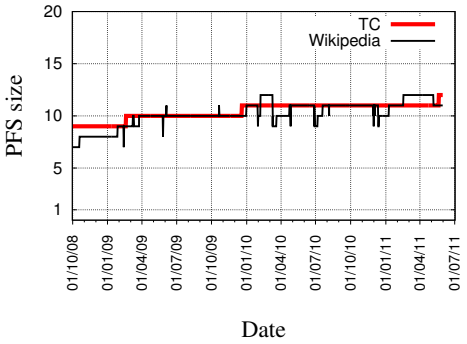


Fig. 3: PFS size according to Wikipedia and TC algorithm for the UCLA datasets.

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 3 depicts the PFS size according to Wikipedia, with short-lived spikes representing spurious revisions. For every day throughout the 32-month interval, figure 3 also plots the PFS size as per equation 1 with the value of k set to 0.00032 using UCLA data, which corresponds to 1 in about 3000 Internet ASes being provider-free.

Whereas the amount of the available data is too limited to recommend strongly the specific value of k or confidently state the proportionality of PFS to the overall population of Internet ASes, our evaluation suggests that equation 1 offers a reasonable approximation for the PFS size [12].

6.3 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.

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 7, 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) and present it in detail in figure 4.

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

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. 4: TC (Temporal Cone) algorithm and its notation.

7 Evaluation

According to sections 4 through 6, the datasets from UCLA are available for more months and less noisy than the CAIDA datasets. To evaluate the developed TC algorithm, section 7.1 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. In section 7.2, we assess the parameter sensitivity of the TC algorithm.

7.1 TC results

During its first iterative stage, the TC algorithm determines the AS customer-cone ranks and PFS sizes for all $M = 32$ months. As shown in figure 3, 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 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 3 sums up the performance of the TC algorithm. A quick comparison of these results with table 1 reveals that the TC algorithm detects PFS significantly better than the straightforward method.

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

Table 3: 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).

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

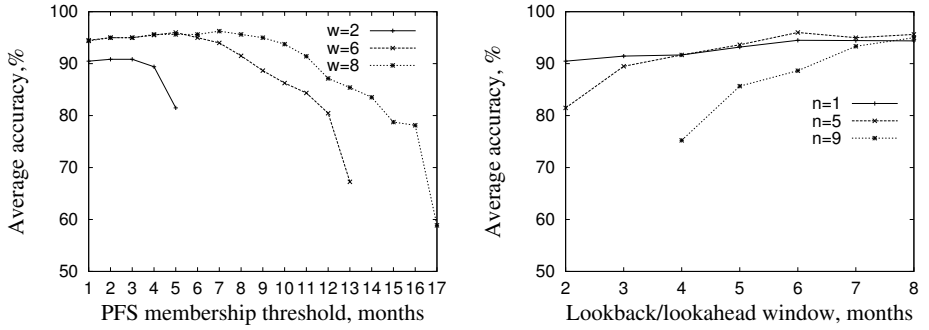
For the TC results in table 3, the accuracy of the PFS detection is 90% for 8 months and perfect 100% for the other 12 months. Thus, the corresponding average accuracy of the PFS detection is 96%.

7.2 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 for the UCLA datasets. Throughout this study, we use $k = 0.00032$ as discussed in section 6.2.

Figure 5(a) shows that 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 5(a) 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 5(b) reveals that as w grows, the average accuracy increases first but then tends to flatten out. With $n = 5$ months, the average accuracy reaches



(a) Sensitivity to n (b) Sensitivity to w

Fig. 5: Sensitivity of the TC algorithm average accuracy to PFS membership threshold n and lookback/lookahead window w for the UCLA datasets.

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 conducted a similar study [12] and observed a qualitatively similar profile for the sensitivity of the TC algorithm average accuracy to the PFS membership threshold and lookback/lookahead window. The optimal setting is 5 months for both parameters w and n and close to the settings recommended above for the UCLA data source.

8 Conclusion

PFS, or the set of provider-free ASes, is important for Internet resilience and economics. Based on the public UCLA and CAIDA datasets of inferred inter-AS economic relationships, our paper 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.

References

1. Akella, A., Maggs, B., Seshan, S., Shaikh, A., Sitaraman, R.: A Measurement-based Analysis of Multihoming. In: Proceedings of ACM SIGCOMM 2003
2. Augustin, B., Krishnamurthy, B., Willinger, W.: IXPs: Mapped? In: Proceedings of ACM SIGCOMM 2009

3. Chen, K., Choffnes, D.R., Potharaju, R., Chen, Y., Bustamante, F.E., Pei, D., Zhao, Y.: Where the Sidewalk Ends: Extending the Internet AS Graph using Traceroutes from P2P users. In: Proceedings of ACM CoNEXT 2009
4. Cooperative Association for Internet Data Analysis: AS Relationships. www.caida.org/data/active/as-relationships
5. Dhamdhere, A., Dovrolis, C.: Ten Years in the Evolution of the Internet Ecosystem. In: Proceedings of ACM IMC 2008
6. Dhamdhere, A., Dovrolis, C.: The Internet is Flat: Modeling the Transition from a Transit Hierarchy to a Peering Mesh. In: Proceedings of ACM CoNEXT 2010
7. Dimitropoulos, X., Krioukov, D., Fomenkov, M., Huffaker, B., Hyun, Y., claffy, k., Riley, G.: AS Relationships: Inference and Validation. ACM SIGCOMM CCR, 2007
8. Gao, L.: On Inferring Autonomous System Relationships in the Internet. IEEE/ACM Trans. Netw. (2001)
9. Ge, Z., Figueiredo, D., Jaiswal, S., Gao, L.: Hierarchical Structure of the Logical Internet Graph. In: Proceedings of SPIE ITCOM 2001
10. Georgia Institute of Technology, AS Topology Data: http://www.cc.gatech.edu/~amogh/topology.html
11. Gregori, E., Improta, A., Lenzini, L., Rossi, L., Sani, L.: BGP and Inter-AS Economic Relationships. In: Proceedings of IFIP Networking 2011
12. Hasan, S., Gorinsky, S.: Obscure Giants: Detecting the Provider-free ASes. http://www.networks.imdea.org/portals/8/Downloads/Publications/Technical-Reports/TR-IMDEA-Networks-2011-3.pdf, Technical report, TR-IMDEA Networks-2011-3, December 2011
13. Leskovec, J., Kleinberg, J., Faloutsos, C.: Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations. In: Proceedings of ACM SIGKDD 2005
14. Ma, R.T.B., Chiu, D.M., Lui, J.C.S., Misra, V., Rubenstein, D.: On Cooperative Settlement Between Content, Transit and Eyeball Internet Service Providers. In: Proceedings of ACM CoNEXT 2008
15. Mahajan, R., Zhang, M., Poole, L., Pai, V.: Uncovering Performance Differences among Backbone ISPs with Netdiff. In: Proceedings of NSDI 2008
16. Oliveira, R., Pei, D., Willinger, W., Zhang, B., Zhang, L.: The (in)Completeness of the Observed Internet AS-level Structure. IEEE/ACM Trans. Netw. (2010)
17. Subramanian, L., Agarwal, S., Rexford, J., Katz, R.: Characterizing the Internet Hierarchy from Multiple Vantage Points. In: Proceedings of IEEE INFOCOM 2002
18. University of California Los Angeles: Internet Topology Collection. irl.cs.ucla.edu/topology
19. Wikipedia: Tier 1 Network. http://en.wikipedia.org/wiki/Tier_1_network
20. Wikipedia: Tier 1 Network: 25/3/2009 revision, en.wikipedia.org/w/index.php?&oldid=279646779
21. Wikipedia: Tier 1 Network: 5/6/2009 revision, en.wikipedia.org/w/index.php?&oldid=294566542
22. Wikipedia: Tier 1 Network, 10/2/2011 revision: en.wikipedia.org/w/index.php?&oldid=413097463
23. Wu, J., Zhang, Y., Mao, Z.M., Shin, K.G.: Internet Routing Resilience to Failures: Analysis and Implications. In: Proceedings of ACM CoNEXT 2007
24. Xia, J., Gao, L.: On the Evaluation of AS Relationship Inferences. In: Proceedings of IEEE GLOBECOM 2004
25. Zhang, B., Liu, R., Massey, D., Zhang, L.: Collecting the Internet AS-level Topology. ACM SIGCOMM CCR, 2005