Leveraging Inter-domain Stability for BGP Dynamics Analysis

Thomas Green[1,2], Anthony Lambert[1], Cristel Pelsser[3], and Dario Rossi[2]

[1] Orange Labs
[2] Telecom ParisTech
[3] University of Strasbourg / CNRS

Abstract. In the Internet, Autonomous Systems continuously exchange routing information via the BGP protocol: the large number of networks involved and the verbosity of BGP result in a huge stream of updates. Making sense of all those messages remains a challenge today. In this paper, we leverage the notion of "primary path" (i.e., the most used inter-domain path of a BGP router toward a destination prefix for a given time period), reinterpreting updates by grouping them in terms of primary paths unavailability periods, and illustrate how BGP dynamics analysis would benefit from working with primary paths.
Our contributions are as follows. First, through measurements, we validate the existence of primary paths: by analyzing BGP updates announced at the LINX RIS route collector spanning a three months period, we show that primary paths are consistently in use during the observation period. Second, we quantify the benefits of primary paths for BGP dynamics analysis on two use cases : Internet tomography and anomaly detection. For the latter, using three months of anomalous BGP events documented by BGPmon as reference, we show that primary paths could be used for detecting such events (hijacks and outages), testifying of the increased semantic they provide.

1 Introduction

The Internet, from an inter-domain perspective, is a collection of routers scattered in about 60,000 Autonomous Systems (ASes) [3]. To ensure the full connectivity over the Internet, routers use the Border Gateway Protocol (BGP) [29] to announce reachability information concerning IP prefixes. More precisely, upon reception of routing updates from a neighbor, a BGP router first applies import policies, which might filter or modify the route. In case this information triggers some change of its routing table, the router may announce an update to its neighbors. Thus, each router announces *at most one best path* (except with BGP multi-path extension [35]) for each destination to its neighbors and sends an update whenever this best path changes. Best path selection is non-trivial due to complex and opaque BGP policies on the one hand, and to the fact that updates propagate hop-by-hop across the network on the other hand: particularly, this results in a limited visibility of the whole topology for any router, and can also lead to slow convergence because of the *path exploration* phenomenon [23].

Path exploration can happen whenever a BGP router has several neighbors announcing a path to a given prefix. Depending on the arrival order of announcements, a router

might explore transient paths before converging to its best path. Note that path exploration may cascade: a router exploring paths may trigger the exploration of paths by its neighbors. In short, while BGP routers seek best paths, opacity and verbosity of BGP along with limited visibility make it hard to analyze BGP dynamics. It is still challenging to determine the causes of BGP updates [16, 17, 14, 6] – which is crucial to detect and mitigate prefix hijacking, as well as for detecting misconfigurations and leakages, or troubleshooting network operations.

The new proposal of this paper is to systematically leverage inter-domain stability to preprocess BGP updates, with the goal of augmenting the data source (Sec. 2). More precisely, we first discuss and validate the notion of primary path, i.e., the most used inter-domain path for a router to a prefix in a given time period. Using primary paths as reference, updates can therefore be interpreted in terms of deviations from a nominal behavior, and grouped accordingly for further analysis (Sec. 3). By leveraging three months worth of BGP updates and publicly available data from a well-known alert service, we demonstrate interest of primary paths on two use cases: inter-domain tomography and anomaly detection (Sec. 4).

2 Related Work

BGP dynamics has been widely studied in the past, both for tomography and anomaly detection purposes. A thorough overview is out of the scope of this paper, but we briefly contextualize where our contributions take place.

Past works on tomography have mainly leveraged temporal and topological properties of updates to characterize BGP dynamics. Labovitz et al. [19, 10] analyzed various temporal properties of updates from inter-arrival time to convergence time. Li et al. [20] extended these works and analyzed the evolution of these properties over a decade. Elmokashfi et al. [13] studied updates churn, pointing out recurrent events on BGP dynamics. Instead, in this article we propose to leverage inter-domain stability to characterize BGP dynamics.

Past significant works on anomaly detection have been broadly reviewed in [2]. Techniques used to analyze updates include time series analysis [22, 27], statistical pattern recognition [12, 32], machine learning [33, 1], and historical data [18, 15]. Other techniques exist, such as visualization approaches [21, 25, 8, 9]. Historical data techniques consist in keeping track of all previously used paths to analyze new announcements. Instead, our proposal is to identify and only use stable paths to interpret updates.

It must also be noted that path stability in the Internet has already been pointed out. In 1996, Paxson [26] sampled routes in use between 37 hosts through periodic $traceroutes$ and showed that they were mostly stable. Moreover, by analyzing their prevalence (probability to observe a particular route) and persistence (probability for a route to be used for a long period of time), they exhibited the existence of dominant routes. Rexford et al. [30] defined events as group of updates arriving close in time and pointed out that inter-domain paths related to popular destinations were undergoing few events. Chang et al. [7] grouped updates into bursts based on temporal thresholds and showed that many path advertisements were resulting from transient path changes. Some works also leveraged the notion of path stability for specific purposes. Butler et

al. [5] showed that ASes have few distinct paths for a prefix over time and proposed to use this observation to reduce the cost of cryptographic BGP path authentication. Qiu et al. [28], assuming inter-domain stability, proposed to leverage it through machine learning to detect bogus routes. In this paper, we extend these works by showing that stability holds across the whole inter-domain and that it can be systematically leveraged for different use cases of BGP dynamics analysis.

3 Inter-domain stability

3.1 Primary paths

Our approach builds on the assumption that the BGP inter-domain structure is highly stable over relatively long periods of time [5, 30, 26]. We show that this is a reasonable assumption in Sec. 4.1. Intuitively, we expect this stability to follow from the timescale of changes among AS agreements that are negotiated few times a year. Consider indeed that BGP best path selection starts by assessing the *local preference* attribute, which encodes business agreements between ASes: it follows that every router r should have a set of preferred paths (with the same highest local preference) toward any prefix p over relatively long periods of time, and deviate from those only during relatively short transient periods (e.g., due to path exploration).

In this article, we additionally argue that, among those preferred paths, there is one dominant path that is consistently chosen as best path during an observation window W: we refer to this path as the **primary path** of r to p. In a more formal way, considering for the time being an offline case for the sake of simplicity, let us define as $T_x(r, p)$ the sum of the cumulative time during W that router r uses path x to reach prefix p. Then the primary path is selected as the one satisfying $argmax_x T_x(r, p)$.

Following from the given definition above, we compute primary paths in an offline fashion from updates collected at the LINX RIPE RIS route collector [31] on a three month time window (from January 1st to March 31st 2017). The dataset consists of $487,104,558$ IPv4 updates ($157,249,182$ IPv6 updates) and $5,482,564$ IPv4 $<router, prefix>$ pairs ($412,350$ IPv6 pairs). It includes 38 IPv4 vantage points (14 IPv6 vantage points) among which 7 announce a "full" routing table in IPv4 (10 for IPv6). To bootstrap the primary path repository, we use the last routing table dump (`bview.20161231.2359`) before the beginning of our observation window. We use BGPstream [24] to decode MRT files. Results are shown in Fig. 1. We start by confirming that in most cases primary paths dominate other paths over relatively long periods of time. Specifically, the figure shows the percentage of time that the primary path was used during the observation period $W = 3$ months for all $<router, prefix>$ pairs. Formally, denoting as before with x_1 the primary path and with $T_{x_1}(r, p)$ the sum of the cumulative time during W where router r uses path x_1 to reach prefix p, the figure shows the complementary cumulative distribution function (CCDF) of the primary path usage during the whole observation period, i.e., $T_{x_1}(r, p)/W$. The data shows that about 85% of the primary paths in IPv4 (90% for IPv6) are in use at least about half of the observation period W, and even more interesting, about 35% IPv4 (42% IPv6) primary paths are in use for over 99.9% of W.

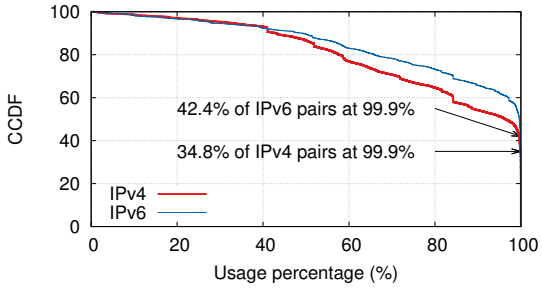


Fig. 1: Complementary CDF of the percentage of time a primary path for prefix p was used by router r over the whole observation period (January 1st to March 31st 2017).

3.2 Pseudo-events

Under the assumption of primary paths stability over long timescales, we argue that BGP dynamics can be described in terms of:

- **Transient events,** where some routers explore paths before reconverging to their primary paths (e.g., possibly due to failure, misconfiguration, attack, etc).
- **Structural events,** where some routers explore paths before switching consistently to a new primary path (e.g., as a result of routing policy or agreement changes).

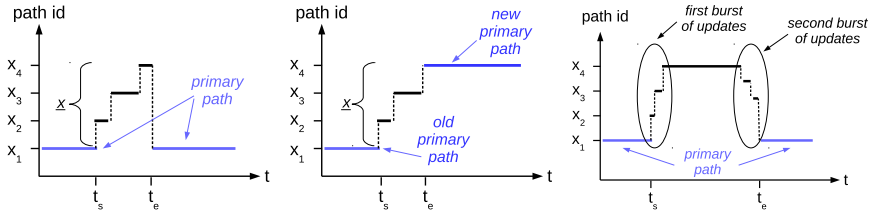
A given event can impact many primary paths from many routers to many prefixes. Therefore, to keep working at the $<router, prefix>$ pair granularity we define the notion of **pseudo-event** as the impact of an event for a given primary path x_1 used by a router r to a prefix p. Thus it is possible to further distinguish between:

- **Transient pseudo-events,** making r explore path(s) to p and reconverge to x_1.
- **Structural pseudo-events,** making r explore path(s) to p and converge to a new primary path x_1'.

Moreover, pseudo-events can be characterized by:

- **a duration**: period of time where the primary path x_1 from r to p is not used, identified by a start time t_s and an end time t_e;
- **a path exploration sequence**: sequence of $N-1$ transient paths $\underline{x} = (x_2, \ldots, x_N)$ to reach prefix p.

Fig. 2a and Fig. 2b portray the above cases. Therefore, pseudo-events enable to group updates following a primary path unavailability, instead of relying on some temporal threshold [23, 7, 30] (which result in grouping updates into bursts). An interesting follow-up characteristic from this paradigm is that pseudo-events are resilient to long-lasting events. Indeed, the longer an event lasts (a failure for instance) the more likely it



(a) Transient pseudo-event. (b) Structural pseudo-event. (c) Multiple bursts.

Fig. 2: Illustration of types of pseudo-events, and multiple bursts of updates scenario for a single event.

is to lead to several bursts of updates as illustrated in Fig. 2c. In such situation grouping updates based on primary paths unavailabilities will group all the bursts into a single pseudo-event. For these reasons, whereas BGP dynamics analysis typically work on the stream of BGP updates, our proposal is to work on the stream of pseudo-events instead.

3.3 Practical primary path computation

In this section we study the feasibility of relying solely on routing table (i.e., RIB, for Routing Information Base) dumps to compute primary paths. Recall that in Sec. 3.1 we computed the primary path repository with BGP updates in an offline fashion. However, to be of any practical interest, primary paths should be easily computable at any time in a simple, efficient and online manner. To do this, we propose to use RIBs. As an alternate form of BGP data, we consider RIBs to be easier to work with than updates because they provide snapshots of paths for all $<router, prefix>$ pairs. Besides, to bootstrap the primary path repository in Sec. 3.1 we already had to rely on a RIB dump.

First question arising is how many consecutive RIBs are required to capture stability. Indeed, there is no guarantee that a single RIB contains the primary paths. However, as primary paths are computed as the most used path over time, they should be easily identifiable from the observation of multiple consecutive dumps. The question is then *how many* consecutive RIB dumps are needed in practice for an accurate selection. To answer this question, we operate as follows. For the 10 days preceding our observation window (i.e., December 22nd to 31st 2016), we select one RIB dump per day during d consecutive days ($\forall d \in [1, 10]$). For a router r and a prefix p, we extract as its primary path the most present path in the d consecutive dumps. We then compare, for all $<router, prefix>$ pairs, what fraction of primary paths obtained with different number $\forall d \in [1, 9]$ of RIB dumps match those obtained with the d =10 full interval (used as reference). Results are shown in Table 1 for IPv4. It can be seen that computing primary path with RIB dumps has over 97% chances of success even from a single snapshot, and rapidly exceeds 99% accuracy by adding a few snapshots. Additionally, the results also suggest primary paths to be stable with high probability on at least a weekly timescale. Now that we know that a few RIB dumps give us the same primary paths than a large number of dumps, we aim to determine if RIB dumps are a suitable mean to compute primary paths. For this purpose, we compare the primary paths obtained with RIBs

Table 1: IPv4 primary path bootstrap accuracy: percentage of primary paths matching those of the 10-snapshots reference when using $d \leq 9$ snapshots.

Snapshots	1	2	3	4	5	6	7	8	9
Accuracy	97.4%	97.9%	98.5%	98.8%	99.0%	99.1%	99.2%	99.4%	99.7%

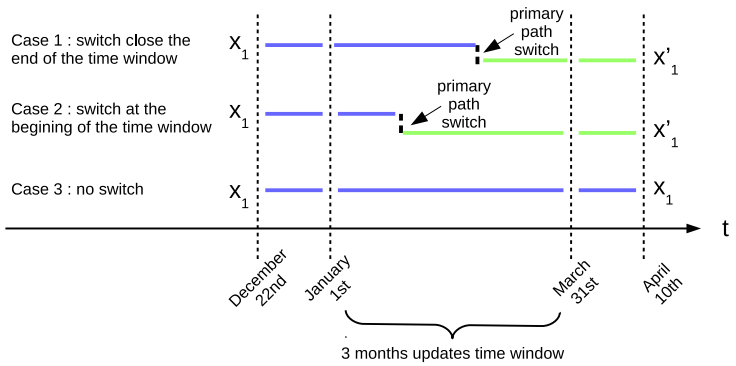


Fig. 3: Cases of primary paths switches during the observation window.

with the ones previously computed from updates in Sec. 3.1. We obtain a matching of 76.48%. The non-matching fraction could result from inefficiency of the method or from primary paths switching during the time window. In fact, if such switches occur at the beginning of the time window, then for a given $<router, prefix>$ pair the primary path x_1 computed with RIBs will differ from the one (x'_1) computed with updates, as illustrated by case 2 in Fig. 3. To investigate whether the non-matching fraction is due to primary path switches, we compute primary paths using RIBs from April 1st to April 10th (i.e., the 10 days following the end of the observation window). This time, we obtain a matching of 85.95% when comparing primary paths from the RIBs in April with those computed with updates. Once again the non-matching fraction could result from method inefficiency or from switches (this time, at the end of the time window, as illustrated by case 1 in Fig. 3). Finally, it appears that 95.5% of primary paths computed on updates are either matching those computed from the RIBs of December or those of April. This confirms that most of the non-matching fraction is due to primary paths switches (i.e., structural pseudo-events) during the time window. It also highlights the need to periodically update the primary paths repository. We leave the study of the characterization of primary path turnover as part of our future work. We will show in the next section that even without updating the primary path repository periodically during the time window, we still get valuable results. In this section we have shown that the paths most present in a few RIB dumps are a good indicator of the primary paths used for the next days. In fact, these paths highly overlap with primary paths obtained from the stream of updates, meaning that RIBs are thus a viable approach to compute primary paths in practice.

Table 2: Volume gain in the pseudo-events domain

	IPv4	IPv6
Updates	487,104,558	157,249,182
Pseudo-events	57,066,053	17,687,525
Structural pseudo-events	1,406,392	78,995
Transient pseudo-events	55,659,661	17,608,530
Reduction factor	8.5	8.9

4 BGP dynamics

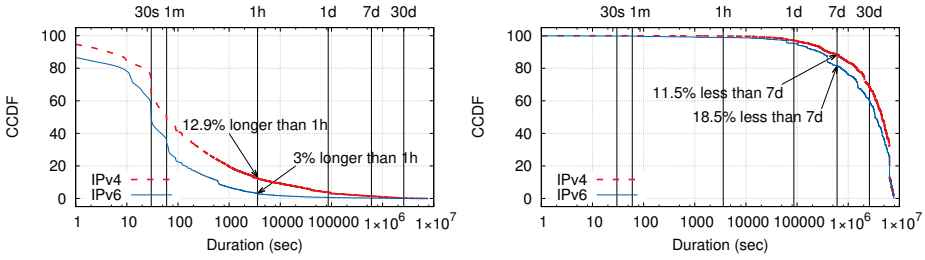
In this section, we apply our methodology on two classic use cases of BGP dynamics analysis: inter-domain tomography and anomaly detection.

4.1 First use case: Inter-domain tomography

The Internet as a set of interacting ASes is a complex environment. Tomography intends to infer the internal characteristics of a system from external observations. Pseudo-events are groups of updates based on primary paths unavailability periods which exhibit two interesting properties: a duration and the sequence of paths explored.

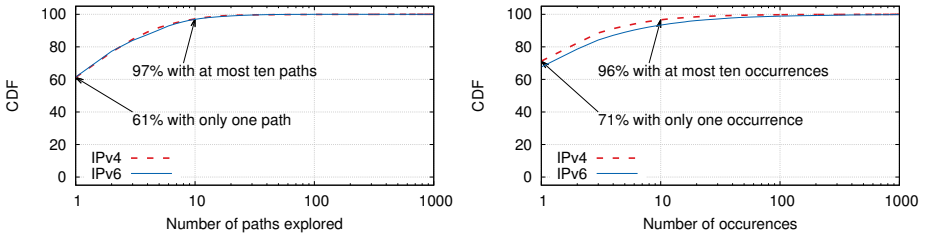
We analyze updates from January 1st 2017 to March 31st 2017 (W = 3 months) collected at the RIPE RIS LINX, in the order of arrival. Each update is processed against the primary path repository built upon the last 10 days of December's RIBs. Upon detection of an update for a $<router, prefix>$ pair announcing the start at t_s of a primary path x_1 unavailability (i.e., the path announced is not the primary path), a pseudo-event object is created. The subsequent updates observed for the same $<router, prefix>$ pair, which relate to the ordered set $\underline{x} = (x_2, \ldots, x_N)$ of paths explored, are indexed into this object and can be processed further, for example to characterize anomalies (e.g., outages, hijack, etc. as explored in Sec. 4.2). If an update announcing x_1 is observed at time $t_e > t_s$, then this pseudo-event is classified as transient, and its duration is t_e - t_s. If at the end of the time window W the pseudo-event has not reconverged to x_1 then this pseudo-event is classified as structural, and its duration is set to $W - t_s$. Results are presented in Table 2. First of all, it can be noticed that the systematic indexing of updates into pseudo-events result in a reduction of the number of objects that will have to be further processed for analysis: i.e., rather than analyzing a stream of updates we analyze a stream of pseudo-events. The reduction factor is *almost one order of magnitude* when transforming the stream of updates into a stream of pseudo-events: more precisely, it is a sizable factor of 8.5 (8.9) volume reduction for IPv4 (IPv6). There is therefore a practical volume gain when working with pseudo-events. Moreover, as it could be expected, it appears that transient pseudo-events largely dominate structural ones: less than 2.5% of pseudo-events are structural. We now further investigate pseudo-event properties and the light they shed on BGP dynamics.

Pseudo-events duration. We first turn our attention to temporal properties of pseudo-events. Comparison of Fig. 4a and Fig. 4b confirms our expectations: transient pseudo-



(a) Transient pseudo-events. (b) Structural pseudo-events.

Fig. 4: Complementary CDF of pseudo-events duration $t_e - t_s$, in semi-log-x scale (bottom x-axis reports duration in seconds, top x-axis uses more human-friendly units).



(a) Path exploration length. (b) Occurrences.

Fig. 5: CDF of transient pseudo-events path exploration length and occurrences.

events (i.e., those which reconverge to the primary path after a path exploration) have indeed small duration (Fig. 4a), while structural pseudo-events (i.e., those which did not reconverge to the primary path) have long durations (Fig. 4b). Particularly, about 50% of IPv4 (60% IPv6) transient pseudo-events last *less than a minute*, whereas only about 11% of IPv4 (18% IPv6) structural pseudo-events last *less than a week.*

Pseudo-events path exploration. The distribution of the path exploration length card($\underline{x}$) is reported in Fig. 5a. It clearly appears that transient pseudo-events explore relatively few paths, just 1 in 60% of cases and rarely more than 10 (3% of cases) before reconverging on the primary paths. In other words, transient pseudo-events index 2 updates in 60% of cases and rarely more than 11 (3% of cases). More interestingly, if we characterize a pseudo-event by its sequence of paths explored then we can detect when a pseudo-event occurs multiple time, as illustrated on Fig. 5b. If most transient pseudo-events occurred only once during the time window (about 70%), it also appears that a few occurred a lot (sometimes hundreds of times).

Global picture. Analysis of pseudo-events properties provide us with the following global picture. BGP dynamics are mostly made of short termed instabilities producing limited path exploration. This would suggest that classic solutions for BGP dynamics regulation such as MRAI timers [29] or route flap damping [34] are efficient (see gaps on Fig. 4a at 30 seconds and 60 seconds, most likely due to MRAI effects, pointing out its ability to limit path exploration). However, results also point out that instabilities are recurrent. In such situations, classical mechanisms are ineffective by design. To be tackled, recurrent instabilities would require advanced contention mechanisms, able to learn and recognize them, which we intend to investigate as part of our future work.

4.2 Second use case: Anomaly detection

We now show how information provided by primary paths can be leveraged for anomaly detection. Specifically, we use as reference the list of noteworthy BGP events published by a well-known alert service, BGPmon [4], which classifies events as either (i) AS or country-level outages, (ii) hijacks and (iii) leaks, and for each event reports the inferred starting time. We argue that the usefulness of pseudo-events is better assessed by focusing on AS-level *outages* (i.e., an outage on AS x is an event impacting prefixes originated by AS x) and *hijacks* (a hijack on AS x by AS y is identified whenever AS y has originated some route for a prefix p, such that p is a prefix, or a more specific prefix, legitimately originated by AS x). Notice indeed that country-level outages and leaks would require to use IP geolocation databases or AS relationships databases respectively: clearly, the use of different databases than BGPmon would be a further source of uncertainty, which would unnecessarily fuzz the comparison.

Considering the same period of our dataset (January 1st to March 31st 2017), BGPmon lists 2369 events (1716 outages and 653 hijacks). Since we are using different (and fewer) vantage points than BGPmon, we need to remove non observable events. We perform this sanitization using BGP updates. For each event starting at time t_{BGPmon} we define a time window as $|t - t_{BGPmon}| < 120$ seconds. The duration of this time window is chosen wide enough to take into account updates propagation time (especially due to MRAI effects) among different vantage points. For an outage on AS x, if no update was seen during the time window for any prefix originated by AS x then we say this event was not observable. For a hijack on AS x by AS y, if no update was seen during the time window for prefix p originated by AS y, then we say this event was not observable. We gather that 441 (94 outages and 347 hijacks) events were not observable at our collector, leaving us with a total of 1928 events (1622 outages and 306 hijacks). We next investigate if we find pseudo-events related to the events inferred by BGPmon.

Outages. For an outage on AS x, we look for pseudo-events for prefixes originated by AS x. As reported in Table 3a, for 1355 (83.5%) outages, we detect such pseudo-events starting during the same time window ($|t - t_{BGPmon}| < 120$): we say that we detect them *on-time*. More interestingly, for 236 (14.6%) outages, the related primary paths unavailability periods already started well before the window: in 229 out of 236 of cases, the time difference with respect to BGPmon is larger than one hour. In this case, it is rather clear that the observed starting time difference cannot be just explained

Table 3: Relevance of primary paths for anomaly detection

(a) Outages

Reported BGPmon events	1716
Observable BGPmon events	1622
Detected pseudo-events	1591 (98.1%)
−*On-time*	1355
−*Early*	236
Undetected pseudo-events	31 (1.9%)

(b) Hijacks

Reported BGPmon events	653
Observable BGPmon events	306
Agreements	173 (56.5%)
Disagreements	133 (43.5%)
−*Explicit*	37
−*Implicit*	96

by propagation delays among multiple collectors. Under our formalism, the reported starting time could correspond to some bursts of updates (recall Fig. 2c) instead of the beginning of primary paths unavailabilities. We argue in this case that we *early* detect pseudo-events related to these outages with respect to BGPmon. Finally, only 31 (1.9%) outages were not detected with pseudo-events (which requires further investigation).

Hijacks. For a hijack on AS x by AS y, we record every update originated by AS y, announced within $|t - t_{BGPmon}| < 120$ by any router r for any prefix p and analyze it against our primary paths repository. More precisely, if there exists a primary path for $<r, p>$, then we compare it to the path in the update and assess whether we agree or explicitly disagree (according to our repository, AS y is legitimate to originate p) with BGPmon. If no primary path for $<r, p>$ exists, then we search for a primary path $<r, p'>$, such that p' is less specific than p (first level less specific) and compare paths. Finally, if no such primary path exists, we implicitly disagree: according to our repository it is a harmless update for a newly originated prefix.

As reported in Table 3b, for 173 (56.5%) hijacks we agree with BGPmon. For 133 (43.5%) hijacks we disagree with BGPmon, either explicitly (37 hijacks) or implicitly (96 hijacks). Investigating the reasons for this important number of disagreements, we discover that 103 of them have occurred in March and impacted the same origin AS (AS 13489) and prefix (2800::/12)[4]. In other words, during March this very prefix and origin AS was hijacked 103 times, moreover by tens of different ASes originating prefixes all more specific than 2800::/12. Analyzing Regional Internet Registries (RIR) statistics files which summarize the current state of Internet number resource allocations and assignments, and executing $whois$ requests on RIR's databases, it appears that 2800::/12 is not allocated nor assigned (at the time of writing). This prefix, which started being originated by AS13489 on March 3rd (according to our dataset) should not have therefore been routed (it was no longer routed at the time of writing). On the contrary, the RIR's databases also indicates that 11622 prefixes more specific than 2800::/12 have legitimately been allocated or assigned. This more likely illegitimate origination of 2800::/12 by AS13489 would therefore have triggered hijacks detection by BGPmon for any legitimate update related to any more specific prefix than 2800::/12. As a con-

[4] We are aware that this prefix was used in Czyz et al. [11]. We believe that the events are unrelated because they do not match either the involved parties, the time window, or the methodology described.

clusion, we are reassured in the relevance of primary paths for hijack detection. The remaining 30 hijacks are marked as disagreement, though reasons of disagreement are still uncertain and require further investigation.

5 Conclusions and future work

This paper discusses the concepts of *primary paths* (most used inter-domain paths in a time period) and *pseudo-events* (primary path unavailability periods). Using three months of BGP updates at a collector, we verify our assumption to hold, and show how to take advantage of the inter-domain stability by augmenting the stream of BGP updates with primary paths, thus creating a new stream of pseudo-events. This new stream exhibits interesting characteristics for BGP dynamics analysis, as shown on two use cases. First, it helps us in building tomographic views of the inter-domain structure, uncovering or confirming many temporal and topological characteristics. Second, our comparison with the BGPmon alert service indicates that the knowledge of the primary path can be used for anomaly detection. It enables to promptly detect any deviation from this nominal behavior, and is also helpful in characterizing the type of deviation.

Therefore, primary paths provide a powerful repository to interpret BGP updates, and this paper just scratches the surface of their usage. As part of our ongoing work, we are investigating their topological properties (to correlate pseudo-events), analyzing temporal properties of structural pseudo-events (to characterize primary paths turnover), with the purpose of proposing an online framework to detect and mitigate BGP events.

Acknowledgments. We thank the anonymous reviewers whose valuable comments helped us improving the quality of this paper.

References

1. Al-Rousan, N.M., Trajković, L.: Machine learning models for classification of BGP anomalies. In: Proc of. IEEE HPSR (2012)
2. Bahaa, A.M., Philip, B., Grenville, A.: BGP Anomaly Detection Techniques: A Survey. In: IEEE Communications Surveys & Tutorials (2016)
3. Bates, T., Smith, P., Huston, G.: CIDR Report (Accessed in 2018), `http://www.cidr-report.org/as2.0/`
4. BGPmon.net: Public event reporting (Accessed in 2018), `https://bgpstream.com`
5. Butler, K., McDaniel, P., Aiello, W.: Optimizing BGP Security by Exploiting Path Stability. In: Proc of. ACM CCS (2006)
6. Caesar, M., Subramanian, L., Katz, R.H.: Root Cause Analysis of BGP Dynamics. In: Proc. of ACM IMC (2003)
7. Chang, D.F., Govindan, R., Heidemann, J.: The temporal and topological characteristics of BGP path changes. In: Proc. of IEEE ICNP (2003)
8. Chen, M., Xu, M., Li, Q., Song, X., Yang, Y.: Detect and analyze large-scale BGP events by bi-clustering Update Visibility Matrix. In: Proc. of IEEE IPCCC (2015)
9. Comarela, G., Crovella, M.: Identifying and analyzing high impact routing events with PathMiner. In: Proc. of ACM IMC (2014)
10. Craig, L., Robert, M.G., Jahanian, Farnam: Origins of Internet routing instability. In: Proc of. INFOCOMM (1999)

11. Czyz, J., Lady, K., Miller, S.G., Bailey, M., Kallitsis, M., Karir, M.: Understanding IPv6 Internet Background Radiation. In: Proc. of ACM IMC (2013)
12. Deshpande, S., Thottan, M., Ho, T.K., Sikda, B.: An online mechanism for BGP instability detection and analysis. In: IEEE Transactions on Computers, vol. 58, pp. 1470–1484 (2009)
13. Elmokashfi, A., Kvalbein, A., Dovrolis, C.: BGP Churn Evolution: A Perspective From the Core. In: Proc. of IEEE Transactions on Networking (2011)
14. Feldmann, A., Maennel, O., Mao, Z.M., Berger, A., Maggs, B.: Locating Internet routing instabilities. ACM SIGCOMM Computer Communication Review (CCR) 34 (2004)
15. Haeberlen, A., Avramopoulos, I., Rexford, J., Druschel, P.: NetReview: Detecting When Interdomain Routing Goes Wrong. In: Proc. of NSDI (2009)
16. Holterbach, T., Vissicchio, S., Dainotti, A., Vanbever, L.: SWIFT: Predictive Fast Reroute. In: ACM SIGCOMM (2017)
17. Javed, U., Cunha, I., Choffnes, D., Katz-Bassett, E., Anderson, T., Krishnamurthy, A.: PoiRoot: Investigating the Root Cause of Interdomain Path Changes. In: ACM SIGCOMM (2013)
18. Karlin, J., Forrest, S., Rexford, J.: Pretty good BGP: Improving BGP by cautiously adopting routes. In: Proc. of IEEE ICNP (2006)
19. Labovitz, C., Malan, G.R., Jahanian, F.: Internet Routing Instability. In: Proc. of ACM SIGCOMM (1997)
20. Li, J., Guidero, M., Wu, Z., Purpus, E., Ehrenkranz, T.: BGP Routing Dynamics Revisited. In: Proc. of ACM SIGCOMM Computer Communication Review (2007)
21. Lutu, A., Bagnulo, M., Pelsser, C., Maennel, O., Cid-Sueiro, J.: The BGP visibility toolkit: Detecting anomalous Internet routing behavior. In: Proc. of IEEE/ACM Transactions on Networking (TON), vol. 24, pp. 1237–1250 (2016)
22. Mai, J., Yuan, L., Chuah, C.N.: Detecting BGP anomalies with wavelet. In: Proc. of IEEE NOM (2008)
23. Oliveira, R., Zhang, B., Pei, D., Izhak-Ratzin, R., Zhang, L.: Quantifying path exploration in the Internet. In: Proc. of ACM IMC (2006)
24. Orsini, C., King, A., Giordano, D., Giotsas, V., Dainotti, A.: BGPStream: A Software Framework for Live and Historical BGP Data Analysis. In: Proc. of ACM IMC (2016)
25. Papadopoulos, S., Moustakas, K., Drosou, A., Tzovaras, D.: Border Gateway Protocol graph: detecting and visualising Internet routing anomalies. In: IET Information Security, vol. 10, pp. 125–133 (2016)
26. Paxson, V.: End-to-end routing behavior in the Internet. In: Proc of. ACM SIGCOMM (1996)
27. Prakash, B.A., Valler, N., Andersen, D., Faloutsos, M., Faloutsos, C.: BGP-lens: Patterns and anomalies in Internet routing updates. In: Proc. of ACM SIGKDD (2009)
28. Qiu, J., Gao, L., Ranjan, S., Nucci, A.: Detecting bogus BGP route information: Going beyond prefix hijacking. In: Proc. of EAI SecureComm (2007)
29. Rekhter, Y., Li, T.: A Border Gateway Protocol 4 (BGP-4). RFC4271 (2006)
30. Rexford, J., Wang, J., Xiao, Z., Zhang, Y.: BGP routing stability of popular destinations. In: Proc. of ACM SIGCOMM Workshop on Internet measurement (2002)
31. RIPE-NCC: Routing Information Service (Accessed in 2018), `https://www.ripe.net/ris`
32. Theodoridis, G., Tsigkas, O., Tzovaras, D.: A novel unsupervised method for securing BGP against routing hijacks. In: Computer and Information Sciences III, pp. 21–29 (2013)
33. de Urbina Cazenave, I.O., Köşlük, E., Ganiz, M.C.: An anomaly detection framework for BGP. In: Proc. of INISTA (2011)
34. Villamizar, C., Chandra, R., Govindan, R.: BGP Route Flap Damping. RFC2439 (1998)
35. Walton, D., Retana, A., Chen, E., Scudder, J.: Advertisment of Multiple Paths in BGP. RFC 7911 (2016)