Noisy Neighbours: Keep the Neighbourhood Quiet

Ebrima Jaw[†], Thomas Krenc[‡§], Moritz Müller[†*], kc claffy[§], Lambert Nieuwenhuis[†], Cristian Hesselman[†*]
[†]University of Twente, Enschede, The Netherlands
[§]CAIDA, UC San Diego La Jolla, CA, USA
[*]SIDN Labs, Arnhem, The Netherlands
[‡]IIJ Research Laboratory, Japan

***Abstract*—The Border Gateway Protocol (BGP) is a crucial inter-domain routing protocol that uses update messages to enable Autonomous Systems (ASes) to share network reachability information. Typically, ASes should only trigger update messages to reflect configuration changes and link failures for optimal path selection. However, we have identified recurring patterns of high-frequency repeated updates without any topological changes, which consume unnecessary resources of the route collectors for archiving and storage, and complicate downstream analysis. Although the phenomenon of noisy BGP peers and prefixes is known, current work has not quantified its scope and characteristics. This study fills this gap and analyzes over 80 billion update messages from multiple RouteViews collectors spanning several years. We identify and characterize high-frequency repeated updates driven by a small fraction of sessions and prefixes. For instance, fewer than 2% of the prefixes accounted for over 90% of update messages in some BGP update traces.**

***Index Terms*—BGP updates, route collectors, collector peers, and MRT update files**

I. Introduction

The Border Gateway Protocol (BGP) is the de facto standard inter-domain routing protocol, enabling Autonomous Systems (ASes) to exchange reachability information with their neighbors and propagate route updates across the Internet. A subset of these ASes (peers) share their routing information with collector projects such as RIPE RIS [1] and RouteViews [2], which is archived in the Multi-Threaded Routing Toolkit (MRT) format [3]. These archives have become indispensable resources: network operators use them to monitor and troubleshoot connectivity, while researchers rely on them to study Internet topology, routing dynamics, and security [4]–[7].

Over the years, the volume of BGP data collected at route collectors has steadily grown, driven by the expanding number of ASes, their increasing interconnections, and the rising number of peers contributing their views of the Internet [8].

However, with the growing complexity of the Internet and variations in BGP implementations, operators of RIPE RIS and RouteViews have observed an increasing trend of incidents in which a small set of their peers generates an extraordinary volume of updates [9]. These events inflate MRT files and strain the synchronization of new data into archives, raising concerns about the scalability of both storage and downstream processing [10]. Consistent with operator reports, we observed a single peer contributing up to 1.6 billion updates in one day.

Identifying the root causes of this pathological behavior is non-trivial. Prior studies highlight multiple contributing factors, such as link failures that trigger repeated next-hop changes in combination with transitive BGP communities [4], as well as persistent route flapping [11], where the same route is repeatedly withdrawn and reannounced. More recently, the BGP vortex phenomenon [12] has been described, where certain routing states cause updates to circulate indefinitely between networks. Other sources speculate about malformed updates [13] and buggy router implementations [14].

In our analysis, we confirm the presence of long sequences of redundant updates that repeatedly advertise identical prefixes, AS paths, and BGP communities, consistent with previous findings [15]. We collectively refer to these redundant updates as *noise*, which clutters measurements, complicates analysis, and distorts the underlying picture of internet routing.

Yet, how exactly this *noise* manifests in MRT data and how it can be safely distinguished from genuine routes remains largely unclear. Our analysis of unfiltered route update archives addresses this gap, revealing the make-up of MRT files and providing insights that inform route collector design while supporting both operators and researchers. In this work, we leverage 13 years of RouteViews MRT data to systematically quantify the prevalence and persistence of routing noise, and complement this with two months of recent data to characterize the detailed patterns of repeated updates and assess their implications. Our main contributions are:

- Our study quantifies the scale of noise and characterizes the types of updates among the top contributing RouteViews collector peers. We show that noise is both prevalent and erratic, and is driven by a few peers.
- We show that only 0.44% of sessions generated over 50% of the 83.17 billion updates in two months, mostly redundant updates from a few prefixes and AS paths.
- We challenge the notion of attributing noise to an AS or a prefix and show that an AS can exhibit different behavior for its originated prefixes.
- Based on our findings, we explore the idea of a pruning method to reduce the size of bloated MRT files due to redundant updates.

Our work highlights that coarse filtering by AS or prefix can lead to misattribution and distort topology inference. Instead, our insights enable researchers and operators to avoid processing potentially redundant updates when appropriate. Our findings provide the first step toward a clearer understanding of noise in MRT files and lay the groundwork for future efforts to identify its root causes and explore collaborative solutions.

978-3-903176-75-1 ©2025 IFIP

II. Background and Problem Scope

A. BGP and route collectors

Networks establish BGP sessions with RIPE RIS and RouteViews collectors to share their view of the Internet. The collectors receive a sequence of update messages containing information such as a prefix and its length (192.168.0.0/16), in the form of *announcements* or *withdrawals*, reflecting changes in reachability or policy [16]. BGP announcements can also include optional transitive attributes, such as BGP communities, which are tagging attributes that can be propagated or filtered at ingress or egress. Informational communities, such as geo-communities, tag announcements with ingress information and are often used for Traffic Engineering (TE). Action communities like the Remotely Triggered Black Hole (RTBH) typically signal a service provider to blackhole all traffic for a specific destination under a DDoS attack [4], [16].

B. Challenges of BGP data collection

Fig. 1 (top) shows the continuous growth of RouteViews' collected BGP data since 2012, reaching 1 billion daily updates in 2025. Route Collectors often receive redundant information because many of their peers share the same or similar views of the Internet, and due to other issues [17].

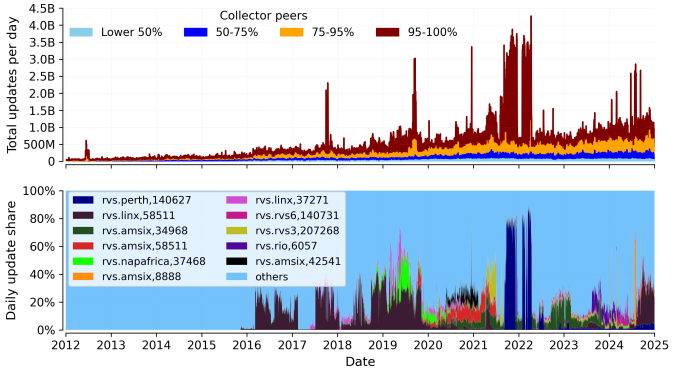


Fig. 1: Daily update share of RouteViews's peers (top) and top contributors over time (bottom). The top 5% of peers contributed a daily average of 306M updates, which is over half of all updates, while the lower 50% contributed just 5.3%.

Prior work has explored peer-selection strategies to identify route collector peers with less redundant views of the Internet [5]. In contrast, our work investigates pathological noise in the collected MRT archives of RouteViews, offering a detailed view of its characteristics and persistence. Our findings complement existing research [5], [6] by giving researchers a clearer picture of this noise and providing RouteViews insights into the factors driving archive growth.

C. Formal Definition of noise

This study considers updates for a given prefix to be noisy and redundant if repeated tens to hundreds of times per second with the same attributes or rapid oscillations of AS-path or community values in successive updates. Such updates inflate MRT files while adding little new routing information. Although this noise might be relevant to protocol behavior analysts or security researchers, it will not contribute any new information for researchers studying Internet topology.

Noise is defined as follows: Let $U = \{u_1, u_2, \ldots, u_n\}$ denote a sequence of update messages observed for prefix p within a given BGP session during a time interval T of 1 minute and each update $u_i \in U_p$ arrives at time t_i, with $t_1 < t_2 < \cdots < t_n$. Let $Atr_p(t_i)$ denote the set of BGP path attributes associated with p for each update $u_i \in U_p$ arriving at time t_i. We classify an update u_i as *noise* if $Atr_p(t_i) = Atr_p(t_{i-1})$ because there are no topological changes for p in the subsequent update messages. We also consider u_i as noise if $Atr_p(t_i) \neq Atr_p(t_{i-1})$ but exhibited rapid oscillation of AS paths or community values, possibly due to path flapping [11].

We observe this noise in various forms, such as fully redundant updates, which are prefix announcements within the same session with no attribute changes (e.g., BGP communities). Such updates could be due to a broken BGP implementation, RIB transfer, or misconfigurations [17], [18]. Additionally, it can manifest as rapid oscillation of AS paths, often associated with route flaps between a few distinct paths [11]. In summary, although both fully redundant and path flapping may manifest differently, they all introduce pathological redundancy into MRT archives. Also, since route flapping manifests as high-frequency, repeated updates that are mostly redundant, we interchangeably use the terms *noise*, *redundancy*, and *high-frequency repeated updates* in the rest of the paper.

III. Datasets

a) Longitudinal BGP dataset: Firstly, we extracted the daily update share per collector peer over 13 years (2012-2025) to quantify the data growth and assess the prevalence of high-frequency repeated updates across collector peers. Additionally, we used two months (December 2021 and 2024) of BGP data from RouteViews to investigate high-frequency repeated updates. Hence, we computed the number of unique AS paths and BGP communities for both IPv4 and IPv6 prefixes. In addition to quantifying the shared AS paths and communities among the two protocol versions, we also computed the total occurrences of AS paths and communities across all MRT files to determine the magnitude of repetition among the announcements. Table I presents details of our datasets.

IV. Methodology

We perform the following analysis to identify and characterize high-frequency repeated updates across collectors, peers, sessions, prefixes, and AS paths. In **Step** ❶, we analyze the prevalence of high-frequency updates to determine whether these update bursts are specific to certain collectors or peers. In **Step** ❷ we trace the origin of these bursts to a small fraction of sessions, prefixes, and AS paths. In **Step** ❸, we apply statistical dispersion metrics to measure the variability of prefix announcement patterns within origin ASes. Then, we conduct an inter-collector comparison analysis to demonstrate

TABLE I: Statistics of the BGP update datasets

Details of the two months BGP MRT update files				
	(2021-12)		(2024-12)	
	IPv4	IPv6	IPv4	IPv6
MRT files	104.15K (86.84K)		98.26K (93.47K)	
Sessions	505	401	682	592
Peer ASNs	305	230	321	278
Announcements	38.67B	11.16B	11.79B	18.52B
Withdrawals	631.39M	546.52M	1.08B	772.50M
Unique prefixes	1.21M	307.23K	1.22M	272.87K
Unique AS paths	90.09M	15.62M	87.47M	25.49M
Unique comm.	4.66M	717.29K	5.53M	1.30M
Details of 13 years BGP metadata				
Start Date: 2012-01-01 End Date: 2025-02-28				
	Collectors	Peers	Prefix Upds	
	36	649	2.64T	

that noise can be highly localized. In **Step** ❹, we use the above insights to propose a repetition volume metric that automatically flags bloated files resulting from high-frequency repeated updates. Finally, in **Step** ❺, we use a prototype sliding window pruning method to illustrate the storage gains achievable by minimizing these repeated updates.

A. Characterizing update contribution across collectors

We aggregate the daily update counts per collector from 2012 to 2025 to quantify the distribution of updates among collectors. We group these daily totals into six-month intervals and sum the total updates per collector within each interval, smoothing short-term fluctuations and highlighting persistent long-term trends. We rank collectors by their average update volume, computed over the six-month intervals. Finally, we extract the top 11 collectors and aggregate the rest. Section V-A presents the findings of this methodology.

B. Analysis of collector peer update contributions

Next, we construct a daily time series of the update counts of the collector peer combination to assess whether high-frequency repeated updates among the collectors are concentrated within a small fraction of peers. We then rank all collector–peer combinations by their update counts and assign them to one of four quantile bins based on these ranks: 95–100%, 75–95%, 50–75%, and the lower 50%. We compute the cumulative update share for each collector peer using the earlier-constructed daily time series to determine the top 11 contributors across the 13 years and aggregate the remaining contributors as "others". Section V-B highlights our findings.

C. Assessing the update distributions and variability

We compute the per-minute update share of each session to construct a time series of update contributions and then calculate the mean percentage contribution of each session across the month. We then compute the standard deviation to quantify the variation in the minute-level update share of each session over time. We obtain the coefficient of variation (CV [19]) by dividing the standard deviation of a session's per-bin shares by its mean share, which normalizes the measure of volatility to show how stable each session's behavior is relative to its average activity.

We also apply the Gini coefficient to measure the inequality of update contributions across sessions, ranging from 0 to 1 [7]. A Gini coefficient of 0 would mean that all sessions contribute equally, while a value near 1 would mean that a small fraction of sessions contribute to all the updates. We compute the mean update share of each session at both per-minute and monthly aggregation levels. We sort the resulting values in ascending order to analyze session-level dominance patterns across both the short and long term. We use the same method to measure the heavily uneven distribution of updates across a few prefixes and AS paths. The findings of these approaches are presented in Section V-C

D. Profiling highly announced prefixes across collectors

We count the number of unique prefix-AS-path tuples (e.g. `192.0.2.0/24`, `AS65536-AS65543-AS65551`) for each collector across December 2021 to determine whether frequently announced prefixes are observed across all collectors. We use the Perth collector as a reference point for high-frequency updates since it stood out during our analysis. We compute the 99th percentile of the distribution of total update counts across all prefixes to determine the top 1% of prefixes (841) as a candidate set of *frequently announced prefixes*. We use these candidate sets and quantify their global visibility across collectors. We used the top 10 collectors as our benchmark collectors for comparison because each of them observes over 99.76% of the 841 prefixes. Then, we aggregate the remaining 21 collectors as "rest", except for rvs6 (IPv6-only collector), which exhibits distinct behavior (Figure 6). On average, the aggregated collectors (rest) collectively observe 61.4% of the candidate set prefixes, with four of them, such as *rvs.rio* and *rvs.saopaulo*, observing ≈99% of these prefixes during December 2021. Finally, we tracked our candidate set of prefixes across all collectors to determine their total update counts, and Section V-E presents our findings.

E. Quantifying redundant update repetition in MRT files

Although redundancy may occur in both large and small MRT files, consistently detecting and quantifying this noise across different file sizes is a nontrivial task. Nonetheless, larger MRT files tend to be dominated by excessive and unnecessary repetitions of prefix announcements, which inflict disproportionately higher storage and processing overhead. We introduce the *Repetition Volume Score (RVS)* metric, which we use to flag and quantify excessive update repetition in MRT files, while scaling its impact according to the file size.

$$\mathcal{RVS} = \left(\frac{U - N}{U}\right) \cdot \log_{10}(U)$$

Where U is the total update count of an MRT file, and N is the number of unique prefixes. The *Repetition ratio* ($\frac{U-N}{U}$)

quantifies the fraction of updates that repeat an already observed prefix with little or no new information (Section II-C). The $\log_{10}(U)$ ensures that our metric prioritizes larger MRT files with high redundancy, while compressing their scale such that they do not completely overshadow smaller MRT files. Also, our metric flags smaller MRT files with higher repetition, but assigns them a proportionally lower score because their operational impact is comparatively small. The repetition ratio captures redundancy even when U is slightly larger than N. We denote $\alpha = U/N$ as the average number of updates per prefix, and $r = 1 - 1/\alpha$ as the repetition ratio. For example, if U = 120K and N = 100K ($\alpha = 1.2$), then $r = 0.167$ and RVS ≈ 0.85. In contrast, if U = 2M and N = 1M ($\alpha = 2$), then $r = 0.5$ and RVS ≈ 3.15, showing that both files have noise, but the larger file receives more weight. We extend the RVS metric to calculate the average repetition per prefix and the proportion of updates from prefixes. We defined bins based on unique prefix counts in a given MRT file (Figure 7) to analyze the repetition behavior across different MRT sizes with a comparable number of unique prefixes. Finally, we use 166,672 (≈93.1%) of the total MRT files for the two months, and exclude 12,307 (≈6.9%) MRT files that have fewer than 100 updates and ten unique prefixes, as they have a minimal impact on noise (Section V-F).

F. Pruning redundant announcements

We use a lightweight, one-second sliding-window pruning method with a 0.5-second stepping interval. The goal is to reduce the size of bloated MRT files caused by high-frequency repeated updates within a short time interval. Our method takes MRT files as input and extracts all BGP sessions, and groups announcements by a unique combination of prefix, AS path, and BGP community. Then, it retains the first occurrence of each unique observed route within the current sliding window. This method enables us to assess the potential storage benefits for both RouteViews and the researchers.

Contrary to the approaches by Alfroy et al. [5], [6], which optimize peer selection and sampling to reduce redundancy during data collection, our work focuses on untangling the already collected MRT archives to understand and characterize the unfiltered noise. Our approach eliminates high-frequency repeated updates, reduces the bloated MRT file size caused by noise, and facilitates archival scalability and usability.

V. Results

We present the details and results derived from the methods described in Section IV

A. Characterizing update contribution across collectors

Figure 2 shows the heavily skewed distribution of updates among collectors over time. In our analysis of 36 collectors, the top five collectors (Linx 15.2% and Perth 15.2%, rvs.amsix 9.6%, rvs3 8.18%, and rvs4 6.6%) together accounted for over 54% of the total updates.

In contrast, over a third of the collectors were each responsible for fewer than 1% of updates, with some contributing

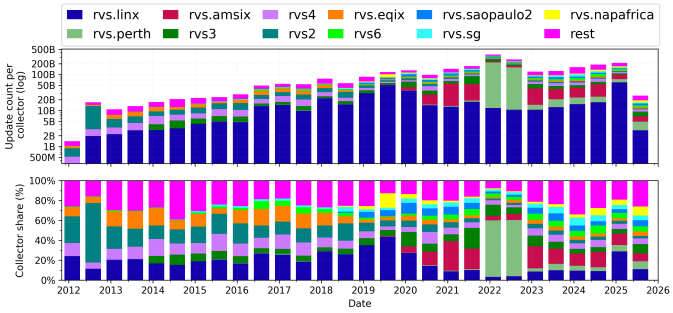


Fig. 2: While over twenty collectors each contributed fewer than 0.5% of the total updates, the top five contributed over 51% of the total update, indicating a heavily skewed update contributions across collectors

fewer than 0.01% during the 13-year period. In November 2021, Perth reached an all-time peak of 74.3 billion updates, resulting in 42.9 GB of compressed files (75% of total updates for that month), and ≈3% of the total updates observed across all collectors over the entire 13-year period. Similarly, *rvs3* contributed 14.1% of the total updates, with the majority consisting of high-frequency, repeated updates. Interestingly, the spikes across collectors occurred independently of each other on different days and times, indicating that these bursts are not necessarily synchronized or coordinated by a single global event and thus also not specific to a single collector.

B. Analysis of the top peer contributors over 13 years

Between 2012 and 2025, RouteViews recorded 2.6 trillion updates from 1,066 collector–peer pairs representing 649 unique ASNs (Figure 1) [6]. Some ASNs peer with multiple collectors (≈27% of all peers), with the most connected ASN peering at 21 collectors. The top 5% of peers contributed 1.5 trillion (55.86%) of the total updates. The top 5% bin, on average, contains about 16 peers per day, which together recorded a daily mean of 306.6M updates. The peers in the 75–95% bin accounted for 674.1 billion updates (25.54%) of the total updates, with a daily average of 140M updates, ≈2M updates per peer. In contrast, the lower 50% of peers generated 136.5B (5.17%) of the total updates from an average of 153 peers per day. A peer in the top 5% on average recorded 16.4M updates per day compared to 169.5K for peers in the lower 50%. Finally, the median and mean daily shares of 43.27% and 45.01% for the top 5% of peers across the 13 years indicate that the concentration of updates among these peers is common and not driven by outliers. Additionally, on 713 days (14.83% of all days), the top 5% of peers contributed over 60% of all updates. These findings suggest that a small fraction of peers is responsible for a significant share of updates.

Figure 1 (bottom) shows the update contribution of the top 11 collector-peer pairs and the remaining aggregated pairs (others) on a daily basis. The top 11 collector peers contributed a daily mean of 17.39% and a median of 12.57% of all updates, which is a result of their periodic bursts of updates. Notably, these top 11 collector peers collectively accounted

for approximately 90% of all observed updates on March 20, 2022. Out of 1,066 unique collector peer pairs observed over the 13-year period, the top 5 collector peers (0.47% of all peers) alone contributed 674.9B updates, which is 26.67% of the total updates. AS140627 alone accounted for ≈14% of all updates and even contributed over two-thirds (69.24%) of the total updates between October 2021 and March 2022. It was thereby the top-ranked peer by update volume on 209 separate days. Additionally, AS140627 had a median daily share of 73.10% of all updates and a peak of 88% of an entire day's update share during this interval. We observed similar anomalous behavior among AS58511, with 7.40% of all updates, and AS34968 accounted for 2.64% of all updates. In summary, a small fraction of collectors and their peers accounted for a disproportionately large share of the updates during our study period. Future studies looking into BGP update behavior should account for the potential bias introduced by a few highly active peers.

C. Assessing the update distribution among sessions, prefixes, and AS paths over one month period

1) Dynamics of skewed update contributions across sessions: The results derived from Section IV-C show that a small fraction of the sessions are responsible for most updates observed among the collectors (Figure 3). For example, while the top 19 sessions (2% of all sessions) contributed more than two-thirds of the one-month update volume, a single session of AS140627 alone accounted for over half of the total updates during December 2021, with a mean contribution of 28.67% per minute, peaking on some days at approximately 82% of all updates during the month. The remaining 98% of the sessions contributed only 28.68% of the total updates for the month, indicating that a small fraction of sessions primarily drives the noise observed at the collectors. We found uneven distributions of updates across sessions. The monthly Gini score (0.825) and per-minute score (0.85) both indicate that a small number of sessions generate the majority of updates, while most contribute few.

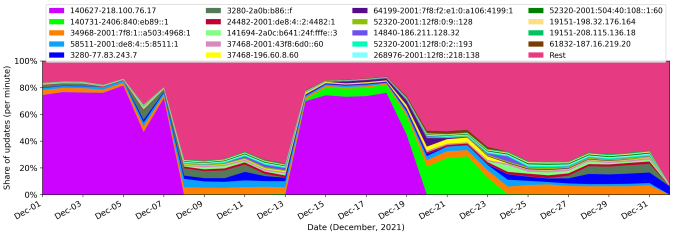


Fig. 3: Variability of update distributions among BGP sessions: While most sessions remained relatively stable, 140627-218.100.76.17 intermittently dominated during the month, fluctuating around an average per-minute share of 28.67%.

We took a closer look at December 2024 using the method described in Section IV-C. The results show two sessions (58511-2001:7f8:4::e48f:1 and 58511-2001:7f8:4:1::e48f:1) that exhibited unusually and persistently erratic behavior across the entire month, with some prefixes announced more than 2M times. These two sessions accounted for a mean of 137M and 146M daily updates, mostly from a small number of prefixes, primarily IPv6. The per-session variability and the Gini equality test show skewed contributions of update volume to a few sessions during this month.

2) Update distribution across prefixes and AS paths: We aimed to understand how the disproportionate contribution of a few sessions to the overall update volume was reflected at the prefix and AS path levels in the global routing table. For each prefix, we computed the total number of updates and ranked prefixes from the highest to the lowest contributors. We then identified the number of prefixes required to account for different update percentiles cumulatively. Our findings revealed that a few prefixes accounted for the majority of the updates. For example, while roughly 17% of prefixes were responsible for 90% of updates, only about 0.3% of prefixes accounted for 25% of all updates in Dec. 2021. Of the 110M AS paths in our dataset, only 2 AS paths accounted for 25% of the total updates. These two paths, on average, propagated about 45K times more updates than the remaining paths.

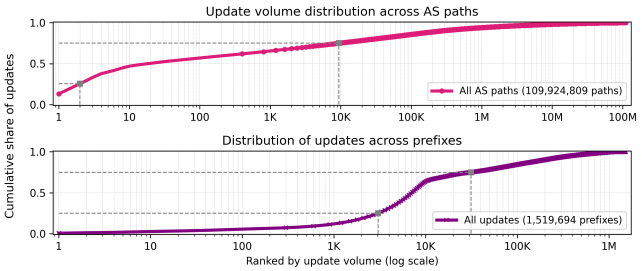


Fig. 4: Heavily skewed distribution of BGP updates across prefixes and AS paths. 75% of the total updates originated from fewer than 3% of the prefixes and a few AS paths.

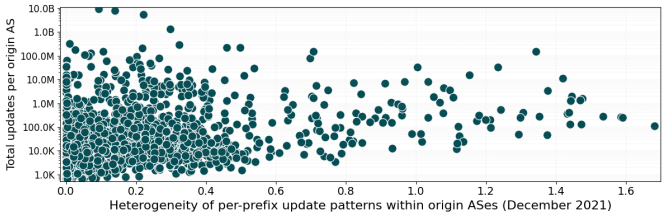
Hence, prefixes or origin ASes are not inherently noisy but rather the routers on the path they traverse. For instance, 50% of the total updates in December 2021 came from only 19 AS paths, and this disproportionate concentration of updates in a small set of paths could have been due to faulty routers or misconfigurations along these paths [17]. Section V-E also shows that a prefix could be noisy at one collector and not at others. We applied the method discussed in Section V-C to validate the skewness observed in the update distribution by prefix. We found a Gini coefficient of ≈0.88 for prefixes and 0.99 for AS paths, respectively, which substantiates our observations in Figure 4 and our argument that a few AS paths account for almost all the updates.

D. Prefix announcement variability across origin ASes

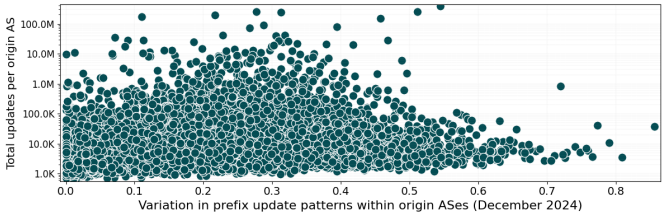
While some origin ASes exhibited uniform and consistent behavior among all their originated prefixes, others displayed erratic and uneven announcement patterns among a small fraction of their originated prefixes. Therefore, the repeated high-frequency updates for specific prefixes could be due to instabilities along the AS paths they traverse, rather than from

the prefixes or their origin ASes. We refer to prefixes that produce a disproportionately high volume of updates relative to other prefixes of the same origin as *noisy* (Section II-C). We label prefixes of the same origin with consistently low update counts over the same time period as *quiet*. Finally, the term *uniform announcement pattern* describes an origin AS whose prefixes exhibited consistent update behavior, such as all prefixes generating uniformly high or low update counts. We evaluated the variability of per-prefix updates within origin ASes using the prefix-AS-path tuples (Section IV-D) that were announced more than 100 times.

We extracted all unique prefix-AS-path tuples and computed their total update counts across the month. We used these values to calculate the average and standard deviation of each unique prefix AS path tuple. We computed the coefficient of variation ($cv = \frac{\sigma}{\mu}$) [19] of these daily counts, which normalizes variability relative to the mean, allowing us to fairly capture how relatively erratic is the behavior of each prefix across varying origin ASes with uneven announcement patterns. This metric is scale-invariant compared to raw standard deviation or variance, which would be biased toward origin ASes with larger absolute counts. Next, we computed the standard deviation of the coefficient of variation values for all prefixes, denoted as $\sigma_{cv}^{\text{AS}} = \text{std}(cv_1, \ldots, cv_n)$, which summarizes the variability of update activity across prefixes within each origin AS (Figure 5a), such as whether their prefixes were uniform, consistent, or inconsistently updated.



(a) Almost 80% of ASes exhibited a uniform announcement pattern of either noisy or quiet, while the remaining 20% showed uneven announcement behavior among prefixes of the respective ASes.



(b) AS8359 exhibited high per-prefix variability (0.545) with update volumes exceeding 250M, implying that a few prefixes generate disproportionately many updates, while others remain relatively quiet.

Fig. 5: Announcement patterns and prefix update dynamics across different classes of origin ASes.

Of the total 3K origin ASes (Figure 5a), 1.43K (48.8%) exhibited uniform announcement behavior with stable (*cv* is less than 0.25) and low update volume across their prefixes. In contrast, 874 ASes (29.7%) consistently exhibited high-volume announcements among their prefixes. 472 ASes (16.0%) showed high variability in their prefix announcements. Finally, 160 (5.4%) of the origin ASes had a single prefix that accounted for more than 70% of their update volume.

The ASes showing high variability in their prefix announcement support our hypothesis that, in these cases, the source of variability lies in *routers* on the AS paths that the prefixes propagate, rather than in the prefixes or origin ASes themselves. We performed the same investigation for December 2024 and observed similar trends. For instance, most origin ASes exhibited low variability across their prefixes with relatively low update counts, as shown in Figure 5b.

In conclusion, noisy prefixes cannot be attributed solely to the origin ASes or the prefix itself, but rather to a combination of other factors, such as misconfigurations or faulty routers on the ASes along the AS paths.

E. Visibility of erratic prefixes across collectors

Our results derived from Section IV-D reveal three distinct announcement patterns, including near-total asymmetry (same sets of prefixes appearing noisy at one collector but quiet in others), heavy skew, and local concentration of announcements for the 841 prefixes (Figure 6). First, we observed consistently higher update counts for all 841 prefixes from the Perth collector, with a mean update volume of 3.3M per prefix. At the same time, Linx and eqix recorded a mean of 253K and 552K, respectively, for the same 841 prefixes in December 2021, indicating a clear contrast in behavior across collectors.

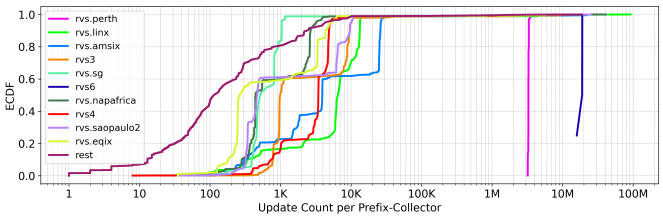


Fig. 6: Dynamics of noisy prefixes across collectors. Most large collectors, such as Linx and amsix, fully observed 100% of highly-announced prefixes, with significantly fewer updates compared to updates for those prefixes from Perth.

Additionally, the 25th percentile of updates per prefix is below 5K in most collectors, except for Perth, which recorded an order of magnitude higher updates per prefix, at 3.2M. We further observed nine prefixes that were consistently noisy across all 10 benchmark collectors (except the aggregated set). However, the intensity varied, with Linx reaching 9.1M updates for one prefix in December 2021.

Similarly, our analysis reveals an interesting behavior of *rvs6* on only 4 (0.4756%) of the 841 prefixes that were not noisy in any of the top-10 benchmark collectors (Section IV-D). AS14210 (a CDN/streaming provider) and AS208046 announced these four IPv6 prefixes 15–19M times in December 2021. We confirmed with RouteViews operators that rvs6 is an IPv6-only collector (34 IPv6 peers), which produced 2.67 billion updates in December 2021. A single peer

of the rvs6 collector contributed 1.96 billion updates (≈73.4% of the total updates at rvs6), while the top five peers together accounted for 82.51%. In conclusion, the sustained, extremely high activity at the Perth and rvs6 collectors makes them clear outliers relative to the other collectors. Notably, 98.93% of the heavily updated prefixes at the Perth and rvs6 collectors exhibited stable behavior in the remaining collectors.

F. Characterizing repetition in MRT files

While most files with fewer unique prefixes exhibited lower levels of repetition, our metric indicates extreme repetition in a few small files. Although these files contain highly noisy updates, they appear with relatively low scores because of their small size (Figure 7). Nevertheless, our metric identifies these small files as heavily repetitive. The subsequent sections summarize the characteristics of all MRT files across different repetition levels.

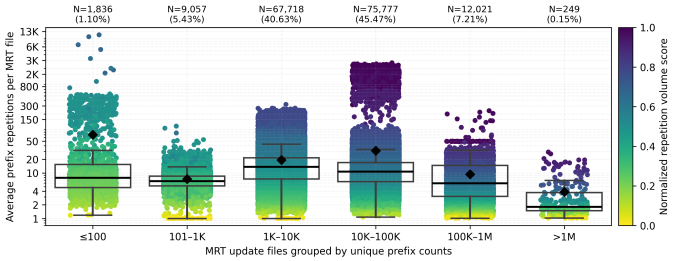


Fig. 7: Announcement redundancy and irregularities in MRT files. Distributions of 166.7K MRT files (after prefiltering) across bins of unique prefix counts within the files.

1) MRT files with fewer than 100 unique prefixes: These files have a mean size of 7.53 KB, with an average repetition of 69.95 and a standard deviation of 422.56. This bin includes MRT files that recorded a mean prefix repetition of 10.98K, 9.95K, and 6.64K with only 11, 12, and 18 prefixes, respectively. On average, a single prefix contributed 25% of the updates in most files in this bin, while some files had a single prefix that accounted for all their updates.

2) The churn zone (10K–100K unique prefixes): These files include excessively repeated updates with a low withdrawal-to-announcement ratio, indicating that the repeated announcements were not due to topological changes. We observed a low average repetition of 31 per prefix, which is mainly due to a large portion of the prefixes within some files remaining quiet. However, some files in this bin have single prefixes repeated more than 2.7K times, with fewer than 2% of the prefixes accounting for 90% of the updates in some files.

3) The chaos region (100K–1M unique prefixes): Even though the MRT files within this range have a mean file size of only 10.57 MB, this range includes files as large as 140 MB (compressed). The average prefix repetition is 9.43 per prefix, with a mean maximum repetition of 234.64. Although these files might appear relatively stable, that is far from reality. For example, the top of the 100K-1M bin (purple) includes 45 MRT files from a significant single-day event (9:45 – 20:45), each averaging 138 MB in compressed size, with 730K unique prefixes and 36.2M updates. We saw an even distribution of announcements across these prefixes because ≈80% of the prefixes in each file were required to reach 90% of the update volume, which suggests a repeated RIB transfer over time.

4) The extreme cases (greater than 1M unique prefixes): Most files in this bin require at least 81% of their prefixes to account for 90% of their total updates, most of which are stable announcements with a mean average repetition of 3.8 and a maximum repetition of 28.9 for some files. These files are mostly large, not due to noise, but because of their significant number of unique prefixes, BGP communities, and AS paths.[1]

G. Exploring potential storage gains from pruning method

The original MRT files for December 2024 require more than 198.8 GB of compressed storage. Applying the pruning method in Section IV-F reduces the size to 55.9 GB, saving 142.9 GB of storage space (72% overall reduction compared to the original size). The MRT files from the noisiest collector (Linx) accounted for 59.1 GB of storage space, comprising repeated updates with no attribute changes and few withdrawal messages. Our method reduced this to 9.4 GB (an 84% reduction rate for MRT files collected at Linx), equivalent to 16.7 MB per file on average. We observed similar trends for AMSIX and RouteViews3, each exceeding 10.3 GB. The high reduction rate for some collectors suggests that their updates are largely repetitive, and pruning can yield storage gains for longitudinal BGP studies requiring terabytes of storage.

1) Future work: validation: CAIDA utilized MRT files observed from route collectors to generate key datasets, including AS Relationships and AS2Org, which have been widely used in numerous measurement studies [4]–[6], [20]. We plan to use our pruned MRT files to regenerate equivalent datasets and validate whether our datasets are consistent with those obtained by CAIDA using the full MRT files. If the outcome of our validation is consistent, we can argue that storing and processing all repeated updates provides us with limited additional value for the selected studies.

VI. Discussion

A. Lesson learned: Rethinking BGP update dynamics

High-frequency repeated announcements, mostly semantically redundant, are prevalent, unpredictable, and often stem from a small fraction of collector peers, sessions, prefixes, and AS paths. Solely attributing noise to prefixes or origin ASes oversimplifies the complexity of prefix announcements. Our analysis shows that the primary sources of this noise are buggy or misconfigured routers of the ASes along the AS path. However, we have also found cases where the origin ASes directly peer with the collector peer, and both ASes could generate the noise. Understanding noise and its origins is essential for accurately interpreting BGP measurements

[1]We found 75 MRT files that did not match the expected 15-minute dump intervals, suggesting delays of 2–55 minutes, which could have been due to delays in the syncing and compressing process of these large MRT files.

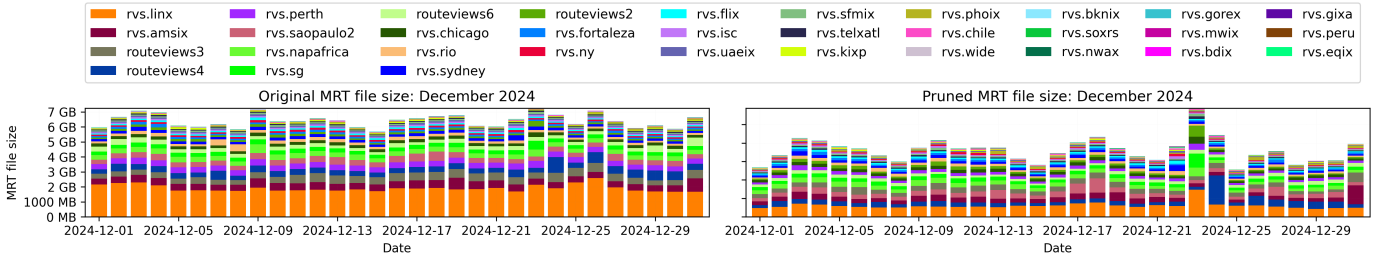


Fig. 8: Daily MRT file size share for December 2024. Pruning reduced total storage by 143 GB (72% overall reduction)

B. Impact of noise on Internet science

The Internet measurement research community relies on diverse sets of collectors and peers to gain a representative and broad view of routing dynamics. Our findings reveal a highly skewed distribution of updates, with a small fraction of collector peers and sessions contributing over 50% of the data [17], most of which are repeated updates with unchanged attributes or only community changes [4]. The remaining majority of collector peers and sessions contributed comparatively few updates.

Large repeated bursts of updates from a few collector peers and sessions could also pose a challenge for studies relying on update bursts as an indicator of hijacks and route leaks [21]. Hence, measurement studies should consider these update bursts from a small set of peers to avoid distorting their broader interpretation of Internet routing behavior.

C. Toward efficient BGP data collection and usage

RIPE RIS and RouteViews have expanded their peering relationships to enhance data richness and Internet visibility, resulting in a substantial increase in data volume. However, our results revealed that there are opportunities for both route collectors and researchers to optimize how BGP data is collected, stored, and processed. For instance, our repetition volume score (RVS), combined with our exploratory pruning methods, suggests that we can gain substantial computational and storage benefits while retaining meaningful information. Therefore, while RouteViews' commitment to archiving all BGP data benefits the community, excessive updates create storage and management challenges that we hope to help them navigate in the future.

D. The need for collaboration with the operators

Recently, a RouteViews operator has speculated that a small number of peers generating disproportionately high volumes of repeated updates from specific locations may be due to infrastructural issues, broken BGP implementations, or BGP attribute changes [17]. This uncertainty underscores the need for closer collaboration with operators to identify the root causes of these frequent updates. Such collaborations would help both route collector operators to accurately pinpoint the offenders responsible for these repeated updates and devise methods to address them closer to the source.

VII. Related Work

Over the past few years, the research community has made progress in understanding and mitigating duplicate or redundant BGP announcements [4]–[6], [15], [22], [23]. However, the route collectors continue to observe a considerable share of redundant updates driven by path instabilities and BGP communities [4]. We highlight the most relevant studies.

Labovitz et al. [24] were among the first to study redundant announcements, demonstrating that routers generated millions of unnecessary daily updates from duplicate withdrawals, thereby consuming resources without any topological changes. Later, Park et al. [23] defined duplicates as updates with identical attributes and, using RouteViews and RIS data, reported an average redundancy of 13

Hauweele et al. [22] conducted controlled experiments and identified route flapping, internal attribute handling, and Adj-RIB-Out (routers have no record of prior announcements and repeatedly re-sends the same updates) trade-offs as the leading causes of duplicate updates. They used BGP traces to evaluate their lightweight, memory-efficient cache, inserted at the router's output, to mitigate these duplicates, reducing it from 98.36% to 5.83%. However, their caching method underperformed during large-scale update bursts.

Ariemma et al. [15] examined 30 billion BGP updates from 172 vantage points across the Internet. They found that nearly 30% of long-lasting sequences of updates (announcements lasting over a week) involved more than 200K unique prefixes. They utilized a Discrete Wavelet Transform and a custom clustering algorithm to demonstrate recurring behavioral patterns, suggesting that a substantial fraction of BGP instability is prolonged and exhibits regular patterns, rather than being short-lived noise, as shown in our study.

In 2022, Alfroy et al. [5] showed that BGP data is highly redundant and proposed MVP to mitigate the excessive growth of redundancy across vantage points. They defined redundancy at the prefix, AS-path, and community levels and introduced algorithms to rank VPs based on marginal information gain. In 2024, Alfroy et al. [6] extended [5] and proposed GILL. This system uses an overshoot-and-discard collection approach to maximize visibility while filtering out redundant updates from highly correlated vantage points, thereby enhancing hijacking detection and AS-relationship inferencing without increasing data volume.

Our study offers a decade-long longitudinal analysis, providing a broader perspective on high-frequency, repeated updates. It characterizes their distribution across collectors, peers, sessions, and prefixes, highlighting how a small set of peers generates the majority of this noise.

VIII. Conclusion

This study takes the first step in quantifying the update shares among RouteViews collectors' peers. We show that only the top 5% of collector peers contributed 1.5 trillion updates (55.86%) of the total updates across 13 years. Our results show that mostly a single or a few noisy sessions account for most repeated updates during our study period. Our findings suggest that the source of these high-frequency repeated updates is not always the AS originating the prefix or the route collector peer, but often a router on the AS path between the origin and the peer. Based on our observations, we have explored a sliding window pruning algorithm to quantify the potential processing and storage gains of filtering high-frequency redundant updates. It yields first promising results that we plan to explore in future work.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable feedback. This work was carried out as part of the Network Security program at the Twente University Centre for Cybersecurity Research (TUCCR) under grant number 20003215. Cristian Hesselman's work was also part of the CATRIN and UPIN projects, both of which received funding from the Dutch Research Council (NWO).

References

[1] RIPE. (2025) Routing information service (RIS). [Online]. Available: https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/

[2] O. Univ. (2025) RouteViews – university of oregon RouteViews project. [Online]. Available: https://www.routeviews.org/routeviews/

[3] L. Blunk, C. Labovitz, and M. Karir, "Multi-Threaded Routing Toolkit (MRT) Routing Information Export Format," RFC 6396, Oct. 2011. [Online]. Available: https://www.rfc-editor.org/info/rfc6396

[4] T. Krenc, R. Beverly, and G. Smaragdakis, "Keep your communities clean: exploring the routing message impact of BGP communities," in *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies*, ser. CoNEXT '20. Association for Computing Machinery, 2020-11-24, pp. 443–450. [Online]. Available: https://doi.org/10.1145/3386367.3432731

[5] T. Alfroy, T. Holterbach, and C. Pelsser, "MVP: measuring internet routing from the most valuable points," in *Proceedings of the 22nd ACM Internet Measurement Conference*, ser. IMC '22. Association for Computing Machinery, 2022-10-25, pp. 770–771. [Online]. Available: https://doi.org/10.1145/3517745.3563031

[6] T. Alfroy, T. Holterbach, T. Krenc, K. C. Claffy, and C. Pelsser, "The next generation of BGP data collection platforms," in *Proceedings of the ACM SIGCOMM 2024 Conference*, ser. ACM SIGCOMM '24. New York, NY, USA: Association for Computing Machinery, 2024-08-04, pp. 794–812. [Online]. Available: https://dl.acm.org/doi/10.1145/3651890.3672251

[7] C. Testart, P. Richter, A. King, A. Dainotti, and D. Clark, "Profiling BGP serial hijackers: Capturing persistent misbehavior in the global routing table," in *Proceedings of the Internet Measurement Conference*, ser. IMC '19. New York, NY, USA: Association for Computing Machinery, 2019-10-21, pp. 420–434. [Online]. Available: https://dl.acm.org/doi/10.1145/3355369.3355581

[8] E. Aben. (2020-10-07) Route collection at the RIPE NCC - where are we and where should we go? [Online]. Available: https://labs.ripe.net/author/emileaben/route-collection-at-the-ripe-ncc-where-are-we-and-where-should-we-go/

[9] R. Fontugne. (2024-02-01) Noisy prefixes in BGP. [Online]. Available: https://seclists.org/nanog/2025/Feb/28

[10] P. Smith. (2024-02-01) Re: Noisy prefixes in BGP. [Online]. Available: https://seclists.org/nanog/2025/Feb/36

[11] K. Cereceda, J. But, and P. Branch, "Experimental observations and analysis of BGP route flapping dynamics," in *2025 International Conference on Information Networking (ICOIN)*, 2025-01, pp. 105–110, ISSN: 2996-1580. [Online]. Available: https://ieeexplore.ieee.org/document/10993166/

[12] F. Stöger, H. Birge-Lee, G. Giuliari, J. Subira-Nieto, and A. Perrig, "BGP vortex: update message floods can create internet instabilities," in *Proceedings of the 34th USENIX Conference on Security Symposium*, ser. SEC '25. USENIX Association, 2025-09-08, pp. 3613–3629.

[13] B. Cartwright-Cox. (2023) Grave flaws in BGP error handling. [Online]. Available: https://blog.benjojo.co.uk/post/bgp-path-attributes-grave-error-handling

[14] D. dos Santos, S. Guiot, S. Dashevskyi, A. Amri, and O. Kerro. (2023-08-09) Route to bugs: Analyzing the security of bgp message parsing. [Online]. Available: https://www.blackhat.com/us-23/briefings/schedule/index.html#route-to-bugs-analyzing-the-security-of-bgp-message-parsing-32162

[15] L. Ariemma, A. Dell'Orco, S. Liotta, M. Candela, and G. Di Battista, "Long-lasting sequences of BGP updates," *Computer Networks*, vol. 220, p. 109481, 2023-01-01. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128622005151

[16] L. Miller and C. Pelsser, "A Taxonomy of Attacks Using BGP Blackholing," in *Computer Security – ESORICS 2019*, K. Sako, S. Schneider, and P. Y. A. Ryan, Eds., vol. 11735. Cham: Springer International Publishing, 2019, pp. 107–127, series Title: Lecture Notes in Computer Science. [Online]. Available: http://link.springer.com/10.1007/978-3-030-29959-0_6

[17] P. Smith. (2025-02-09) Re: Noisy prefixes in BGP. [Online]. Available: https://seclists.org/nanog/2025/Feb/39

[18] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP misconfiguration," in *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, ser. SIGCOMM '02. Association for Computing Machinery, 2002-08-19, pp. 3–16. [Online]. Available: https://dl.acm.org/doi/10.1145/633025.633027

[19] W. Shao, L. Iannone, J.-L. Rougier, F. Devienne, and M. Viste, "Scalable bgp prefix selection for effective inter-domain traffic engineering," in *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*. IEEE Press, 2016, p. 315–323. [Online]. Available: https://doi.org/10.1109/NOMS.2016.7502827

[20] T. Alfroy, T. Holterbach, T. Krenc, K. Claffy, and C. Pelsser, "Internet science moonshot: Expanding BGP data horizons," in *Proceedings of the 22nd ACM Workshop on Hot Topics in Networks*. ACM, 2023-11-28, pp. 102–108. [Online]. Available: https://dl.acm.org/doi/10.1145/3626111.3628202

[21] P. Moriano, R. Hill, and L. J. Camp, "Using bursty announcements for detecting BGP routing anomalies," *Computer Networks*, vol. 188, p. 107835, 2021-04-07. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128621000207

[22] D. Hauweele, B. Quoitin, C. Pelsser, and R. Bush, "What do parrots and BGP routers have in common?" *SIGCOMM Comput. Commun. Rev.*, vol. 46, no. 3, pp. 2:1–2:6, 2018-07-27. [Online]. Available: https://dl.acm.org/doi/10.1145/3243157.3243159

[23] J. H. Park, D. Jen, M. Lad, S. Amante, D. McPherson, and L. Zhang, "Investigating occurrence of duplicate updates in BGP announcements," in *Passive and Active Measurement*, A. Krishnamurthy and B. Plattner, Eds. Springer Berlin Heidelberg, 2010, vol. 6032, pp. 11–20, series Title: Lecture Notes in Computer Science. [Online]. Available: http://link.springer.com/10.1007/978-3-642-12334-4_2

[24] C. Labovitz, G. Malan, and F. Jahanian, "Internet routing instability," *IEEE/ACM Transactions on Networking*, vol. 6, no. 5, pp. 515–528, 1998-10. [Online]. Available: http://ieeexplore.ieee.org/document/731185/