Estimating the Impact of BGP Prefix Hijacking

Pavlos Sermpezis[¶], Vasileios Kotronis[†], Konstantinos Arakadakis[†,‡], Athena Vakali[¶]
[¶]*Aristotle University of Thessaloniki, Greece;* [†]*FORTH-ICS, Greece;* [‡]*University of Crete, Greece*

***Abstract*—BGP prefix hijacking is a critical threat to the resilience and security of communications in the Internet. While several mechanisms have been proposed to prevent, detect or mitigate hijacking events, it has not been studied how to accurately quantify the impact of an ongoing hijack. When detecting a hijack, existing methods do not estimate how many networks in the Internet are affected (before and/or after its mitigation). In this paper, we study fundamental and practical aspects of the problem of estimating the impact of an ongoing hijack through network measurements. We derive analytical results for the involved trade-offs and limits, and investigate the performance of different measurement approaches (control/data-plane measurements) and use of public measurement infrastructure. Our findings provide useful insights for the design of accurate hijack impact estimation methodologies. Based on these insights, we design (i) a lightweight and practical estimation methodology that employs ping measurements, and (ii) an estimator that employs public infrastructure measurements and eliminates correlations between them to improve the accuracy. We validate the proposed methodologies and findings against results from hijacking experiments we conduct in the real Internet.**

I. Introduction

The Border Gateway Protocol (BGP) is used by the Autonomous Systems (ASes) to establish routing paths in the Internet. Due to its distributed nature and lack of authentication in the exchanged information, BGP is susceptible to illegitimate route advertisements. *BGP prefix hijacking* is the most prominent example. Numerous hijacking events with global impact on the availability and confidentiality of communications, e.g., [1], [2], [3], [4], and concerns expressed by network operators [5], show that BGP prefix hijacking is a common and persistent threat to the Internet ecosystem.

Defenses against BGP prefix hijacking consist of (i) prevention measures, such as prefix filtering or RPKI, which block the propagation of illegitimate routes [6], [7], [8], and (ii) detection techniques [9], [10], [11], [12] that inform operators to proceed to counteractions or trigger mitigation techniques [9], [13]. The efficiency that these mechanisms are expected to have (on average) under various hijacking scenarios has been studied in literature. However, *when a hijacking event takes place, there do not exist techniques that can provide accurate information about its actual impact.*

Knowing the impact of a hijack is important for several reasons: (i) Inform operators about the effect of an ongoing hijack (e.g., global, limited to a few ASes). This information may drive their actions, for example, to select mild (e.g., ask other operators to filter hijacked routes) or more aggressive (e.g., prefix de-aggregation) countermeasures to mitigate the hijack, based on its impact [9]. (ii) Evaluate the actual efficiency of a mitigation action (which is deemed as a key priority by operators [5]). Knowing the remaining impact of the hijack after the mitigation, can help operating and business decisions; for instance, to decide whether further actions are required, or to assess the cost/value of a paid service (e.g., blackholing [14] or MOAS announcements [9], [13]).

These reasons highlight the need for designing hijack impact estimation techniques, which could be incorporated in existing defense systems and become a valuable asset for network operations and security. In this direction, *the goal of this work is to study the problem of estimating the impact of an ongoing hijack through measurements, and take the first steps towards designing accurate hijack impact estimation methodologies.*

Specifically, we follow an approach comprising analysis, simulations, and real experiments and measurements: (i) We analytically study fundamental aspects of the hijack impact estimation, and derive results that identify limits and quantify trade-offs on the accuracy of different estimation methods. (ii) We employ realistic simulations to create datasets of hijack incidents[1], based on which we investigate the performance that can be achieved by using different measurement techniques (control/data plane) and available resources (RouteViews, RIPE RIS, RIPE Atlas). (iii) We conduct controlled hijacking experiments and extensive network measurements in the real Internet to validate our theoretical/simulation findings.

The main contributions of this work are summarized as:

- ***Understanding of the hijack impact estimation.*** We study the accuracy of the hijack impact estimation under different types of measurements (§III). We show that very high accuracy is possible by sampling/measuring any network in the Internet (e.g., $\sim$1% estimation error with 1000 samples), while using public measurement infrastructure (RIPE, RouteViews) results in an estimation error of around 10%. The root cause of this error is the correlation between the locations of public infrastructure monitors.
- ***Design of efficient estimators.*** Motivated by our findings, we propose efficient estimation methodologies with and without using public measurement infrastructure. We first propose an estimator based on ping campaigns (§IV), which does not rely on public infrastructure and can be implemented by any network. We find that by pinging a couple

This research is co-financed by Greece and European Union through the Operational Program Competitiveness, Entrepreneurship and Innovation under the call RESEARCH-CREATE-INNOVATE (project T2EDK-04937), and the European High-Performance Computing Joint Undertaking (GA No. 951732).

ISBN 978-3-903176-39-3

[1]To the best of our knowledge, there are no datasets with real data from past hijacking events that could enable an extensive and in depth investigation.

of reachable IP addresses in a few hundreds of ASes is enough for achieving very low errors. Then, we design an estimator based on public infrastructure measurements (§V), which employs statistical learning to eliminate the effect of correlation in measurements, and achieves an accuracy comparable to the (best performing) ping-based estimator.

To facilitate future research we make our code and data from our experiments available in [15].

II. Preliminaries

We first define the main quantities of the considered problem (§II-A), provide an overview of the different hijack types and their impact characteristics (§II-B) and the available network measurement techniques and services (§II-C), and present the simulation (§II-D) and experimental (§II-E) methodology used in the paper.

A. Definitions: Quantities and Metrics

Infected AS: an AS is infected when it routes its traffic to or through the hijacker's AS.

Hijack impact, I: fraction of ASes that are *infected*, $I \in [0, 1]$.

Monitor: an AS for which it can be known (e.g., after a measurement) if it is *infected* or not.

Impact estimator: a methodology that uses a set of *monitors* and provides an estimation of the *hijack impact*.

Let I be the actual impact of a hijack, and an estimator $\mathcal{E}$ whose estimation is $\hat{I}_{\mathcal{E}}$. The main metrics to characterize the performance of an estimator is the *bias* and the *Root Mean Square Error (RMSE)*:

$$Bias_{\mathcal{E}} = E[\hat{I}_{\mathcal{E}} - I] \qquad RMSE_{\mathcal{E}} = \sqrt{E[(\hat{I}_{\mathcal{E}} - I)^2]}$$

The bias quantifies how far the expected value of the estimator is from the actual value, and the RMSE quantifies the accuracy of the estimator. The desired characteristics for an estimator are to be *unbiased* (zero bias) and have low RMSE.

B. Hijack Types and Impact Characteristics

We consider an AS that owns and announces a prefix IP_*; we denote this AS as AS_V and call it "victim AS". Let the best path to IP_* for an AS_X be

$$[AS_X, AS_Y, ..., AS_V | IP_*]$$

where AS_X has learned this path through its neighbor AS_Y. A hijack takes place when another AS announces an illegitimate path for the prefix IP_* or for a more specific prefix. This is the "hijacker AS" and we denote it as AS_H. The hijacker's announcement may propagate to the Internet and "infect" some ASes; the extent of the infection is the hijack impact.

There are different ways to perform a hijack and their impact may vary. Below, we present a taxonomy of hijacks [9], [16] and discuss their impact characteristics.

Origin-AS (Type-0) or Fake-path (Type-N, $N \geq 1$) hijack. In the *origin-AS* hijack the AS_H originates the IP_* as its own, while in the *fake-path* case the AS_H announces a fake path to the IP_* to its neighbors; e.g., for an AS_X

Type-0: $[AS_X, AS_Y, ..., \mathbf{AS_H} | IP_*]$
Type-N: $[AS_X, AS_Y, ..., \mathbf{AS_H}, AS_Z, ..., AS_V | IP_*]$

where the link $AS_H - AS_Z$ is fake; the number N denotes that the hijacker's ASN appears in the N^{th} hop away from the origin AS.

Impact characteristics: In Type-N cases the hijacker originates a longer path than in Type-0 (i.e., with N extra hops). Thus, the paths to the hijacker are longer and less preferred by some ASes; this results in a lower impact for higher N [9], [8]. Note that this holds when no proactive measures, such as RPKI, are deployed; for prefixes protected by RPKI (less than 20% today [17], [18]), the impact decreases in Type-0 attacks due to route origin validation [19], while the impact of Type-N hijacks is not affected since RPKI cannot detect fake links.

Exact prefix or Sub-prefix hijack. The hijacker can perform a Type-0 or Type-N hijack for the same prefix IP_* announced by the victim (*exact prefix*) or for a more specific prefix in IP_* (*sub-prefix*); for example, let the IP_* be the prefix $10.0.0.0/8$, then a sub-prefix hijack takes place if the hijacker announces the prefix $10.0.0.0/9$ (or any $10.0.0.0/d$ with $d \geq 9$).

Impact characteristics: Default routing in BGP prefers paths to more specific prefixes [20]. Hence, the impact of a sub-prefix hijack will be larger than an exact prefix hijack; in fact, a sub-prefix hijack will infect the entire Internet (i.e., impact 100%) unless a proactive or filtering mechanism is applied.

Data-plane traffic manipulation. For all the aforementioned hijack types, the hijacker can manipulate the traffic that it attracts by: (i) dropping it (*blackholing*), (ii) impersonating a service (*imposture*), or (iii) manipulating or eavesdropping it and then forwarding it to the victim (*man in the middle, MitM*).

Impact characteristics: While the traffic manipulation in the data plane by the hijacker does not affect the impact on the control plane (i.e., as defined in this paper), it determines what hijack detection and impact estimation approaches can be applied (data/control plane measurements) as we discuss later.

C. Measuring the Hijack Infection

The hijack impact is determined by the number of infected ASes. Hence, the basic step for an impact estimation is to identify whether an AS/monitor is infected or not. In principle, this can be done by applying any hijack detection method [9], [10], [11], [12] per monitor.

Detection methods are mainly based on three network measurements types. Below, we provide some indicative examples for each type, and discuss their main characteristics. Our first goal in this paper is to investigate how efficient is to use each of these measurement types for impact estimation (see §III).

Route collectors (RC) - BGP routes: A monitor that provides information about its BGP routes (BGP updates or RIBs) can be detected as infected or not (from the AS-path or the prefix in its selected BGP route (see [9] for a comprehensive approach). For example, for a Type-0 hijack , if the first ASN in the path is different than AS_V, then the monitor is infected.

The RIPE RIS [21] and RouteViews [22] projects provide BGP RIBs/updates collected from hundreds ASes. We refer to these ASes that peer with RIPE RIS / RouteViews route collectors and provide BGP feeds as "route collector monitors",

or for brevity "RC". In the paper, we consider a set of 228 RC that consistently provided data in our experiments (see §II-E).

The main characteristics of this approach is that it is based on control-plane information, it is lightweight (requires passive measurements, which can be retrieved from public APIs [23]), and can be real-time since several RC provide live-feed of their BGP updates [22], [24], [25].

RIPE Atlas probes (RA) - traceroutes: Conducting a traceroute from a monitor to the hijacked prefix, returns a path of IP addresses. Mapping the IPs to ASNs, we can infer the AS-path, and thus detect (similarly to the BGP routes) if the monitor is infected. However, in practice the IP to ASN mapping may be inaccurate for some hops, and advanced methods may be needed to avoid path misinformation [26], [27].

The RIPE Atlas [23] platform comprises more than $25k$ probes, *i.e.,* devices able to run traceroutes towards certain Internet destinations. We refer to the set of ASes with at least one RIPE Atlas probe as "RA" monitors, which in our experiments account for 3420 ASes.

This approach combines data plane (traceroute) and control plane (IP-to-ASN mapping) information, and requires active measurements (RIPE Atlas can return a batch of measurements within a few minutes).

Pings: The victim can ping (from its network) an IP address in a remote AS (monitor); if the ping response returns to the victim's network, then the AS can be inferred as not infected (see, e.g., the techniques of [28], [10]). This inference can be correct in blackholing and imposture hijacks, but not in MitM.

It is important to note that while the first two measurement approaches can use only the monitors of the public services (RC and RA), in this latter case any AS with a responsive pingable IP address (i.e., almost every AS) can be a monitor.

Finally, this approach is based on data-plane information, and requires active measurements (whose results can typically be returned within a few seconds)

D. Datasets and Simulation Methodology

To study different impact estimation approaches, we would need ground truth data about hijack events and their impact. However, typically this information is not publicly reported, and detailed datasets do not exist, to the best of our knowledge. Hence, we use realistic simulations to generate datasets of different hijack types.

Specifically, we simulate the Internet routing system using a largely adopted methodology [8], [9], [29]: (i) we use the AS-relationship dataset [30] that contains AS-links and inferred inter-AS economic relationships (*customer to provider*, *peer to peer*), based on which (ii) we build the Internet topology graph representing each AS as a single node (a reasonable assumption for the vast majority of ASes [31]) and (iii) we define the routing policies as in the Gao-Rexford model [32], where an AS prefers routes learned from its customers, then its peers, and then its providers, and (iv) we simulate BGP using the simulator of [29]. For each hijack type, we run 1000 scenarios with different $\{$victim, hijacker$\}$ ($\{$V,H$\}$) pairs. Each RC and RA monitor is represented by the AS that hosts it.

While, admittedly, simulations may not generate the exact impact output per $\{$V,H$\}$ case, they have been shown to capture well the routing decisions for the majority of ASes [33], [29]. In this work, we study the statistical characteristics of impact estimation rather than the per case behavior, and thus the involved uncertainty is not expected to significantly affect our findings. Nevertheless, we also conduct real hijacking experiments in the Internet (§II-E), to validate our methods and findings with real data.

E. Real-world Experiments

We conduct hijacking experiments in the real Internet using the PEERING testbed [34]. PEERING owns ASNs and IP prefixes, and has BGP connections with networks in several locations (*sites*) around the world. The experiments consist of the following steps:

Selection of $\{$V,H$\}$ pair. We create two virtual ASes, assign to them the ASNs 61574 and 61575, and connect them to two distinct sites of the PEERING testbed. We select one of them to be the victim AS (V) and the other the hijacker AS (H).

BGP announcements and Hijacking. We conduct Type-0 hijacks, i.e., we announce the prefix $184.164.243.0/24$ from V, and then announce (i.e., hijack) the same prefix from H.

Impact measurement: pings (ground truth). To measure the impact of the hijack, we perform a ping campaign: We select 46k ASes and ping (from a host within PEERING) a reachable IP address in each of them (see §IV). We monitor through which PEERING site the ping reply returns: if it returns through the H (or, V) site, we consider the corresponding AS as infected (or, not infected). We use this as the ground truth for the hijack impact of each experiment.

Impact measurement: public services. To apply the different measurement approaches (§II-C), we conduct data-plane (traceroutes) and control-plane (BGP updates) measurements after the hijacking announcement: (i) We employ traceroutes from RIPE Atlas probes towards the announced prefix. We check in the traceroute the last IP address before it enters PEERING. We map this IP address to an AS (using the prefix-to-AS dataset of §IV), and if the ASN belongs to an upstream provider of the H (or, V) site, then we infer that the AS of the RIPE Atlas probe is infected (or, not infected). (ii) Using CAIDA's BGPStream tool [25] we collect BGP updates received by RouteViews and RIPE RIS monitors. From the AS paths in the BGP updates, we extract the origin ASNs and use them to infer to which site the monitor AS routes its traffic. For example, from the AS path $[AS_X, AS_Y, \ldots, AS_H]$, we infer that the monitor AS_X is infected.

In total, we considered a set of 6 PEERING sites that: were responsive at the time of our experiments, their BGP announcements propagated to the entire Internet, and they were reachable through data-plane measurements (pings, traceroutes) from the majority of ASes. We considered all possible combinations of pairs $\{$V,H$\}$ for these sites. Omitting the experiments in which the hijack impact was trivial (100% or 0%) or very small/large ($> 97\%$ or $< 3\%$), we end up in a set of 22 "valid" experiments with different $\{$V,H$\}$ pairs.

III. Understanding the Impact Estimation

In this section, we aim to understand the problem of hijack impact estimation through measurements, and provide useful insights for the design of practical estimation methodologies.

A. Naive Impact Estimation (NIE)

The most intuitive approach to estimate the impact of a hijack is to measure a set of monitors, and estimate it as the fraction of infected monitors. We refer to this approach as the *Naive Impact Estimation*.

Definition 1 (Naive Impact Estimator (NIE))**.** *Let a set of monitors $\mathcal{M}$ ($|\mathcal{M}| = M$), and an indicator m_i denoting whether monitor $i \in \mathcal{M}$ is infected ($m_i = 1$) or not ($m_i = 0$). The Naive Impact Estimator estimates the hijack impact as*

$$\hat{I}_{NIE(\mathcal{M})} = \frac{1}{M} \sum_{i \in \mathcal{M}} m_i \tag{1}$$

The NIE can be used with any type of measurements (BGP routes, traceroutes, pings) that can provide information to calculate the indicator m_i. In the following we study the properties and accuracy of NIE, under different types of measurements and monitor sets.

B. Accuracy of the NIE

1) NIE with Random Set of Monitors.

In the following theorem, we prove that, when the set of monitors $\mathcal{M}$ is randomly selected, the NIE is an *unbiased estimator*, and we derive an expression for its RMSE that is a function of the number of monitors M and the hijack type.

Theorem 1. *Under a randomly selected set of monitors $\mathcal{M}$, the bias and root mean square error of NIE are given by*

$$Bias_{NIE} = 0 \qquad RMSE_{NIE} = \frac{1}{\sqrt{M}} \cdot c_I$$

where $c_I = \int_0^1 \sqrt{I \cdot (1-I)} \cdot f(I) \cdot dI$, is a constant that depends on the impact distribution $f(I)$.

Proof. The proof is given in Appendix A. □

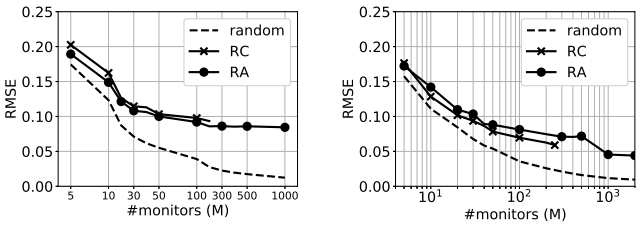
Remark: The impact distribution $f(I)$ depends on the {V,H} pairs that are expected to be involved in a hijack, and the hijack type. For example, if any pair of ASes is equally probable to be the {V,H} pair, then the impact I of a Type-0 hijack is approximately uniformly distributed in $[0, 1]$.

Table I gives the values of the constant c_I for random {V,H} pairs. We also consider scenarios that are closer to reported hijacking activity: namely, scenarios where (i) the {V,H} ASes correspond to the events identified as potential hijacks by the BGPmon service in 2018 [35], and (ii) hijackers are from the set of 22 ASes classified as "serial hijackers" [36] and victims are selected randomly.

Remark: The RMSE(I) is not equal for all values of the real impact I (see Appendix A). In fact, it is a concave function with a maximum $\frac{1}{2 \cdot \sqrt{M}}$ at $I = 0.5$, and minimum 0 at the corner cases of $I = 0$ or 1. In other words, it becomes more difficult to estimate with high accuracy when the victim and hijacker attract similar fractions of AS routes. This explains the lower values of c_I for higher-N hijack types: as N increases, the mass of the impact distribution concentrates around smaller values –closer to $I = 0$– in the area in which the RMSE(I) takes low values.

TABLE I
Experimentally calculated c_I (in parentheses, the average impact $E[I]$) for different hijack types and {V,H} pairs.

	Type-0	Type-1	Type-2
random {V,H} pairs	0.39 (0.50)	0.36 (0.30)	0.31 (0.19)
BGPmon {V,H} pairs [35]	0.35 (0.43)	0.29 (0.26)	0.22 (0.17)
random V, "serial" H [36]	0.37 (0.68)	0.40 (0.49)	0.36 (0.31)



(a) theory & simulations (b) PEERING experiments

Fig. 1. RMSE of the NIE (y-axis) vs. number of monitors (x-axis) for Type-0 hijacks and *random*, *RC*, and *RA* sets of monitors. (*Note the different x-axes*)

Figure 1(a) shows the RMSE of NIE with a random set of monitors (dashed line) for hijacks of Type-0, as calculated from Theorem 1 (the simulation results for random monitors almost coincide with the theoretical, i.e., the dashed line, and thus are omitted). The curves for other hijack types and/or {V,H} pairs have the same shape, since only the multiplicative factor c_I changes (see Table I).

2) NIE with Monitors from Public Measurement Services.

We now investigate the accuracy of NIE when using the public infrastructure monitors RC and RA (§II-C), which are *not* uniformly located in the Internet [21], [22]

Key finding: *The (non-uniform) locations of the public infrastructure monitors heavily affect the accuracy of NIE.*

Table II, where we compare simulation results for the RMSE of the NIE using public monitors vs. random sets of (equal number of) monitors, clearly demonstrates that the accuracy heavily depends on the set of employed monitors.

TABLE II
RMSE for NIE with public monitors (and random set of monitors) for different hijacks types; simulation results.

	Type-0	Type-1	Type-2
RC, 228 monitors	10% (2.6%)	9% (2.4%)	8% (2.1%)
RA, 3420 monitors	9% (0.7%)	8% (0.6%)	7% (0.5%)

A NIE using the monitors of the public services has a RMSE of 9%-10% for Type-0 hijacks, while the same estimator with randomly selected monitors would achieve almost 5 times (2.6%) and 10 times ($< 1\%$) lower RMSE (for the same number of monitors); similar results hold for all hijack types. It is interesting to observe that despite the fact that there are an order of magnitude more data-plane monitors (RA) than

control-plane monitors (RC), the accuracy is very similar: RA achieves only 1% lower RMSE than RC.[2]

Key finding: *Measuring 50 monitors of public services, is typically enough for achieving close to the(ir) highest accuracy.*

Figure 1(a) compares the RMSE of NIE with random, RC, and RA monitors for Type-0 hijacks as a function of the number of monitors (in the RC and RA cases, we select a random subset of size M in each simulation). We observe that the RMSE of NIE with RC or RA reaches the plateau of around 10% after 30-50 monitors; for the same M, the RMSE of NIE with random monitors is two times lower (around 5%) and further decreases with the number of monitors. Similar findings hold for the case of hijack Types-1 and 2 as well.

The experimental results (Fig. 1(b)) are in line with the simulations: (i) public monitors perform consistently worse than random monitors, (ii) the RC and RA curves are similar, and (iii) $M = 50$ monitors already achieve 7-8% RMSE. Note that we use the experiments only for a qualitative validation; the limited number of possible experiments, cannot provide strong statistical significance for the actual RMSE values (e.g., confidence intervals for $M = 100$ are $\pm 2.7\%$ and $\pm 3.4\%$ for RC and RA, respectively).

C. Designing Impact Estimation Methods

Below we discuss some practical aspects on the implementation of an impact estimator, which –in combination with the above findings– drive our design for the hijack impact estimation methodologies in §IV and §V.

Random monitors vs. Public infrastructure. Our results show that selecting monitors randomly (among all ASes in the Internet) results in significantly lower error. Thus, random monitors are preferable when accuracy is the main goal. However, this approach can be implemented only with ping measurements, since there are no public monitors in all ASes.

Ping campaigns: challenges and limitations. Measuring with pings whether a monitor is infected has some challenges in practice. Pinging an IP address does not necessarily mean that it will reply; in fact, a very small fraction of the addresses in the IP space respond to pings [37], [38]. While there are lists of "pingable" IP addresses per AS [39], they still not always respond to pings. If a monitor does not reply to a ping, we may falsely infer the monitor as infected, and thus overestimate the impact of the hijack. To overcome this challenge, in §IV we first study and quantify the effect of ping failures, and then design a methodology that can still be accurate, by carefully selecting the set and number of IPs per AS to ping.

Finally, a limitation of the ping measurements approach is that it is not applicable to MitM hijacks: all replies will end up to the victim, thus falsely denoting a monitor as non-infected. This can be only overcome with control-plane approaches.

Potential of public infrastructure estimations. Using control-plane information (e.g., BGP updates from RC monitors) applies to any hijack type [9]. Moreover, it can be real-time [22], [24], [25], and implemented by a third-party (i.e., not necessarily the victim network). In this context, and since applying the basic NIE with RC monitors leads to lower accuracy, in §V we design an estimator more sophisticated than NIE, which uses public infrastructure monitors and achieves comparable performance to the ping-based estimations.

[2]While studying the geographical distribution of RC/RA monitors is out of our scope, Appendix C gives some results on its effect on the NIE accuracy.

IV. Impact Estimation with Pings

We propose a hijack impact estimation methodology based on ping campaigns, which is summarized as follows:

Ping-IE: Hijack impact estimation with ping campaigns

1) Select randomly a set of M ASes.
2) For each AS, select a set of N_{IP} responsive ("pingable") IP addresses, ping them, and monitor for the replies.
3) If at least one IP address of an AS i replies to the ping, then set $\hat{m}_i = 0$, otherwise set $\hat{m}_i = 1$.
4) Estimate the hijack impact from the NIE expression in Eq. (1), by using $\hat{m}_i$ instead of m_i.

Despite the simplicity of the Ping-IE steps, the accuracy heavily depends on the parameters M and N_{IP}, and the set of "pingable" IP addresses. In the remainder, we study the expected accuracy and how to carefully tune these parameters.

A. The Effect of Failed Measurements on the NIE

Let assume that we conduct ping measurements to an IP address in the AS i to infer if it is infected (no ping reply received) or not (ping reply received). However, if AS i is not infected, but the selected IP address is configured to not reply to pings, or for some other reason unrelated to the hijack a ping reply never reaches our system, then we will incorrectly infer that the AS i is infected. If this happens with several ASes, the NIE will overestimate the hijack impact.

The following theorem quantifies the introduced bias (i.e., overestimation of hijack impact) and the RMSE of the NIE as a function of the measurement failure probability.

Definition 2 (Measurement failure probability)**.** *Let m be an indicator that denotes whether a monitor is infected ($m = 1$) or not ($m = 0$), and $\hat{m}$ be its measured value. The measurement failure probability, p, is the probability of measuring as infected a non-infected monitor, i.e.,*

$$p = Prob\{\hat{m} = 1 | m = 0\}$$

Theorem 2. *Under a randomly selected set of monitors $\mathcal{M}$, and a measurement failure probability p, it holds for NIE:*

$$Bias_{NIE} = c_I^{'} \cdot p$$

$$RMSE_{NIE} = \int_0^1 \sqrt{\frac{A_{I,p}}{M} + B_{I,p}} \cdot f(I) \cdot dI$$

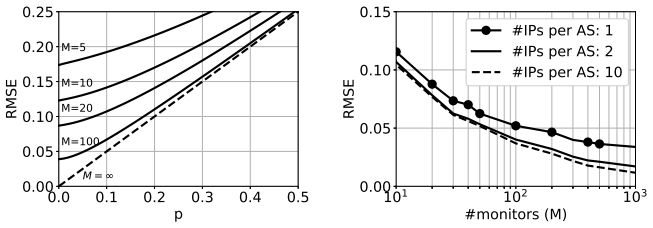
where $c_I^{'} = 1 - E[I]$ is a constant that depends on the impact distribution $f(I)$, and $A_{I,p}$ and $B_{I,p}$ are given by

$$A_{I,p} = (I + (1 - I) \cdot p) \cdot (1 - I) \cdot (1 - p)$$

$$B_{I,p} = (1 - I)^2 \cdot p^2$$

Proof. The proof is given in Appendix B. □

Corollary 1. $RMSE_{NIE} \geq RMSE_{NIE}(M \to \infty) = c_I^{'} \cdot p$.



(a) (b)

Fig. 2. RMSE (y-axis) of the NIE / Ping-IE under random set of M monitors: (a) theoretical results for the NIE ($N_{IP} = 1$) under different ping failure probabilities p (x-axis); (b) results from the PEERING experiments for the Ping-IE under different number of monitors (x-axis) and N_{IP} (legend).

The values $c_I^{'} = 1 - E[I]$ for the different hijack types and {V,H} pairs can be calculated from Table I. For example, for random {V,H} pairs and hijack Types-0, 1, and 2, the constant $c_I^{'}$ is 0.5, 0.7, and 0.81, respectively. *Remark:* It is interesting to note that the RMSE of NIE increases with the hijack type when $p > 0$, whereas the opposite holds when $p = 0$ (see §III). This is due to the fact that p affects only the monitors that are *not* infected, and thus in cases where the impact is lower, i.e., for higher hijack types, the error due to ping failures is higher.

Key finding: *When the failure probability is larger than 20% ($p \geq 0.2$), ping campaigns –no matter how many ASes are pinged– are less accurate than NIE with RC or RA monitors.*

As expected, NIE under ping failures becomes a biased estimator. Also its RMSE increases with the failure probability p, which means that for high p the NIE with ping campaigns becomes worse than the NIE with public services. To better quantify the expected estimation error, we present in Fig. 2(a) the RMSE for hijacks of Type-0 (for Types ≥ 1 the RMSE is higher) as a function of the failure probability p (x-axis) and the number of monitors M (i.e., pinged ASes).

Based on the analytical findings, we proceed to design and fine tune a methodology by considering practical issues.

B. Practical Design of the Ping-IE

The above results indicate that for a low RMSE, we need to have low ping failure probabilities p. One can achieve this by (i) carefully selecting the set of IPs to be pinged so that they are "pingable", and/or (ii) pinging more than one IPs per AS and waiting for a reply from at least one IP. In the methodology we propose, we do both. Specifically, we first quantify the ping failure probability we expect to have in practice, and based on this, we select the set of IP addresses (which IP addresses and how many per AS) to be pinged.

Selection of pingable IP addresses. If we select arbitrarily an IP address within the prefixes of an AS, then the failure probability is very large (more than 90% [37], [38]); this would lead to a very inefficient methodology with $RMSE > 45\%$. Therefore, we use a list of IP addresses provided by ANT Lab [39] that have high probability of replying to pings. Specifically, we compile a list of pingable IP addresses per AS, by combining the following two datasets:

- *IP hitlists from ANT Lab* [39]: These are lists of IP addresses that are found to be reachable via `ping` with high probability, based on past measurements. We select only the IP addresses with more than 90% confidence score.
- *Prefix-to-AS mapping:* We consider information from RIPE RIS [21] (via the RIPEstat API [40]) and RouteViews [22] (via CAIDA's `pfx2as` [41]). These datasets map IP prefixes announced in BGP to the originating ASNs. We filter out mappings that are inconsistent within a period of a month (e.g., due to transient incidents), and merge the two datasets.

Selection of the number of IP addresses to ping per AS (N_{IP}). Pinging more than one IP addresses per AS, increases the probability to obtain a correct inference about whether it is infected (i.e., we need at least one ping reply to infer an AS as non-infected). But, *how many IP addresses need to be pinged per AS to have a low error*? To quantify this, we conduct a set of measurements using the PEERING testbed: we announce a prefix from PEERING, ping from a host within PEERING the top 10 pingable IP addresses per AS for $\sim 46k$ ASes, and monitor which IP addresses reply to the pings.

Key finding: *To achieve low estimation error we need to ping at least 2 IP addresses from the ANT Lab's IP hitlists [39] per AS. Pinging more than 3 IP addresses, does not significantly improve accuracy.*

Table III (top row) presents the fraction of ASes for which we did not receive any reply from pinging their top-x ($x = 1, ..., 10$) IP addresses. We can see that the failure probability is quite high (12.8%) when pinging only one IP address per AS, which indicates that more measurements per AS are needed to enable an accurate impact estimation. With 3 measurements the ping failure probability decreases to 2.1% and further decreases gradually to 0% with 10 pings per AS. The middle rows of Table III show the *lower bound* for the RMSE ($M \to \infty$) that can be achieved by Ping-IE for different hijack types, which indicates that only one ping per AS may not be enough to outperform NIE with public monitors.

Finally, the bottom rows give the RMSE for Type-0 hijacks and practical values of M, calculated from the expressions of Theorem 2 (using the values p that correspond to the N_{IP} from the first row of the table). While pinging only 10 ASes is not efficient, pinging 100 ASes already achieves an accuracy relatively close to the best achievable ($M \to \infty$).

TABLE III
TOP ROW: PROBABILITY OF PING FAILURE FOR ANT LAB'S IP HITLISTS [39] (PEERING EXPERIMENTS). MIDDLE/BOTTOM ROWS: RMSE OF THE PING-IE VS. NB. OF PINGED TOP IPS PER AS (THEORY).

		# of pinged IP addresses per AS				
		1	2	3	...	10
% ASes with no reply		12.8%	4.2%	2.1%	...	0%
RMSE M=∞	Type-0	6.4%	2.1%	1.0%	...	0%
	Type-1	9.0%	3.0%	1.4%	...	0%
	Type-2	10.4%	3.4%	1.7%	...	0%
RMSE Type-0	M=10	14.9%	12.9%	12.6%	...	12.3%
	M=50	9.0%	6.2%	5.7%	...	5.5%
	M=100	7.9%	4.7%	4.1%	...	3.9%

In Fig. 2(b) we present the corresponding results from the real experiments, which are in agreement with the main theoretical findings[3]. In particular, we observe that pinging 2 IP addresses per AS ($N_{IP} = 2$), significantly reduces the RMSE compared to the case of $N_{IP} = 1$. However, the improvement by further increasing the N_{IP} (up to $N_{IP} = 10$) is marginal. Moreover, in all N_{IP} cases, increasing the number of pinged ASes more than $M = 500$ barely improves accuracy (as was already indicated by Fig. 2(a)).

V. Improving Impact Estimation with Public Infrastructure Monitors

The biased view of public monitors. As already discussed, public monitors are not uniformly deployed in the Internet, and this increases the error of NIE. Consider the following example scenario: The victim is an AS that is located in a geographical area where many monitors exist (e.g., with direct peering or short paths to these monitors), and the hijacker AS is in a different area with less monitors. The actual impact of the hijacks is $I = 50\%$ (i.e., half of all ASes are infected), however, the NIE underestimates the impact (i.e., $\hat{I}_{NIE} < I$) because more monitors would prefer the paths to the victim.

Generalizing the above example, the error of NIE increases when the monitors are not representative of the global connectivity, or –more abstractly– when there are correlations between their measurements (due to locations, underlying topology, AS-relationships, etc.). Hence, *to improve the estimation accuracy of NIE under public monitors, one needs to take into account the correlations between the monitors*. To this end, in the following we design a statistical learning methodology that exploits information of past events (to identify correlations between public monitors), fits a model that diminishes the effect of correlations, and returns an estimation for the impact.

The linear regression estimator (LRE). The methodology we propose is summarized as follows:

LRE: Linear Regression Estimator

1) Compile a dataset from N past events (hijacks, anycast announcements, etc.), where for each event j, $j = 1, ..., N$, collect the measurements $m_i^{(j)}$ of the monitors $i \in \mathcal{M}$, and the actual hijack impact $I^{(j)}$.
2) Fit a least squares estimator, by calculating the weights w_i, $i \in \mathcal{M}$, as:

$$\mathbf{w} \leftarrow \arg\min_{\mathbf{w}} \ \left(||\mathbf{M} \cdot \mathbf{w} - \mathbf{I}||_2\right)^2 + \alpha \cdot \left(||\mathbf{w}||_2\right)^2$$

where $\mathbf{w} = [w_1, ..., w_M]$ and $\mathbf{I} = [I^{(1)}, ..., I^{(N)}]$ are vectors, $\mathbf{M}$ is the matrix with elements $m_i^{(j)}$ at the i^{th} row and j^{th} column, and $||\cdot||_2$ denotes the l2-norm.
3) Estimate the hijack impact from the current monitor measurements m_i and the calculated weights w_i as

$$\hat{I} = \textstyle\sum_{i \in \mathcal{M}} m_i \cdot w_i$$

The first step is to collect data that contain information about the correlations between the measurements of the different public monitors. To do this, one can consider a set of past/ongoing events $\mathcal{N}$ ($|\mathcal{N}| = N$), where two (at least) ASes announce the same prefix. Such events can be actual or emulated (e.g., as in our experiments; §II-E) hijacking events, or legitimate anycasting announcements (which from a routing point of view are equivalent to Type-0 hijacks) [28]. For each of these events $j \in \mathcal{N}$, we collect the measurements $m_i^{(j)}$ of the public monitors $i \in \mathcal{M}$. In the case of RC monitors the measurements can be retrieved from the the RIPE RIS [21] and the RouteViews [22] services directly, or from the open-source tool BGPStream [42], [43] that aggregates these measurements, and in the case of RA from the RIPE Atlas API (by triggering measurements for ongoing events, or collecting the periodic measurements for past events) [23]. Moreover, for each event we need to know the actual hijack impact, which can be exactly measured with exhaustive ping measurements (similarly to our methodology in §II-E for collecting the ground-truth in our experiments, or other related approaches [28]) or approximated well with a few thousands of ping measurements using the methodology of §IV.

The second step is to identify any correlations between the measurements of the monitors, and eliminate their effect in the estimation. We select to do this, by using a least squares approach, and, in particular, a linear regression estimator with regularization of the weights (i.e., Ridge regression). Our choice, is motivated by the fact that (i) the least square estimator (i.e., linear regression) has the the lowest variance within the class of linear unbiased estimators [44][4], and (ii) the regularization significantly reduces the variance of the estimations when multi-collinearity occurs; in fact, the public monitor measurements are highly collinear, and thus large values of the regularization parameter α are needed (e.g, we found that values $a \geq 50$ performed best)

Finally, having fitted the model (i.e., the weights w_i) that eliminates the correlations between monitor measurements, we can apply it to any new hijacking event and estimate its impact.

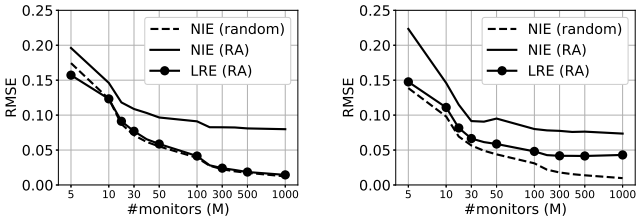
LRE vs. NIE estimation accuracy. We compare the accuracy of the impact estimations by LRE and NIE with RA monitors in Fig. 3 (similar results hold for the RC monitors). We use 1000 simulations as the past-events dataset, from which we collect the data to fit the LRE, and apply the LRE and NIE to a different set of 1000 simulations.

Key finding: *LRE can eliminate the effect of correlations in public monitor measurements and achieve similar efficiency to the (best performing) ping-based estimators.*

We see that LRE has significantly lower RMSE than NIE. In fact, in the case of Type-0 hijacks (Fig. 3(a)) LRE achieves equal accuracy to the NIE with random monitors (Theorem 1), or even better accuracy for small number of monitors. This is an important finding that demonstrates that we can design estimators based on public monitors with similar efficiency to the ping-based estimators. LRE outperforms NIE for the case

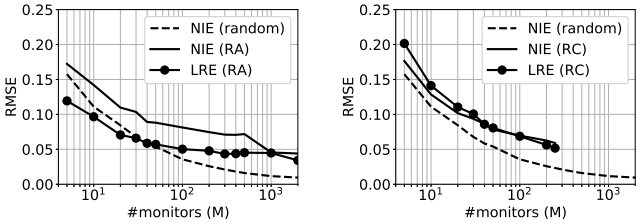
[3]For small N_{IP} or M, the RMSE values in our experiments are a bit lower than the corresponding theoretical values (Table III); this is due to the small number of experiments (confidence intervals are larger for small N_{IP} or M).

[4]We tested several non-linear estimators as well (e.g., support-vectors, random forests, neural networks). However, they had similar (or worse) performance to LRE. We selected the LRE, as a simple model, which comes with the advantages of interpretability, need for less training data, etc.



(a) Type-0 hijack (b) Type-2 hijack

Fig. 3. RMSE (y-axis) of the LRE with RA monitors and the NIE with random or RA monitors, vs. number of monitors (x-axis), in simulations of (a) Type-0 and (b) Type-2 hijacks.



(a) RA monitors (b) RC monitors

Fig. 4. RMSE (y-axis) of the LRE with public monitors and the NIE with random or public monitors, vs. number of monitors (x-axis) for Type-0 hijacks for (a) *RA* and (b) *RC* monitors in the PEERING experiments.

of Type-2 hijacks as well (Fig. 3(b)), e.g., having almost 50% less RMSE for $M \geq 50$ monitors. Comparing the RMSE of LRE in the cases of Type-0 and Type-2 hijacks, we can see that it increases with the hijack type; this is due to the fact that the actual impact of higher type hijacks is lower, and thus there are more observations $m_i = 0$, which makes more difficult for a model to identify the existing correlations in measurements.

We proceed to test the efficiency of LRE in the real experiments with PEERING. We remind that we have only 22 experiments, which is a very small dataset for training a model. Hence, this is not a conclusive evaluation (it can be rather seen as a stress-test of LRE); nevertheless, our findings are very promising. For each experiment $\{$V,H$\}$, we consider as the dataset with past events the other 20 experiments (omitting also the experiment $\{$H,V$\}$) and fit the LRE. Figure 4 presents the results for RC and RA monitors. In the case of RA monitors (Fig. 4(a)), we can see that LRE clearly outperforms NIE with RA monitors, despite the very limited available data for training the LRE. In the case of RC (Fig. 4(b)), LRE has a similar performance to NIE with RC monitors. These findings verify the efficiency of LRE, and indicate that even information from only a few past events can lead to more accurate estimations. However, they also highlight the importance of collecting past event data for training estimators; our experiments data [15] can contribute to this direction.

VI. Related Work

The majority of works on BGP prefix hijacking (or other types of events affecting the Internet operation, e.g., outages [45], [46]) focus on the detection of an event, using network measurements on the control plane [9] or the data plane [10] or both [11], [12]. The difference between *detection* and *impact estimation* lies in the fact that having information for at least one infected AS is typically enough to enable the detection of a hijack, however, it gives only limited information (if any) about its overall impact. Our work focuses on quantifying the impact of a (detected) hijack, and thus complements the existing detection methods.

A few works studying (through simulations) the hijack impact focus mainly on the *average* impact of different hijacking attacks [47] or hijacker ASes [13], [48], whereas our goal is to estimate (through measurements) the *actual impact of an ongoing hijack*. Ballani et al. [47] consider interception hijacks , and study when they are expected to have significant impact, by providing coarse estimates for groups of ASes (e.g., Tier-1) that could act as hijackers. Similarly, the potential impact of a hijacker AS (or, conversely, the resilience of a victim AS to a hijack) is studied in [48], where the topological characteristics (e.g., node degree) of ASes are used to classify potential hijackers based on the impact they can cause. TowerDefense [13] aims to find a set of monitors that maximizes the probability to detect a hijack (i.e., at least one monitor is infected); a problem that is complementary to impact estimation, whose aim is to find a *representative set of monitors* (i.e., a set where the fraction of infected monitors is close to the overall hijack impact).

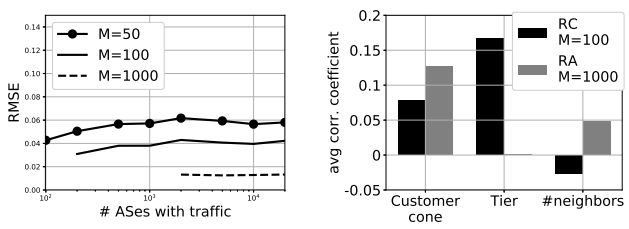
Finally, the framework of [29] for predicting the catchment of an anycast deployment, could be used for hijack impact estimation (a hijack can be seen as a setup where the pair $\{$V,H$\}$ anycasts the same prefix). However, the routing information that is required may not be always accurately known in practice, which would lead to higher errors than a measurement-based NIE (we verified this in our experiments).

VII. Conclusion

The problem of hijack impact estimation has not been given attention in literature, despite its usefulness for network operations and economy, e.g., to know how an ongoing hijack affects a network, or to select and evaluate the efficiency/cost of different mitigation measures. In this paper we made the first steps towards understanding the fundamental (limits, trade-offs, etc.) and practical aspects (use of public infrastructure, measurement failures, etc.) of the impact estimation problem. We also designed accurate estimation techniques that are easy to implement and incorporate in existing defense systems.

We believe that this work can motivate further research on the topic; we identify and discuss two interesting directions:

A network may exchange traffic with only a subset of ASes in the Internet (and/or different volumes of traffic per AS). In this case, a more fine-tuned estimation of the impact (on the exchanged traffic) can further help network operators. Sophisticated estimators, such as weighted versions of the NIE, Ping-IE, or LRE, can be designed. A preliminary view in this direction is given in Fig. 5(a), where we apply NIE to estimate the impact only on subsets of ASes: the RMSE of NIE with random set of monitors remains almost constant



(a) Traffic subset (b) LRE weights

Fig. 5. (a) RMSE (y-axis) of NIE with random set of M monitors measuring the hijack impact on different number of ASes with traffic to the victim AS (x-axis). (b) Correlation between LRE weights w_i and AS-topology characteristics of monitors i; average values over 30 simulation scenarios.

(we observed similar behavior for the NIE with RC and RA monitors as well). While a formal and detailed study is needed to draw firm conclusions, this is an indication that the main findings of this paper can hold for more generic cases as well.

The LRE findings can motivate research on the selection and combination of measurements from public infrastructure, with broader applications on the Internet monitoring. To this end, Fig. 5(b) provides some initial statistics on the correlation between the LRE weights w_i (i.e., the importance of monitor i) and the topological characteristics of monitors i. Admittedly correlations are weak, however, they reveal some underlying trends and give rise to some interesting questions: (i) Observations from RC and RA monitors/ASes with larger customer cones seem to play a more important role; does this indicate that we should deploy more monitors on such networks? (ii) Top tier networks (e.g., Tier-1) contribute more on the LRE, but only in the RC case; should we devise different strategies for selecting measurements from RC and RA? (iii) Finally, the number of neighbors a monitor has, seems to be a less important feature; would this mean that monitors at IXPs (where networks establish a lot of peerings) are equally important with monitors at stub networks? Further research and (open) data could help answering such questions.

REFERENCES

[1] BGPMon, "BGP leak causing Internet outages in Japan and beyond," bgpmon.net/bgp-leak-causing-internet-outages-in-japan-and-beyond/.

[2] NANOG mailing list archives, "Another day, another illicit SQUAT," seclists.org/nanog/2016/Oct/578, Oct. 2016.

[3] www.wired.com/2014/08/isp-bitcoin-theft/.

[4] www.bgpmon.net/chinese-isp-hijacked-10-of-the-internet/.

[5] P. Sermpezis, V. Kotronis, A. Dainotti, and X. Dimitropoulos, "A survey among network operators on bgp prefix hijacking," *ACM SIGCOMM Computer Communication Review*, vol. 48, no. 1, pp. 64–69, 2018.

[6] M. Lepinski, R. Barnes, and S. Kent, "An infrastructure to support secure internet routing," RFC6480, 2012.

[7] M. Lepinski, "BGPSEC protocol specification," RFC8205, 2015.

[8] A. Cohen, Y. Gilad, A. Herzberg, and M. Schapira, "Jumpstarting bgp security with path-end validation," in *Proc. ACM SIGCOMM*, 2016.

[9] P. Sermpezis, V. Kotronis, P. Gigis, X. Dimitropoulos, D. Cicalese, A. King, and A. Dainotti, "Artemis: Neutralizing bgp hijacking within a minute," *IEEE/ACM Trans. on Networking*, vol. 26, no. 6, 2018.

[10] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush, "iSPY: detecting ip prefix hijacking on my own," *ACM SIGCOMM CCR*, 2008.

[11] J. Schlamp, R. Holz, Q. Jacquemart, G. Carle, and E. Biersack, "HEAP: Reliable Assessment of BGP Hijacking Attacks," *IEEE JSAC*, vol. 34, no. 06, pp. 1849–1861, 2016.

[12] X. Shi, Y. Xiang, Z. Wang, X. Yin, and J. Wu, "Detecting prefix hijackings in the Internet with Argus," in *Proc. ACM IMC*, 2012.

[13] T. Qiu, L. Ji, D. Pei, J. Wang, and J. Xu, "Towerdefense: Deployment strategies for battling against ip prefix hijacking," in *IEEE ICNP*, 2010.

[14] M. Nawrocki, J. Blendin, C. Dietzel, T. C. Schmidt, and M. Wählisch, "Down the black hole: Dismantling operational practices of bgp blackholing at ixps," in *Proc. ACM IMC*, 2019, pp. 435–448.

[15] "Supplementary material: Estimating the impact of bgp prefix hijacking," https://github.com/sermpezis/bgp-estimation, 2021.

[16] L. Miller and C. Pelsser, "A taxonomy of attacks using bgp blackholing," in *European Symposium on Research in Computer Security*, 2019.

[17] NIST, "RPKI Monitor," rpki-monitor.antd.nist.gov/, 2019.

[18] T. Chung *et al.*, "Rpki is coming of age: A longitudinal study of rpki deployment and invalid route origins," in *Proc. ACM IMC*, 2019.

[19] A. Reuter *et al.*, "Towards a rigorous methodology for measuring adoption of rpki route validation and filtering," *ACM SIGCOMM CCR*, vol. 48, no. 1, 2018.

[20] Cisco, "BGP Best Path Selection Algorithm," https://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/13753-25.html.

[21] RIPE NCC, "Routing Information Service (RIS)," https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris, 2018.

[22] University of Oregon, "Route Views Project," www.routeviews.org.

[23] RIPE NCC, "RIPE Atlas," https://atlas.ripe.net/, 2018.

[24] "RIPE RIS - Streaming Service," labs.ripe.net/Members/colin_petrie/updates-to-the-ripe-ncc-routing-information-service.

[25] C. Orsini, A. King, D. Giordano, V. Giotsas, and A. Dainotti, "Bgpstream: a software framework for live and historical bgp data analysis," in *Proc. ACM IMC*, 2016, https://bgpstream.caida.org/.

[26] Z. M. Mao, J. Rexford, J. Wang, and R. H. Katz, "Towards an accurate as-level traceroute tool," in *Proc. ACM SIGCOMM*, 2003, p. 365–378.

[27] A. Marder, M. Luckie, A. Dhamdhere, B. Huffaker, k. claffy, and J. M. Smith, "Pushing the boundaries with bdrmapit: Mapping router ownership at internet scale," in *Proc. ACM IMC*, 2018, p. 56–69.

[28] W. B. De Vries, R. de O Schmidt, W. Hardaker *et al.*, "Broad and load-aware anycast mapping with verfploeter," in *Proc. ACM IMC*, 2017.

[29] P. Sermpezis and V. Kotronis, "Inferring catchment in internet routing," *Proc. ACM Meas. Anal. Comput. Syst. (Sigmetrics)*, vol. 3, no. 2, 2019.

[30] CAIDA, "AS relationships," data.caida.org/datasets/as-relationships/.

[31] W. Mühlbauer, A. Feldmann, O. Maennel, M. Roughan, and S. Uhlig, "Building an as-topology model that captures route diversity," *ACM SIGCOMM CCR*, vol. 36, no. 4, pp. 195–206, 2006.

[32] L. Gao and J. Rexford, "Stable internet routing without global coordination," *IEEE/ACM TON*, vol. 9, no. 6, pp. 681–692, 2001.

[33] R. Anwar, H. Niaz, D. Choffnes *et al.*, "Investigating interdomain routing policies in the wild," in *Proc. ACM IMC*, 2015.

[34] B. Schlinker, T. Arnold, Í. Cunha, and E. Katz-Bassett, "Peering: Virtualizing bgp at the edge for research," in *Proc. ACM CoNEXT*, 2019, https://peering.usc.edu/.

[35] K. Arakadakis and *et al.*, "Analysis of BGP prefix hijacking events: a commercial service's view," ACM CoNEXT (poster), 2018.

[36] C. Testart, P. Richter, A. King, A. Dainotti, and D. Clark, "Profiling bgp serial hijackers: Capturing persistent misbehavior in the global routing table," in *Proc. ACM IMC*, 2019.

[37] D. Soler, "https://www.securityartwork.es/2013/02/07/the-result-of-pinging-all-the-internet-ip-addresses/," 2013.

[38] J. Heidemann and et al., "Census and survey of the visible internet," in *Proc. ACM IMC*, 2008, pp. 169–182.

[39] ANT Lab, "Ip address space hitlists," https://ant.isi.edu/datasets/ip_hitlists/format.html.

[40] RIPE NCC, "RIPEstat," stat.ripe.net/.

[41] CAIDA, "Routeviews Prefix-to-AS mappings (pfx2as) for IPv4 and IPv6," http://data.caida.org/datasets/routing/routeviews-prefix2as/, 2019.

[42] CAIDA, "BGPStream," bgpstream.caida.org/.

[43] C. Orsini, A. King, D. Giordano, V. Giotsas, and A. Dainotti, "Bgpstream: A software framework for live and historical bgp data analysis," in *Proc. of ACM IMC*, 2016, pp. 429–444.

[44] J. Davidson, *Econometric theory*. Wiley-Blackwell, 2000.

[45] L. Quan, J. Heidemann, and Y. Pradkin, "Trinocular: Understanding internet reliability through adaptive probing," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 255–266, 2013.

[46] R. Padmanabhan, A. Schulman, A. Dainotti, D. Levin, and N. Spring, "How to find correlated internet failures," in *Proc. PAM*, 2019.

[47] H. Ballani, P. Francis, and X. Zhang, "A study of prefix hijacking and interception in the internet," *ACM SIGCOMM CCR*, vol. 37, no. 4, 2007.

[48] M. Lad, R. Oliveira, B. Zhang, and L. Zhang, "Understanding resiliency of internet topology against prefix hijack attacks," in *IEEE DSN*, 2007.

APPENDIX A
PROOF OF THEOREM 1

The hijack impact I is the fraction of ASes that are infected. When choosing randomly the ASes/monitors to be measured (and the number of measurements is much less than the total number of ASes), we can reasonably model each measurement as an independent Bernoulli trial with probability I, i.e., the probability that a selected monitor is infected is I. The NIE involves the sum of M independent Bernoulli trials, and thus the sum follows a binomial distribution, i.e., $\sum_{i\in\mathcal{M}} m_i \sim Binomial(M, I)$, for which it holds that

$$E\left[\sum_{i\in\mathcal{M}} m_i\right] = M \cdot I \quad (2)$$

$$Var\left[\sum_{i\in\mathcal{M}} m_i\right] = M \cdot I \cdot (1-I) \quad (3)$$

Bias. The bias of an estimator is defined as

$$Bias = E[\hat{I} - I] = E[\hat{I}] - I \quad (4)$$

For the NIE it holds that

$$E[\hat{I}] = E\left[\tfrac{1}{M} \cdot \textstyle\sum_{i\in\mathcal{M}} m_i\right] = \tfrac{1}{M} \cdot E\left[\textstyle\sum_{i\in\mathcal{M}} m_i\right] = I \quad (5)$$

where in the last equation we used the expression of Eq. (2). Substituting Eq. (5) in Eq. (4) gives $Bias = 0$.

RMSE. The RMSE of NIE, given the actual impact I, is:

$$\begin{aligned} RMSE(I) &= \sqrt{E[(\hat{I} - I)^2]} = \sqrt{E[(\hat{I} - E[\hat{I}])^2]} \\ &= \sqrt{Var[\hat{I}]} = \sqrt{Var[\tfrac{1}{M}\textstyle\sum_{i\in\mathcal{M}} m_i]} \\ &= \sqrt{\tfrac{1}{M^2} \cdot Var[\textstyle\sum_{i\in\mathcal{M}} m_i]} = \tfrac{\sqrt{I\cdot(1-I)}}{\sqrt{M}} \end{aligned} \quad (6)$$

where we use the definition of the variance ($Var(x) = E[(x - E[x])^2]$), and in the last equality the expression from Eq. (3).

Then, the RMSE of NIE follows by taking the expectation of $RMSE(I)$ in Eq. (6) over the impact distribution $f(I)$:

$$RMSE = \textstyle\int_0^1 \tfrac{\sqrt{I\cdot(1-I)}}{\sqrt{M}} \cdot f(I) \cdot dI = \tfrac{1}{\sqrt{M}} \cdot c_I \quad (7)$$

APPENDIX B
PROOF OF THEOREM 2

If $m_i = 1$, then $\hat{m}_i$ is 1 as well. However, if $m_i = 0$, then $\hat{m}_i$ is 1 with probability p (Definition 2) and 0 with probability $1 - p$. Taking into account the fact that the indicator m_i follows a Bernoulli trial with probability I (see Appendix A), it follows that $\hat{m}_i$ follows also a Bernoulli trial with probability

$$P\{\hat{m}_i{=}1\} = P\{m_i{=}1\} + p \cdot P\{m_i{=}0\} = I + (1-I)p \quad (8)$$

and thus it holds

$$E[\hat{m}_i] = P\{\hat{m}_i = 1\} = I + (1-I) \cdot p \quad (9)$$

In the case of measurement failures, the expression of NIE is calculated from the indicators $\hat{m}_i$. Thus, the expectation is

$$\begin{aligned} E[\hat{I}] &= E[\tfrac{1}{M}\textstyle\sum_{i\in\mathcal{M}} \hat{m}_i] = \tfrac{1}{M} \cdot E[\textstyle\sum_{i\in\mathcal{M}} \hat{m}_i] \\ &= \tfrac{1}{M} \cdot M \cdot (I + (1-I)\cdot p) = I + (1-I)\cdot p \end{aligned} \quad (10)$$

where in the third equality we used the expression of Eq. (9).

Bias. The bias follows by substituting Eq. (10) in Eq. (4):

$$Bias_{NIE} = E[\hat{I} - I] = (1-I)\cdot p \quad (11)$$

RMSE. The RMSE of NIE, given the actual impact I, is:

$$RMSE(I) = \sqrt{E[(\hat{I} - I)^2]} = \sqrt{E[\hat{I}^2] + I^2 - 2\cdot I \cdot E[\hat{I}]}$$

and using the property $Var(x) = E[x^2] - (E[x])^2$, gives:

$$\begin{aligned} RMSE(I) &= \sqrt{Var[\hat{I}] + (E[\hat{I}])^2 + I^2 - 2\cdot I\cdot E[\hat{I}]} \\ &= \sqrt{Var[\hat{I}] + (E[\hat{I}] - I)^2} \\ &= \sqrt{Var[\tfrac{1}{M}\cdot\textstyle\sum_{i\in\mathcal{M}}\hat{m}_i] + (I + (1-I)\cdot p - I)^2} \\ &= \sqrt{\tfrac{1}{M^2}\cdot Var[\textstyle\sum_{i\in\mathcal{M}}\hat{m}_i] + ((1-I)\cdot p)^2} \end{aligned} \quad (12)$$

The quantity $\sum_{i\in\mathcal{M}} \hat{m}_i$ is the sum of M independent Bernoulli trials with probability given by Eq. (8). Therefore

$$\begin{aligned} Var[\textstyle\sum_{i\in\mathcal{M}}\hat{m}_i] &= M \cdot P\{\hat{m}_i = 1\}\cdot(1 - P\{\hat{m}_i = 1\}) \\ &= M\cdot(I + (1-I)\cdot p)\cdot(1 - I + (1-I)\cdot p) \end{aligned} \quad (13)$$

Substituting Eq. (13) in Eq. (12), and taking the expectation over the distribution $f(I)$, gives the expression of the theorem.

APPENDIX C
THE EFFECT OF THE PUBLIC MONITORS' LOCATION ON THE NIE ACCURACY

The simulation results in Fig. 6 validate that the lower accuracy of public monitors is due to their non uniform locations. We grouped hijacks based on the locations of the {V,H} ASes (i.e., continents where their headquarters are located[5]). We observe that when {V,H} reside in different continents (left subplot), the difference in the visibility from public monitors leads to higher RMSE (while the difference in random monitors is much smaller). The right subplot presents some indicative cases, where we see that when V or H are located in Asia (where public monitors are scarce) the RMSE is significantly higher than in the case where {V,H} are in N./S. America; on the contrary, random monitors yield similar accuracy in all cases.

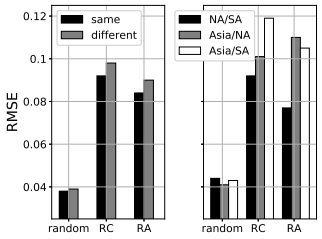


Fig. 6. RMSE of NIE (y-axis) using $M = 100$ monitors from different sets (x-axis); comparing results where {V,H} are located in the same/different continents (left subplot) and in N.America/S.America/Asia (right subplot).

[5]Retrieved from CAIDA's AS-Rank dataset as-rank.caida.org