SAARLAND UNIVERSITY

Faculty of Mathematics and Computer Science
Department of Computer Science
Dissertation

Routegazing: Analysing the Evolving Internet Routing Ecosystem

Dissertation zur Erlangung des Grades des
Doktors der Ingenieurwissenschaften (Dr.-Ing.)

der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

vorgelegt von
Lars Prehn

Saarbrücken, 2023

Date of the colloquium:	July 6th., 2023
Dean:	Prof. Dr. Jürgen Steimle
Chairman of the examination board:	Prof. Dr. Ingmar Weber
Reporter:	Prof. Dr. Anja Feldmann Prof. Dr. Laurent Vanbever Dr. Oliver Gasser
Scientific Assistant:	Dr. Jialong Li

Notes on sytle:
As most of the work presented in this dissertation was done in collaboration with other researchers, the scientific plural "we" is used.

Saarland University
Faculty MI – Mathematics and Computer Science
Department of Computer Science
Campus - Building E1.1
66123 Saarbrücken
Germany

Abstract

The Internet's routing ecosystem constantly evolves to meet the needs of its stakeholders and users. Tracking this evolution is essential, e.g., to identify business opportunities, address security challenges, or inform protocol design. However, most Internet protocols were designed without measurability in mind; hence, many measurements and inference methods rely on exploiting protocol-specific side effects.

This dissertation first assesses the limitations of our deployed observation infrastructures and commonly used inference methods via three orthogonal contributions: a case study on a European Internet Exchange Point to assess our visibility into the Internet's AS topology; a framework to identify and measure biases in the placement of our vantage points across multiple dimensions; and a systematic analysis of the biases and sensitivity of AS relationship inference algorithms. We found that our view of the Internet's AS topology diminishes over time, and that our AS relationship models are more biased and sensitive to short-term routing dynamics than previously assumed.

With these limitations in mind, we focused on one of the most critical routing ecosystem changes, IPv4 exhaustion, and two ways network operators can deal with it. First, we explored the IPv4 buying and leasing markets, identified market trends, and discussed the viability of these markets for different network types. Second, we analyzed the benefits, usage patterns, and disadvantages of announcing tiny address blocks—which we call "hyper-specific." We argue that a combination of leased IPv4 addresses and hyper-specific prefix announcements likely suffice for many networks to bridge the gap until full IPv6 adoption.

Besides its IPv6 adoption, the routing ecosystem also evolved in other dimensions. We first studied AS path prepending to assess the security implication of these changes. We found a typical configuration with no benefits yet an increase of an AS's vulnerability to prefix hijacks. Infrastructural changes led to an overall decrease in prepending sizes over time and hence a safer use of the technique. However, we demonstrated that we can exploit the same changes to re-orchestrate prefix de-aggregation attacks to overcome widely deployed prevention mechanisms. We validated our assumptions and attack model using a real-world testbed and proposed updates to existing prevention mechanisms. Our two-stage disclosure campaign contributed to a safer routing ecosystem.

Zusammenfassung

Das Routing-Ökosystem des Internets entwickelt sich ständig weiter, um den Bedürfnissen der Beteiligten und Nutzer gerecht zu werden. Die Verfolgung dieser Entwicklung ist wichtig, um z.B. Geschäftsmöglichkeiten zu erkennen, Sicherheitsprobleme zu antizipieren oder neue Protokolle zu entwickeln. Die meisten Internetprotokolle wurden jedoch ohne Rücksicht auf ihre Messbarkeit entworfen; daher beruhen viele Messungen und Schlussfolgerungsmethoden auf der Ausnutzung protokollspezifischer Nebeneffekte.

In dieser Dissertation werden zunächst die Grenzen der von uns eingesetzten Beobachtungsinfrastrukturen und der gängigen Inferenzmethoden anhand von drei orthogonalen Beiträgen bewertet: eine Fallstudie an einem europäischen Internet-Austauschpunkt zur Bewertung der Vollständigkeit unserer Sicht auf die AS-Topologie des Internets; ein Rahmenwerk zur Identifizierung und Messung von Verzerrungen bei der Platzierung unserer Beobachtungspunkte über mehrere Dimensionen hinweg; und eine systematische Analyse der Verzerrungen und der Empfindlichkeit von Algorithmen zur Inferenz von AS-Beziehungen. Unser Blick auf die AS-Topologie des Internets nimmt mit der Zeit ab, und unsere AS-Beziehungsmodelle sind voreingenommener und empfindlicher gegenüber kurzfristigen Routing-Dynamiken als bisher angenommen.

Mit diesen Einschränkungen im Hinterkopf haben wir uns auf eine der kritischsten Veränderungen im Routing-Ökosystem, die Erschöpfung von IPv4, und zwei Möglichkeiten, wie Netzbetreiber damit umgehen können, konzentriert. Zunächst untersuchten wir die Kauf- und Leasingmärkte für IPv4 Addresses, ermittelten Markttrends und diskutierten die Nutzbarkeit dieser Optionen für verschiedene Netzwerktypen. Danach haben wir die Vorteile, Nutzungsmuster und Nachteile der Nutzung von Routen für winzige Adressblöcke, die wir "hyperspezifisch" nennen, analysiert. Wir argumentieren, dass eine Kombination aus geleasten IPv4-Adressen und hyper-spezifischen Routen für viele Netze ausreichen dürfte, um die Zeit bis zur vollständigen Verfügbarkeit von IPv6 zu überbrücken.

Neben der IPv6-Einführung hat sich das Routing-Ökosystem auch in anderen Bereichen weiterentwickelt. Wir untersuchen zunächst exemplarisch das AS Path Prepending, um die Auswirkungen dieser Änderungen auf die Sicherheit zu bewerten. Wir haben eine typische Konfiguration gefunden, die keine Vorteile bringt, aber die Anfälligkeit eines Netzwerks für Präfix-Hijacks erhöht. Infrastrukturelle Änderungen führten zu einem allgemeinen Rückgang der global verwendeten Prepending-Längen im Laufe der Zeit und damit zu einem sichereren Einsatz der Technik. Wir zeigen jedoch, dass wir dieselben Änderungen ausnutzen können, um Präfix De-Aggregations-Angriffe zu konstruieren welche weit verbreitete Präventionsmechanismen überwinden können. Wir haben unsere Annahmen und unser Angriffsmodell anhand einer realen Testumgebung validiert und Aktualisierungen für bestehende Schutzmechanismen vorgeschlagen. Unsere zweistufige Aufklärungskampagne hat zu einem sichereren Routing-Ökosystem beigetragen.

List of Publications

Parts of this dissertation are based on either pre-published work or work that currently undergoes peer-review. These works are co-authored with other researchers as listed below.

International Conference Publications

P. Sermpezis, **L. Prehn**, M. Flores, S. Kostoglou, A. Vakali, E. Aben. "Unbiasing Internet Measurement Platforms." *7th Network Traffic Measurement and Analysis Conference (TMA). IEEE, 2023.* [Results appear in Chapter 3]

L. Prehn, F. Lichtblau, C. Dietzel, and A. Feldmann. "Peering Only? Analyzing the Reachability Benefits of Joining Large IXPs Today." *International Conference on Passive and Active Network Measurement (pp. 338-366), 2022.* [Results appear in Chapter 3]

L. Prehn and A. Feldmann. "How biased is our Validation (Data) for AS Relationships?" *Proceedings of the ACM Internet Measurement Conference (pp. 612-620), 2021.* [Results appear in Chapter 3]

L. Prehn, F. Lichtblau, and A. Feldmann. "When Wells Run Dry: The 2020 IPv4 Address Market." *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies (pp. 46-54), 2020.* [Results appear in Chapter 4.]

P. Marcos, **L. Prehn**, L. Leal, A. Dainotti, A. Feldmann, and M. Barcellos. "AS-Path Prepending: there is no rose without a thorn:" *Proceedings of the ACM Internet Measurement Conference (pp. 506-520), 2020* [Results appear in Chapter 5]

Peer-reviewed Journal Publications

K. Z. Sediqi, **L. Prehn**, and O. Gasser. "Hyper-Specific Prefixes: Gotta Enjoy the Little Things in Interdomain Routing." *ACM SIGCOMM Computer Communication Review 52.2 (2022): 20-34.* [Results appear in Chapter 4.]

Work under submission

L. Prehn and A. Feldmann. "How Short-Lived Routing Dynamics Distort AS Relationship Inferences." [Results appear in Chapter 3]

L. Prehn, P. Foremski, O. Gasser. "Kirin: Hitting the Internet with Millions of Distributed IPv6 Announcements." [Results appear in Chapter 5]

Posters & Demos

L. Prehn, A. Improta, A. Feldmann. "Two years into Quantum-Leaping Route Collector Visibility?" *International Conference on Passive and Active Network Measurement, 2020.*

L. Prehn and A. Feldmann. "Patch me if you can: Analyzing the potential of data sources to reveal the Internet's AS topology." *International Conference on Passive and Active Network Measurement, 2019.*

Contents

Chapter 1
Introduction

Initially starting as a research endeavour, the Internet has become a vital tool for modern society, providing individuals and businesses unparalleled access to information, communication, and opportunities for commerce and social interaction [201, 269, 287, 320]. It has played a crucial role in facilitating the democratization of information, allowing for greater access to education, news, and other forms of knowledge [169, 265, 387, 443]. Today, approximately 5.3 billion people [119] access the vast services offered by the Internet (e.g., social media, video streaming, or gaming) using a wide range of consumer devices such as smartphones, PCs, or wearables [143].

The Internet's routing ecosystem, which enables end-to-end communication between those devices, consists of tens of thousands of interconnected networks called Autonomous Systems (ASes). ASes exchange routing information (i.e., how to reach specific sets of destinations) via the Border Gateway Protocol (BGP). While BGP remained mostly unchanged over the decades, the routing ecosystem itself continuously evolved.

In the early 1990s, The routing ecosystem had a hierarchical structure [509] in which larger Internet Service Providers (ISPs) provided transit for smaller ones. Since then, the Internet's routing hierarchy began to flatten as ASes reduced their reliance on costly transit providers by establishing settlement-free peering agreements between each other [49, 71]. To facilitate peering, networks within the same region began establishing specialized facilities called Internet Exchange Points (IXPs). Today, there are thousands of IXPs [378], some forwarding tens of terabits per second of data between thousands of members [135]. The vast deployment of IXPs also enabled Content Delivery Networks (CDNs) to directly peer with thousands of networks and deliver their traffic as close to the customer as possible [122, 512].

Besides the vast deployment of new IXP infrastructures, the routing ecosystem is in the midst of adopting Internet Protocol Version 6 (IPv6). This adoption became necessary as the available Internet Protocol Version 4 (IPv4) address space rapidly diminished throughout the last decades. While the community monitored, analyzed, and decelerated (e.g., via the introduction of carrier-grade Network Address Translation (NAT) and stricter allocation policies) this process—which is also known as IPv4 Exhaustion—new allocation requests can no longer be fulfilled in all regions.

It is important for network operators to continuously monitor these changes to remain competitive. By understanding current and emerging trends, network operators can anticipate the impact of new technologies on their business, identify opportunities for growth and innovation, and proactively address (security) challenges. In addition to network operators, academics and policy-makers can also benefit from an updated view of the Internet's routing ecosystem. Insights about available resources, achievable performance, and critical issues can inform future system designs and help to determine the appropriate level of potential regulation.

The goal of this dissertation is to assess and improve our ability to model the Internet's routing ecosystem and its evolution and to investigate the ramifications of changes in the routing ecosystem for network operations and security. This goal is challenging as most of the Internet's protocols and applications were designed without measurability in mind [20]—an oversight that impacted recent design and enhancement efforts [213, 375]. Researchers and operators often have to exploit protocol-specific side-effects [50, 259, 382], re-purpose previously collected data sets [179, 293, 313], or deploy their own infrastructure [446, 448, 497] to even have a chance of measuring the Internet. For example, the widely used traceroute tool exploits the IP protocol's loop-prevention mechanism to obtain information about routers along the forwarding path to a destination [302]. These challenges motivate our first research question:

(1) How accurately can we model and track the Internet's routing ecosystem and its evolution with our deployed observation infrastructures and commonly used inference methods?

Once we have identified the limitations of our observation infrastructure and modelling methods, we are prepared to focus on the routing ecosystem's currently ongoing changes. One of the most noticed, analyzed, and discussed changes throughout the last decade is the rapid exhaustion of available IPv4 address space. While the Internet's routing ecosystem made significant progress adopting IPv6 (e.g., in protocol specifications [56, 314], hardware [112, 152], route announcements [224], or filtering recommendations [153, 356]), many ASes—and subsequently many services—do not support IPv6 yet [191, 202, 346]. This situation drives our second research question:

(2) How can network operators cope with the exhaustion of IPv4 addresses while parts of the Internet still lack sufficient IPv6 adoption?

Besides its ongoing adoption of IPv6, the routing ecosystem has drastically evolved regarding available peering infrastructure, involved stakeholders, and the complexity of routing operations. At the same time, some of BGP's initial shortcomings, such as the lack of authenticity and legitimacy of control information [262], were only addressed via a patchwork of supplementary security recommendations (e.g., RFD [498] and MAX-PREFIX [111] limits or ROV [330], ASPA [51], BGPSec [281], and MANRS [306] filtering) and their gradual adoption [108, 155, 192, 468]. Yet, as some of these proposals were designed decades ago with a substantially smaller, less complex, and less dynamic Internet structure in mind, we pose our third research question:

(3) How does the evolution of the routing ecosystem affect the security of routing operations?

In the remainder of this chapter, we first expand upon these three questions in §1.1, §1.2, and §1.3. Afterwards, we clarify the contributions of this dissertation in §1.4 and outline its content in §1.5.

1.1 Inferring the Routing Ecosystem's Structure

Inferring, analyzing, and tracking the Internet's routing structure is essential. It allows operators to design, grow, and manage networks more effectively [72, 447, 512, 518], informs the design process of novel protocols and systems [257, 262, 522], and aids policymakers in the decision process for regulatory interventions [429, 469, 527]. Hence, the operator community introduced route collectors to gather routing information. BGP route collectors are dedicated devices that peer with hundreds of volunteering ASes to receive, dump, and archive the routing information they propagate. As BGP's route redistribution is policy-based and strongly depends on the business relationships among ASes, route collectors provide only a limited view into the routing ecosystem [101, 188, 204, 435, 436]. Over the years, researchers introduced different data-plane and control-plane methods to partially overcome this lack of visibility [49, 103, 162, 188].

Beyond uncovering interconnections, route collector data is often used to infer the business relationships between ASes. These play a beneficial role in determining peering partners, estimating route propagation, or pinpointing performance bottlenecks. Throughout the last two decades, researchers proposed various inference algorithms using sets of heuristics, machine learning, or stochastic modelling to correctly label each AS link as either peering, transit, or sibling (i.e., two ASes that are owned by the same organisation) connection [167, 185, 188, 248, 251, 292]

This dissertation revisits our inference models for the routing ecosystem's structure. First, we infer the peering fabric of one of the world's largest IXPs; analyze its routes' availability, cost, and importance; and compare its interconnections to those visible via public route collectors. We then re-examine state-of-the-art business relationship inference algorithms focusing on their inference stability, their geographical and topological biases, and the generalizability of their evaluation performance. Finally, we introduce a simple yet easily extendable framework to detect bias in the placement of route collector vantage points.

1.2 Addressing Despite IPv4 Exhaustion.

ASes must obtain and announce address space to participate in the Internet's routing ecosystem. Besides this basic requirement, many networks benefit from additional addresses. With more address space, ASes may diversify their address assignment policies and route announcements, ultimately enabling fine-grained traffic engineering [174, 260, 396]. While IPv6 was introduced decades ago, only <40% of the top 1 million websites [202] and ~40% of Google users [191] support IPv6 in January 2023, forcing operators to rely on IPv4 addresses to ensure global accessibility of their addresses.

The resource allocation process. Traditionally, Internet resources (such as IP addresses and AS numbers) are requested and allocated at little cost by the five Regional Internet Registries (RIRs). As depicted in Figure 1.1, each RIR is responsible for the resource assignment, bookkeeping, and community support within its own service region, i.e., AFRINIC, APNIC, ARIN, LACNIC, and the RIPE NCC serve the African, Asia Pacific, American, Latin American, and European & Middle Eastern region, respectively.[1]

[1]Notably, there are some "legacy" resources assigned before the introduction of the RIR framework. While some of these were re-incorporated over the years, others still remain outside the control and management of the RIR framework.

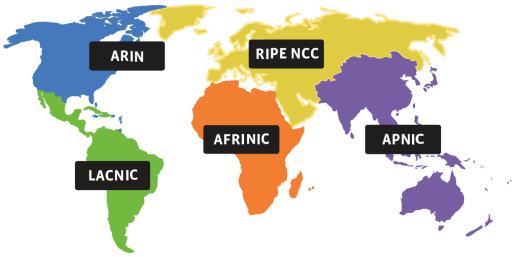


Figure 1.1: Geographic service regions for the 5 Regional Internet Registries, taken from [504].

IPv4 exhaustion. With tens of thousands of networks requesting address space across multiple decades, the RIRs have almost entirely exhausted the initial pool of available IPv4 addresses [363]. As of January 2023, most RIRs have started recovery programs requesting LIRs to return unused IPv4 address space to their allocation pools [28, 28, 127, 418]. In the meantime, most RIRs have no more IPv4 addresses left to allocate immediately; hence, allocation requests are added to a waiting list and fulfilled on a first-come-first-serve basis [29, 45, 277, 428]. Due to their shortened supply, IPv4 addresses have become a valuable asset [241], impeding their recovery. To illustrate: There are currently more than 1100 requests in RIPE NCC's waiting list, some of which are already waiting for more than 300 days [428]. In summary, the standard RIR allocation process is too slow to keep up with today's fast-paced business environments.

In this dissertation, we look closely at two options that network operators may choose to fulfil their addressing needs despite IPv4 exhaustion. First, we explore the availability, benefits, and cost of obtaining IPv4 resources via buying and leasing markets. Afterwards, we analyze the viability, potential use cases, and drawbacks of announcing and redistributing *hyper-specific* prefixes, i.e., prefixes so specific that operators commonly recommend filtering them.

1.3 Secure Routing Operations

The routing ecosystem is a complex system involving interactions between tens of thousands of networks. Despite most having genuine intentions, some networks use the Internet for malicious activities such as data theft, espionage, or infrastructural attacks to gain economic benefits, protest against political decisions, or limit the communication of conflicted countries [58, 300, 442, 476, 490, 506]. While some attack types—e.g., amplification-based Distributed Denial of Service (DDOS) attacks—target the victim's data-plane [148, 502], there are various exploitable control-plane issues resulting from BGP's lack of authenticity and legitimacy verification. In a BGP hijack, for example, an attacker announces the same prefix as another AS, generating a scenario in which the routes from two different ASes compete for the incoming traffic [331, 456, 487]. To limit the impact of BGP hijacks and similar (intentional and unintentional) attacks, operators retro-actively deploy prevention mechanisms such as Route Origin Validation (ROV) [488], BGPSec [281], or route filters [221, 319].

Despite their best efforts, the routing ecosystem and its respective routing threats continuously evolve. Hence, operators must frequently evaluate and update their prevention strategies to adapt to the evolving threat landscape. This dissertation further informs this process in two ways. First, it analyzes the usage, effectiveness, and security drawbacks of

AS path prepending—a prominent inter-domain traffic engineering technique. Second, it revisits prefix de-aggregation attacks and demonstrates that the continuous evolution of the Internet's routing ecosystem weakened the already deployed prevention mechanisms for such attacks.

1.4 Contributions

The goal of this dissertation is to assess and improve our ability to model the Internet's routing ecosystem and its evolution and to investigate the ramifications of changes in the routing ecosystem for network operations and security. We make the following contributions towards this goal:

Case study about the peering fabric and available routes at a large European IXP: We systematically analyze and compare route server snapshots from eight of the world's largest IXPs. We further analyse bilateral and private peering at one of those IXPs based on its peering LAN traffic and queries to carefully selected looking glasses. We compare the richness of the uncovered peering fabric with the view obtainable from public route collectors, and we assess the importance of available prefixes via two orthogonal metrics: the number of domains served from the prefix and the traffic volume that a large eyeball network egress towards it.

Re-examination of the generalizability of business relationship inference evaluations: We systematically analyze the geographical and topological biases within the sets of inferred and validated AS relationships. We empirically demonstrate that bias mismatches may affect classification correctness for three recent classification algorithms. We discuss, in-depth, different approaches for compiling less biased and more complete validation data sets and highlight (i) the need for active discourse with operators and (ii) how the routing ecosystem's continuous change can be exploited to over-sample validation data.

Analysis of the short-term stability of business relationship inferences: We systematically generate and analyze tens of thousands of input data sets that slightly differ by the data they use, the time window over which they are aggregated, and the exact time at which their time window starts. Based on these data sets, we perform an in-depth study of one of the most-used inference algorithms. Our results suggest that future inference efforts should consider the variance in inference performance introduced by short-term routing dynamics.

Framework to quantify bias in the AS-level placement of vantage points: We define bias in a multi-dimensional context and develop a simple, generic, and extendable framework to quantify the biases in the placement of internet measurement vantage points. We provide an in-depth study of the placement bias within RIPE Atlas, RIPE RIS, and RouteViews, and show that our framework confirms known issues while uncovering new ones. We show that our framework can be used to either sub-sample or further extend the existing infrastructure to minimize bias. We perform a blind test of our sub-sampling method, showing that it significantly reduced the bias for measuring the end-to-end latency distribution of a large content provider.

First look at the emerging IPv4 leasing and buying markets: We outline the current IPv4 exhaustion state, address allocation policy, and waiting list status for all five RIRs. We argue that traditional allocation requests for IPv4 resources are too slow for operator needs and focus on two newer business models: buying and leasing of IPv4 resources. We analyze the current IPv4 address transfer market based on RIR transfer statistics and privately obtained pricing data from four of the world's largest IPv4 brokers. We further

compare different methods to infer leasing agreements based on publicly available BGP data and RIPE's Registration Data Access Protocol (RDAP) database. Finally, we contextualize our findings with insights from discussions with IP brokers.

Analysis of the viability of hyper-specific prefix announcements: We perform an in-depth, longitudinal analysis of Hyper-Specific Prefixs (HSPs). We compare their visibility within the routing ecosystem with their prevalence in different routing databases and argue about their use cases based on insights from analysing CIDR sizes, BGP communities, and service hit rates. We reason about their accidental or intentional use and discuss how HSPs might be used and treated in the future.

Analysis of the usage patterns, effectiveness, and shortcomings of AS Path Prepending: We perform a longitudinal characterization of ASPP utilization, prepend sizes, and geographic policies. We use a real-world BGP testbed to quantify the effectiveness and security issues of path prepending experimentally. We identify prepending policies with no apparent ITE effect yet potentially detrimental security implications and analyze their prevalence in the wild. We discuss our findings with network operators and devise recommendations for using ASPP.

Re-evaluation of prefix de-aggregation attacks: In light of Internet flattening, we revisit the concept of BGP prefix de-aggregation attacks and show that current prevention mechanisms only transform such attacks into a session-hunting problem. We analyze the resources required to perform prefix de-aggregation attacks by formulating them as an Integer Linear Programming problem on top of real-world routing information. After demonstrating their theoretical feasibility, we test for practical hurdles by deploying the infrastructure required to perform a small-scale prefix de-aggregation attack using 4 IXPs, and validate our assumptions via BGP data analysis, real-world measurements, and router testbed experiments. We extensively discuss possible defence mechanisms and perform a two-stage vulnerability disclosure campaign.

1.5 Overview and Structure

We align the structure of this dissertation with the above-mentioned challenges and contributions:

Chapter 2 provides the required background information for this dissertation. This chapter covers, e.g., the Internet's current routing infrastructure, BGP, and routing policies.

Chapter 3 takes a closer look at how we infer the structure of the Internet's routing ecosystem. It first validates previous assumptions using a case study at a large European IXP. Then it analyzes biases in the validation (data) for business relationship inferences and quantifies the impact of short-term routing dynamics on one of the most prominent algorithms. Finally, it introduces a framework to quantify bias in the AS-level placement of vantage points. While some parts of this chapter have been pre-published in [389, 390], other parts are currently under submission.

Chapter 4 focuses on how operators may fulfil their addressing needs despite IPv4 exhaustion. Its analyses on the current IPv4 buying and leasing markets and the viability of hyper-specific prefix announcements are pre-published in [391, 452].

Chapter 5 dives deeper into the security aspects of routing operations. It analysis the usage patterns, effectiveness, and security drawbacks of AS Path prepending and further investigates the viability of prefix de-aggregation attacks in a hyper-connected routing

ecosystem. While the former has been pre-published in [312], the latter is currently under submission.

Chapter 6 concludes this dissertation with an in-depth discussion about its findings and greater impact.

Chapter 2
Background

This chapter will cover the background for this dissertation. We start with a summary of IP addresses, their allocation, and usage in Section 2.1. We then explain how networks communicate paths towards addresses in Section 2.2 and provide a high-level overview of the routing ecosystem's major components in Section 2.3. Finally, we provide an overview of frequently used vantage points and operator databases in Section 2.4 and provide a description of the neologism "routegazing" in Section 2.5.

2.1 Addressing

Throughout this section, we will discuss what kinds of addresses exist and how they are represented in subsection 2.1.1, how network operators can obtain addresses in subsection 2.1.2, and what types of special addresses exist in subsection 2.1.2.1.

2.1.1 Internet Protocol

The Internet uses Internet Protocol (IP) addresses to uniquely identify endpoints, i.e., devices or, more precisely, their interfaces. While the Internet governance body specified many different versions of IP addresses over the years, most of today's devices use IPv4 and IPv6 addresses. The IPv4 and IPv6 header represent addresses by 32 and 128 bits, respectively. While this limits the number of unique IPv4 addresses to roughly 4.3 Billion, there is a virtually infinite pool of 3.4×10^{38} unique IPv6 addresses [7]. IPv4 addresses are commonly expressed in dotted decimal notation, where each of the four octets is converted to an Integer and then joined via dots, e.g., `1.2.3.4` [129]. In contrast, IPv6 addresses are usually expressed in (shortened) colon-hexadecimal notation, where each of the eight hextets is represented in hexadecimal and joined via colons, e.g., `1:2:3:4:5:6:7:8`. The shortened format allows to omit 0-valued hextets, e.g., the IPv6 address `1:2:3:0:0:0:0:8` can be expressed as `1:2:3::8` [209]. Notably, there are many ways to represent IP addresses, yet we will stick to the two before-mentioned formats throughout this dissertation.

2.1.2 Resource Allocations & Assignments

In the Internet's earliest days, IP address blocks (and other Internet resources) were manually assigned and managed by John Postel. The management effort required to handle and track these assignments grew with increasing popularity until the current allocation and assignment framework was ultimately established. In this framework, the Internet Assigned Numbers Authority (IANA) received all addresses and subsequently allocated them to five Regional Internet Registries (RIRs). These five RIRs are AFRINIC, APNIC, ARIN, LACNIC, and RIPE and serve the African, Asia Pacific, American, Latin American, and European & Middle Eastern regions, respectively. Each of the five RIRs can further allocate address blocks to Local Internet Registries (LIRs)—which further assigns the allocated address space—or directly assign an address block to end users. Network operators need to send an address allocation request to either LIRs or their respective RIR in order to receive an assignment. Section 4.1 further provides additional details about existing requirements and current policies for this process.

2.1.2.1 Special (Purpose) Addresses

The Internet governance body has dedicated specific Internet resources to specific use cases, e.g., the IPv4 address `255.255.255.255` is used when an interface wants to broadcast a message to all devices within the same subnet. As these "reserved" resources have globally unique meanings, networks should not introduce them to the routing ecosystem; hence, they are also referred to as "bogon" resources.

Besides bogon resources, there are resources that were never integrated into the IANA/RIR framework. The resource holders received their resources directly from John Postel and avoided the integration process, as it would require them to abide by the RIR policies. These resources and their assignments are often called "legacy" resources/assignments.

2.1.3 Network Organisation

The routers within a network need to fulfil two tasks. First, the need to route, i.e., determine a feasible path towards a destination IP address. Second, they need to forward traffic, i.e., send a data packet to one of the connected interfaces. Yet, storing paths for all possible IPv4 and IPv6 addresses is memory-wise infeasible; hence, large numbers of IP addresses are grouped into subnets. These subnets represent continuous address blocks based on the left-most bits (also called network bits) that their binary representations share; therefore, they are also called prefixes. Prefixes are often represented in CIDR notation, i.e., as a combination of a network address (i.e., the lowest address within the described address block) and the number of network bits; to illustrate: `1.2.3.0/24` would be a prefix that would include all addresses between `1.2.3.0` and `1.2.3.255` (both included). To further reduce the memory footprint of paths and facilitate network management, networking devices virtually abstract the networking infrastructure under a single administrative entity into so-called Autonomous Systems (ASes) and represent them via a single Autonomous System Number (ASN). The combination of a prefix and an AS path is referred to as a route.

2.2 Inter-domain Routing

2.2.1 Border Gateway Protocol

In order to make their resources globally reachable, ASes (more accurately, their AS border routers) exchange routes between each other via the de-facto standard inter-domain routing protocol called the Border Gateway Protocol (BGP). Hereby, a BGP session runs on top of a Transmission Control Protocol (TCP) connection between two IP routers. Whenever an AS announces or redistributes a route, a route prepends its AS' ASN to the AS path, effectively assembling the AS path during the propagation process. Besides announcing a route, an AS can also withdraw a route, which signals other ASes that it is no longer available. Besides the prefix and AS path, route announcements can be tagged with additional attributes such as BGP Communities—4 [96] or 8 [207] Byte long binary fields with uniquely encoded meanings (except for a few well-defined values, see, e.g., [228, 264]). Community encodings can represent arbitrary things such as geographic locations [357] or the board state in a game of battleships [125]. Each router stores the currently available paths and their attributes for each prefix in its Routing Information Base (RIB) and further enters the IP-level next-hop for its chosen "best" path per prefix into its Forwarding Information Base (FIB).

2.2.2 Policies

BGP is a policy-based routing protocol that allows routers to apply policy-based actions at different points in time (e.g., when a route is received or redistributed via some session or before it enters the RIB). Policies are triggered based on, e.g., the session, router, or AS a route was received from, the specific prefix it describes, or the AS path it traversed—just to name a few. Once triggered, a policy may perform actions such as dropping/filtering an announcement, modifying the AS path or redistributing the route to a predefined set of neighbouring ASes. BGP policies often directly reflect or indirectly rely on business relationships with neighbouring ASes, e.g., it is recommended to filter announcements from customers that have already passed through major transit providers to limit the propagation of certain misconfigured announcements (also known as route leaks).

2.2.3 AS Business Relationships

While the business relationships between ASes may be complex and vary by, e.g., geographic location, connection type, or traffic volume, the academic literature often abstracts them into three different classes: (1) peering relationships allow ASes to exchange traffic and routes between each other and their respective customers without monetary compensation, (2) customer-to-provider (or transit) relationships in which one AS pays another to forward its traffic and access its routes, and (3) sibling relationships between ASes operated by the same organisation and hence can have arbitrary economic and routing-related policies. While no authoritative entity globally collects relationship information, a large branch of academic work focuses on the inference of AS business relationships. For a more in-depth discussion of state-of-the-art inference methods and additional details on the complexity of business relationships, please refer to Chapter 3, more specifically Sections 3.3 and 3.4.

2.3 Anatomy of the Internet Routing Ecosystem

The Internet's routing ecosystem consists of tens of thousands of interconnected networks with hundreds of thousands of interconnections. This section provides a high-level overview of the more prominent components and their characteristics.

2.3.1 Internet Service Providers

As their name implies, Internet Service Providers (ISPs) are ASes that provide Internet connectivity to their customers (e.g., other ASes or end-hosts). Besides typical transit, broadband, and mobile providers, there are many specialised types of ISP, such as high-frequency trading networks [68], low-Earth orbit satellite networks [67], or blockchain-based wireless networks [246]. ISPs are often classified into multiple tiers depending on their size and function. Tier-1 ISPs operate globally and often own subsea cable deployments and optical backbone infrastructure. Given their size, reach, and importance, Tier-1 networks can not rely on other transit providers but rather have to peer with all other Tier-1 ISPs to get access to the remainder of the Internet that is not (in)directly connected to them. Tier-2 ISPs are often referred to as National Service Providers (NSPs) as they rely on Tier-1 ISPs for most of their international traffic delivery. As their infrastructure spans a smaller geographic area than Tier-1 ISPs, they rarely own subsea cables yet usually operate optical backbones. The remaining ISPs are usually classified as Tier-3 ISPs. These ASes tend to cover only a small geographic area, often limited to certain regions within a single country.

2.3.2 Internet Exchange Points

Internet Exchange Points (IXPs) allow ASes in the same geographic region to cost-effectively interconnect via a shared Layer-2 peering Local Area Network (LAN). The LAN often spans multiple colocation data centers where hundreds of IXP participants establish a physical presence via AS border routers. Many IXPs operate multiple peering LANs (e.g., in multiple cities) and host value-adding services such as Route Servers (RSs), which allow the connected members to receive routes from many other participants easily. We provide deeper insights into the structure of IXPs, their available interconnection models, their peering opportunities, and associated costs in Chapter 3, more specifically Section 3.1.

2.3.3 Content and Cloud Providers

Major content and cloud providers (also known as "hypergiants") like Apple, Amazon, Facebook, Google, etc. originate substantial amounts of Internet traffic, e.g., Labovitz reported that up to 90 % of consumer traffic is served by hypergiants [270]. To hand off their vast amounts of traffic as cost and performance effective as possible, hypergiants deployed enough infrastructure to directly connect to tens of thousands of networks [49, 122]. Besides deploying their own networking infrastructure (e.g., data centers [200, 332], fiber lines [329], or submarine cables [190, 326]), they achieve such high degrees of connectivity by heavily relying on IXPs. To further improve content delivery, hypergiants also deploy content caches within other ASes—a strategy known as off-net deployments [179].

2.4 Vantage Points & Databases

There are a number of vantage points and databases that allow operators and academics to measure, secure, and debug the routing ecosystem. Throughout this section, we will briefly introduce those that we rely on later in the thesis.

2.4.1 Route Collectors

Route Collectors (RCs) are devices dedicated to receiving, dumping, and archiving routing information. RCs establish sessions with many ASes at the same physical location (e.g., the same IXP) or via multi-hop across the entire Internet. While they never send any traffic data and rarely[2] produce any route announcements, route collectors dump the BGP updates they receive into "Multi-Threaded Routing Toolkit (MRT)"-formatted [70] files. Besides update files, route collectors also dump a snapshot of their RIB periodically. The most well-known route collector projects are RIPE RIS [349] and Routeviews [364]. While Packet Clearing House (PCH) operates route collectors for approximately the same time as the two before-mentioned projects [379], accessing large amounts of historical data via their web front-end is prohibitively slow, leading to a limited usage of their data in academic studies. Isolario used to be a fast-growing route collector project that actively reached out to networks in under-represented locations [244], yet it was terminated at the end of 2021. In 2022, bgp.tools started to set up its own route collector project and already assembled more than 500 BGP sessions by the beginning of 2023 [64]. Nowadays, MRT data is most commonly accessed via either BGPStream [109] or BGPKit [62], yet older alternatives that work on already downloaded MRT files, e.g., BGPScanner [243] or BGPdump [350], are still functional.

2.4.2 RIPE Atlas

RIPE Atlas is a measurement platform with probing devices in thousands of ASes. Users that host a probe continuously earn credits which can then be used to run active measurements (e.g., ping, traceroute, or DNS lookups). RIPE also provides researchers with Atlas credits upon request. While each probe supports a minimal set of measurement types, certain probes (e.g., the better-equipped Atlas Anchor probes) have additional measurement types (such as throughput measurements). The RIPE Atlas platform is frequently used for measurement campaigns due to its easy accessibility, unified measurement interface, and broad coverage.

2.4.3 The PEERING Testbed

The PEERING testbed is an AS that allocates networking resources and infrastructure to experiments proposed by researchers [444, 448]. Once an experiment proposal is approved, an experiment receives resources (e.g., dedicated prefixes), credentials (to access and control the infrastructure), and capabilities (that restrict the types of actions that can be performed as part of the experiment). Using this setup, researchers can perform real-world experiments by announcing BGP routes via more than a hundred ASes directly connected to the PEERING testbed.

[2] some periodically announce and withdraw routing beacons, see [348]

2.4.4 Internet Routing Registries & WHOIS

Internet Routing Registrys (IRRs) are a set of distributed databases that contain information about the routing policies of ASes (e.g., an AS may enter a set of neighbours from which it imports all routes) in the Routing Policy Specification Language (RPSL). In a similar fashion, the WHOIS protocol (and respective service) allows users to query registration information (e.g., the maintainer, the responsible RIR, or a point of contact) for Internet resources (e.g., ASNs or IP addresses) via a set of distributed databases.

2.4.5 PeeringDB & EuroIX

PeeringDB and EuroIX freely provide user-maintained information and statistics about IXPs, their members, and their infrastructure [159, 381]. While both organisations are non-profit and community-driven, their information is often the first stop for finding new peering partners, potential collocation facilities, and other peering-related details. Both databases can be accessed via APIs, and CAIDA even generates daily snapshots of the PeeringDB database [90].

2.4.6 Resource Public Key Infrastructure

The Resource Public Key Infrastructure (RPKI) is the most widely adopted framework to secure routing operations. One of its most prominent features is Route Origin Validation (ROV)—a mechanism to validate the legitimacy of route announcements based on signed Route Origin Authorization (ROA) records. Resource holders can generate signed ROA records for their resources, and ASes can then incorporate the ROA validation state of a route into their BGP policy rules (e.g., drop the route if the ROA status is invalid).

2.5 Routegazing

The term "routegazing" describes the act of stargazing in a routing context. Like an astronomer who studies the stars, I also gazed at billions of routes to uncover their mysteries. The more I explored this complex system, with its incompatible components, erratic configurations, and unexpected events, the more my curiosity was piqued to venture into previously uncharted territory. Similar to the vastness of space, it is the wealth of unexplored and stimulating areas that, I believe, not only propelled past research but will also inspire the future exploration of the routing ecosystem.

Chapter 3
Modelling the Routing Ecosystem

Understanding the Internet's routing ecosystem is essential for troubleshooting connectivity issues, optimizing network performance, and making more informed decisions about network design, infrastructure, and policies. It further helps to anticipate, mitigate, and prevent potential security vulnerabilities, informs the regulation finding process of policy makers, and aids in the development of new protocols and systems.

Quality of the Observation infrastructure. For multiple decades, academics used route collectors—which were initially deployed by operators to track the propagation of their announcements—to study the Internet's routing ecosystem. Over the years, various independent studies unearthed problems in the assertiveness of route collector data. While routing policies prevent route collectors to obtain full visibility into the routing ecosystem, their vantage-point placement is also highly skewed towards larger, more-central ASes. While route collector platforms are keen to provide meaningful data to their community, their acquisition models are largely based on passively receiving peering requests. The first half of this chapter, in particular §3.1 and §3.2, will focus on re-assessing and reducing the limitations of the current route collector infrastructure.

Quality of the Business relationship inferences. While tracking the ecosystem's current routes and interconnections is valuable to assess its size and complexity, many essential tasks (e.g., predicting route propagation, inferring spoofing activity, or monitoring inter-domain congestion) require explicit knowledge of the business relationships among ASes. Hence, a large branch of literature focuses on finding and refining methods to infer business relationships based on primarily routing information using, e.g., domain knowledge in the form of heuristics, machine learning, or stochastic modelling. The second half of this chapter, specifically §3.3 and §3.4, will take a closer look at the most prominent AS business relationship inference algorithms, focusing on the effect of short-term routing dynamics and the existence of biases within our evaluation data.

The contributions of this chapter can be summarized as follows:

- We analyze and compare Route Server snapshots from eight of the ten largest IXP peering LANs worldwide. We find that all Route Servers show consistent insights: (1) only 10 % of Route Server peers provide more than 100 routes while 30% provide less than ten routes, (2) approximately half of the Route Server routes have

a minimum path length of three ASes (announced by close and distance peers alike) and about two-thirds of all routes lead to out-of-continent destinations, and (3) most large Route Servers have a prefix overlap of ~50 % while the actually reachable IPs overlap by ~60-70 %. We then run a case study on one of Europe's largest IXPs. We infer routes available via bi-lateral and private peering. Similar to Ager et al. [15], we observe that most ASes use the switching fabric to establish additional transit sessions. As such connections can drastically influence our inferences of available routes, we developed a methodology to increase the coverage of relationship inference algorithms at IXPs from initially only 22 % to 74 %, and we use the resulting relationships to isolate transit connections during the inference process. Similarly, we introduce a methodology to infer routes available via private peering based on the careful selection and querying of looking glass utilities. Finally, we compare the IPv4 and IPv6 routes available via multi-lateral, bi-lateral, and private peering against two top-10K prefix lists: one based on the number of served domains and one based on the traffic volume of a large European eyeball network (see §6). We find that nearly all top-10k IPv4 prefixes are available via bi-lateral peering. For IPv6, we observe that prefixes serving many domains are often unavailable (up to 15 %) or can only be obtained via private peering.

- We define bias in a multi-dimensional context and present a simple yet generic and easily-extendable framework to quantify the biases in the AS-level placement of vantage-points. When analyze the biases of RIPE Atlas, RIPE RIS, and RouteViews, our framework clearly confirms well-known biases, e.g., RIPE RIS is heavily biased towards larger networks and IXPs. Yet, it can go beyond these observations, e.g., we show that while networks that peer at many IXPs are over-represented in RIPE RIS, their peering policies are representative of the Internet's peering ecosystem (as captured by PeeringDB). Leveraging our framework, we design methodologies to reduce bias when using the existing platforms, and demonstrate based on a real-world use-case of a large anycast CDN that reduced placement bias may yield more representative measurements. Finally, we focus on strategies to guide the acquisition process for new vantage points and show that placement bias can be reduced while following independent goals.

- We analyze to which degree the geographical and topological biases within sets of inferred and validated relationships match. We uncover significant mismatches: While the "best-effort" validation data covers 31 % of all links between ASes in the ARIN region, it only covers less than 1 % of links in the LACNIC region. Yet, both regions contain roughly 15 % of the inferred relationships. We further analyze how such bias mismatches may affect classification correctness for three (ASRank [292], ProbLink [248], and TopoScope [251]) classification algorithms and uncover substantial drops in precision for certain groups of peering links. In particular, we observe that the near-perfect precision of 96-98 % for the entire validation data set drops by 14-25 % (depending on the algorithm) for peering relationships between Tier-1 and transit providers. Finally, we discuss, in-depth, different approaches for compiling less biased and more complete validation data sets and highlight (i) the need for active discourse with operators and (ii) how the routing ecosystem's continuous change can be exploited to over-sample validation data.

- We systematically generate tens of thousands of input data sets for ASRank that slightly differ by the data they use (RIBS and/or updates), the time window over which they are aggregated, and the exact time at which their time window starts.

When analysing the differences between the inference outcomes for these data sets, we find the following: We first uncover that ASRank's inferred clique is highly sensitive to one of its input parameters and frequently includes "hypergiants" (e.g., Akamai or Amazon) as the transit degree metric relies on an imperfect assumption. We show that ASRank infers ~94 % of all links consistently (i.e., with the same label each time) through- out our three month period. When extending this observation to the validation phase, we distinguish between two classes of errors: *persistent* and *transient*. While the former occur in all input sets and hint at deeper algorithmic problems, the latter change their label across different input sets, likely due to short-term routing changes. Even though only ~6 % of links are inconsistently inferred, 55 % and 85 % of all inference errors for the median and worst snapshot are transient, respectively. While recent works achieved a 1.6× error-rate reduction over ASRank for certain snapshots, we show that ASRank's error-rate can be reduced by 5.4× just by picking a different time for the evaluation. We conclude our work with insights into the minimum requirements needed to accurately detect the impact of short-term routing dynamics in future evaluation efforts.

3.1 Case Study: a Large European IXP

Traditionally, the Internet follows a hierarchical structure. At the top of this hierarchy resides a set of large transit providers—also called Tier 1 networks—that exchange traffic with each other at no monetary compensation. The literature commonly refers to this type of interconnection (and business relation) between two ASes as "peering."

When logically descending from the top, higher-tier networks deliver traffic for their lower-tier customers, i.e., they provide transit. Since the early 2000s, the "topology flattening" phenomenon gradually superseded this hierarchical structure. Lower-tier networks started to shift more of their transit traffic to newly established peering connections. The continuous acquisition of new peering partners is often incentivised by cost reduction and potential latency improvements [16].

The fast and widespread deployment of Internet eXchange Points (IXPs) has further accelerated the establishment of new peering connections. Traditionally, IXPs allow physically-close networks to exchange traffic via a shared layer-2 switching fabric; thus, they eliminate unnecessary routing detours, which reduces the overall latency and helps to "keep local traffic local." Today, the largest IXPs have grown to multiple hundreds—sometimes even thousands—of members (see Figure 3.1) and handle peak traffic volumes of more than 10 Tb/s [22, 134, 245].

As different networks have different negotiation positions, various forms of peering have emerged. The simplest form, bi-lateral peering, refers to a direct connection between two ASes via the IXP's switching fabric.

To ease the life of their customers, most IXPs also offer Route Servers that redistribute all routes they received from one IXP member to all others via a single BGP session per member. As this form of peering involves more than two networks, the community refers to it as multi-lateral peering. As a third option, networks can establish private peering sessions amongst each other. Instead of using the IXP's layer-2 fabric, ASes establish these peering sessions via a dedicated cross-connect in the same colocation facility (or via layer-2 transport for different colocation facilities).

While peering itself is a well-established concept that has been broadly discussed in the research literature (e.g., [15, 49, 71, 72, 94, 101, 288, 311]), we still lack fundamental

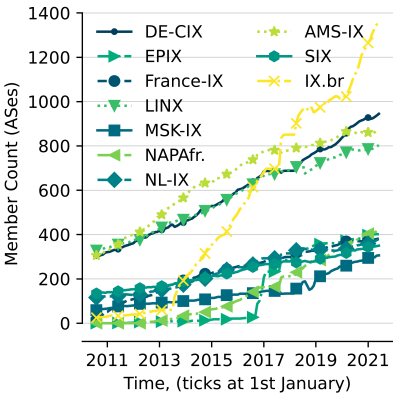


Figure 3.1: Number of members over time based on PeeringDB

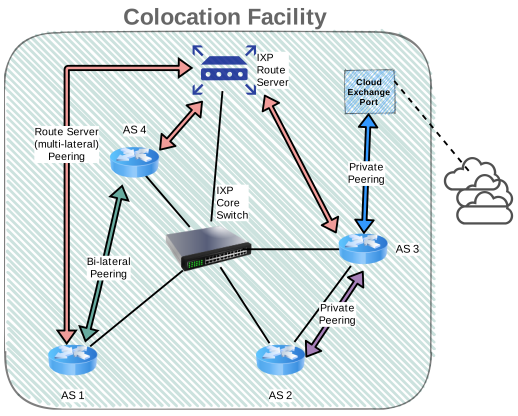


Figure 3.2: Illustration of different peering types at an IXP.

insights into the actual extent and importance of the routes available at large IXPs. In this section, we take a closer look at how the different forms of peering translate into transit-free prefix reachability. We characterize and compare the multi-lateral peering routes available at the Route Servers of the world's largest IXPs and further estimate the bi-lateral and private peering routes available at one large IXP in Europe that we refer to as L-IXP. We contrast our reachability analysis using two dimensions of importance: the number of top domains that a route serves and the traffic volume that one of the largest European eyeball networks egresses towards it. In particular, our contributions can be summarized as follows:

- **Characterization of Multilateral Peering:** We analyze and compare Route Server snapshots from eight of the ten largest IXP peering LANs worldwide (see, §3.1.3). We find that all Route Servers show consistent insights: (1) only 10 % of Route Server peers provide more than 100 routes while 30 % provide less than ten routes, (2) approximately half of the Route Server routes have a minimum path length of three ASes (announced by close and distance peers alike) and about two-thirds of all routes lead to out-of-continent destinations, and (3) most large Route Servers have a prefix overlap of ~50 % while the actually reachable IPs overlap by ~60-70 %.
- **Characterization of Bi-lateral & Private Peering:** For one of Europe's largest IXPs, we infer routes available via bi-lateral and private peering (see §3.1.4.1). Similar to Ager et al. [15], we observe that most ASes use the switching fabric to establish additional transit sessions. As such connections can drastically influence our inferences of available routes, we developed a methodology to increase the coverage of relationship inference algorithms at IXPs, and we use the resulting relationships to isolate transit connections during the inference process. Similarly, we introduce a methodology to infer routes available via private peering based on the careful selection and querying of looking glass utilities.
- **Route Importance:** We compare the IPv4 and IPv6 routes available via multi-lateral, bi-lateral, and private peering against two top-10K prefix lists: one based on the number of served domains and one based on the traffic volume of a large European eyeball network (see §3.1.5). We find that nearly all top-10k IPv4 prefixes are available via bi-lateral peering. For IPv6, we observe that prefixes serving many domains are often unavailable (up to 15 %) or can only be obtained via private peering.

3.1.1 Background

In this section, we provide an introduction to the different interconnection models and highlight important observations from related work. We refer to Figure 3.2 as a visualization of the individual components explained throughout this section. While interconnection agreements can be rather complex in practice, the scientific literature abstracts mainly into two categories: transit and peering.

In a transit agreement, a customer pays a transit provider for delivering its traffic from its egress router to any IP. In a (settlement-free) peering agreement, two ASes—usually of similar size and with roughly equal traffic volume towards each other—forward each other's traffic without substantial amounts of money flowing in either direction. As neither of the peering partners is a provider for the other, both ASes have to negotiate where to physically interconnect and who is bearing the infrastructure costs. Over time and with the spread of Internet Exchange Points across the globe the peering ecosystem itself became rather complex and different peering practices emerged. In the following, we give an overview of the fundamentals of current peering models.

Internet Exchange Points. As establishing a single BGP peering session for every interconnection partner separately is rather wasteful, operators started building common switching infrastructure that could be shared (w.r.t. usage and cost) among ASes. These switching infrastructures—envisioned to keep local traffic local— belong to so-called Internet eXchange Points (IXPs) located in well-connected colocation facilities. Those colocation facilities provide dedicated infrastructure (e.g., rack space, electricity, and cooling) for the housing of peering equipment. Figure 3.2 gives an abstract example for a layer-2 peering fabric. While IXPs may attract very diverse sets of members, previous work reported that they observe traffic for 40 % or more of all theoretically possible peering connections [76]. As some large IXPs observe traffic originated by or destinated towards tens of thousands of ASes and millions of servers [101] and could theoretically reach 70 % of all routed addresses [71], it nowadays is also common that networks pay remote-peering providers to get access to remote IXPs [94]. A recent study by Nomikos et al. [360] revealed that around 90 % of 30 tested IXPs had more than 10 % of their members connecting via remote peering. They further reported that for certain large IXPs up to 40 % of members can be connected via remote-peering.

Bi-lateral Peering. This practice describes a BGP peering session between two member ASes at an IXP via the shared peering fabric as depicted in Figure 3.2 (green arrows). While legal processes and concerns of peering policy leakage slow down the acquisition of bi-lateral peering partners [310], Marcos et al. proposed a framework that allows IXP members to quickly provision peering sessions based on an intent abstraction and digitally handled legal contracts [311]. Interestingly, Ager et al. showed in 2012 that also Tier1 providers peer at IXPs and that they use their IXP peerings not only as backup routes. They further showed that these Tier1 providers also abuse the peering LAN for transit connections to their customers [15].

Multi-lateral Peering. As briefly discussed previously, IXPs provide a Route Server for their members to establish multi-lateral peerings. In addition to reducing the number of needed interconnections to reach most IXP members[3], Route Servers can also implement additional functionality (e.g., the frequently used per-peer blackholing [148]) to make them more attractive to IXP members. Those services are often realized by attaching a specifically formatted BGP Community onto Route Server announcements. As a route

[3] A Route Server reduces the number of totally needed BGP sessions for a fully-meshed topology from $n * (n-1)/2$ to n, where n is the number of BGP speakers.

server has to store such information to act properly based on it, some IXP members do not establish a session with the route server as they expect that it might expose their peering policies [105]. As a notable example of such exposition, Giotsas et al. showed that it is possible to uncover 200k multi-lateral peering agreements by analyzing the BGP community values visible at few Route Servers [188].

Private Peering. When present at the same colocation facility, e.g. because they are members of the same IXP, two networks can establish a private peering session via direct cross-connect avoiding the IXP's peering fabric. Especially large ASes prefer this peering practice as it is provides a very fine-grained control over their peering sessions. Hence, networks that, e.g., need to egress a high traffic volume often require direct peering sessions on dedicated physical infrastructure with guaranteed capacity. This form of interconnection usually comes with monetary compensation for certain Service-Level Agreements (SLAs). Even though private peering keeps the peering policies of an AS hidden and often provides dedicated capacity, even private peering sessions can suffer from outages when, e.g., the entire colocation facility goes down—a not so uncommon scenario as Giotsas et al. reported (160 outages in 5 years) [183].

Cloud and Content Provider Connectivity. Many Cloud and Content providers peer at hundreds of physically distinct locations [72] to thousands of different networks [49]. While they often require private peering connections, they sometimes also rely on bi-lateral peering to ensure that they directly connect with as many eyeball ASes as possible [106] or to gain tens of milliseconds of latency improvements over their transit providers [446]. Hence, it is unsurprising that those providers also dominate the peering LAN traffic (as shown for two medium-sized IXPs by Cardona et al. [92]). Yet, as most networks try to establish private peering connections with them directly in the colocation facilities, those facilities have established so-called cloud exchanges—specific ports which directly provide connectivity, called Virtual Private Interconnections (VPIs), to any number of cloud service providers within the colocation facility [519].

Identifying Peering Partners. Many network operators rely on a network policy database called PeeringDB to identify potential peering partners [381]. In particular, PeeringDB differentiates between four peering policy types: (1) open: A network with an open peering policy that peers with any other network, (2) selective: A network that will peer under certain conditions, e.g., minimum traffic volume or location, (3) restrictive: A network that already has an existing set of peers and needs strong, convincing arguments to establish a peering connection, and (4) no peering: These networks do not peer at all and rely entirely upon transit [362]. Notably, the vast majority of peering policies in PeeringDB are of the 'open' type. Yet, PeeringDB is known to have certain inaccurate entries [288, 467]. Further, many small networks—especially in developing regions—do simply not register in PeeringDB [288].

3.1.2 Preface: Data Sets

While we introduce each data set separately when using it, this section summarizes the used data sets to provide a better overview of time coherence and caveats.

3.1.2.1 Main Data Sets

PeeringDB snapshots (2010/08/01—2021/06/01, monthly). PeeringDB is a community-effort database containing information about the infrastructure and policies for IXPs, colocation facilities, peering LANs, and networks [381]. PeeringDB is known to have

a small set of inaccurate entries [288, 467]. Similarly, Lodhi et al. reported that PeeringDB underrepresents small—especially developing country—networks [288]. The Center for Applied Internet Data Analysis (CAIDA) produces monthly snapshots of this database [90].

Route Server snapshots (2021/06/06—21, once). WWe compiled a set of Route Server snapshots for the largest (in terms of members) peering LAN for eight of the world's largest IXPs. We received these snapshots via multiple personal contacts throughout 15 days.

IXP traffic data (2021/05/01-2021/06/07). We obtain IPFIX traffic captures from one of the largest European IXPs. The traffic is sampled at a rate of 1 out of 10K (1:10k) flows. The captures encompass all traffic exchanged via the peering LAN; hence, it contains traffic exchanged via multi-lateral and bi-lateral peering sessions but misses private peering traffic. In particular, we utilize the data from May 2021 to analyze how our observation period influences our results and subsequently report most of our results based on the first week in June 2021.

ISP traffic data (2021/06/10). We obtain a single workday of egress traffic captured from all border routers from a large European eyeball network. The data was sampled at a rate of 1:1K packets.

Domain-based prefix top list (2021/04/30). We obtain a recently recomputed domain-based prefix top list from Naab et al. [336]. Their methodology relies on a domain top list as input, then resolves those domains to IP addresses from a single physical location, and finally aggregated the number of Fully Qualified Domain Names that is served by every norm-prefix (i.e., a /24 prefix in IPv4 and a /48 prefix in IPv6). We use the prefix top list that relied on Umbrella's domain top list [115] as input, as it was the only one that could provide us with 10K IPv6 prefixes. Notably, this domain-based prefix top list is biased towards the European service region as DNS load-balancing [449] and caching [399] may lead to strongly regionalized address resolutions.

Please note that we handled our traffic data sets in compliance with **measurement ethics** and best practices. We performed all data analyses on servers located at the respective premises of our vantage points using data collected as a part of their routine network analysis. We analyzed flow data summaries based on packet headers that did not reveal any payload information. We further anonymized all flow attributes not explicitly needed for the results presented in this section. This is in line with Ethical Committee policies. For the remaining data sets, we rely on publicly available sources only.

3.1.2.2 Orthogonal Data Sets

Maxmind GeoLite2 snapshot (2021/06/01). We utilize a snapshot of Maxmind's GeoLite2 database [316] to geolocate Route Server prefixes. While they can have significant inaccuracies on a city or country-level [97], even freely available databases achieve near-perfect continent-level predictions [315].

CAIDA's AS relationships snapshot (2021/06/01). CAIDA produces monthly snapshots of the business relationships inferred by ASRank [292] based on routing information collected by RouteViews [364] and RIPE/RIS [343] from the first five days within the month [87]. While it misses many peering links, this data is reasonably complete for transit links [204, 365, 368]. Further, the inference algorithm is known to near-perfectly infer transit relationships but often misinfers peering relationships as transit [167, 248, 251], i.e., it overestimates the number of transit relationships.

CAIDA's IP-to-AS mapping snapshot (2021/06/10). CAIDA generates daily IP to AS mappings based on routing information from selected Route Views [364] collectors [86].

CAIDA's AS-to-Org mapping snapshot (2021/04/01). CAIDA produces quarterly snapshots of AS-to-Organization mappings generated based on the WHOIS databases of all Regional and some National Internet Registries [219]. Notably, WHOIS data is known to contain malformatted and hard-to-parse entries [284], leading to potential inaccuracies in the inferred AS-to-Organization mapping. The April snapshot is the latest available snapshot before our measurement period.

3.1.3 Multilateral Peering

We start our analysis with the lowest-hanging fruit: multi-lateral peering. While some IXPs have explicit APIs that could be used to re-build the current routing table of their route servers, we explicitly request Route Server snapshots for the largest peering LAN of different IXPs. Out of the ten IXPs shown in Figure 3.1, only Nl-IX and EPIX did not fulfil our request. Our eight Route Server snapshots are from differents days between 6th and 21st June, 2021[4] and contain the entire routing information base for each session, i.e., they contain all paths from all neighbours (rather than just one best path) for a given prefix. Using those snapshots, we look at what routes an AS may expect from the Route Server and how consistent those findings are across different IXP Route Server. In particular, we arrive at the following takeaways:

- Large Route Servers across the world are very similar: They not only have the same distribution of routes per peer but also share the majority of reachable prefixes and IPs, i.e., joining a second, third, etc. Route Server only negligibly improves reachability.
- Due to the growing trend of remote peering, Route Servers provide only a limited amount of in-continent routes.
- We observe that most routes (at all analyzed Route Servers) contain at least three hops. While both close and distant peers announce those lengthy, unattractive routes, we find that members often only use one-hop Route Server routes.

How consistent are the distribution of routes to peers across route servers? Our snapshots show that connecting to the Route Server immediately provides routes from up to 650 IXP members. Yet, Richter et al. already reported that not all IXP members announce the same number of prefixes [412]. As a first look at how similar Route Servers are, we analyze whether this distribution is consistent across them. Figure 3.3 shows the number of prefixes (y-axis, logarithmic) announced by every peer (x-axis) per Route Server. Indeed, we observe strong consistency across different IXP Route Servers regardless of the protocol. For the AMS-IX Route Server (top curve), the top ~1.5, 10, 30, and 70 % of Route Server peers announce routes for more than 10K, 1K, 100, and 10 IPv4 (1K, 100, 20, and 5 IPv6) prefixes. While most Route Servers are close to AMS-IX, peers at NAPAfrica (bottom curve) announce around an order of magnitude fewer prefixes.e fewer prefixes, most other IXPs are closer to AMS-IX.

Notably, not all prefixes are necessarily exported to all peers by the Route Server. To estimate how many prefixes can only be received conditionally, we inspect the Route

[4] As we obtained similar results for all Route Server related plots for a set of inital snapshots that we obtained throughout January and February, we do not expect any major inconsistencies due to a two week offset.

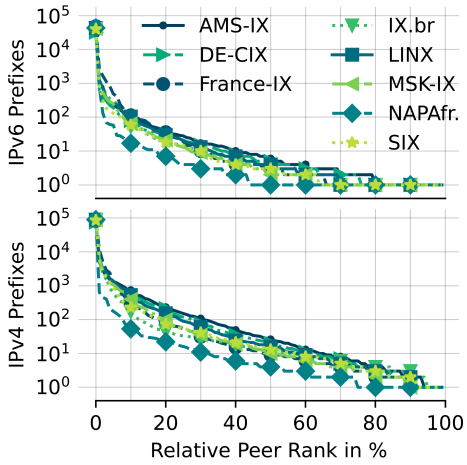


Figure 3.3: Number of prefixes announced per peer

Server snapshots for BGP communities that control its redistribution rules. For, e.g., DE-CIX, we inspect routes with the `0:6695` Community that is used to exclude all peers; this community is usually combined with other BGP Communities of the form `6695:X` which instruct the Route Server to explicitly redistribute a route to peer `X`. Overall, we find that 31.3 % of IPv4 and 11.2 % of IPv6 Route Server prefixes are not globally exported.

Do Route Servers help to keep local traffic local? As briefly discussed in §3.1.1, IXPs initially were established as a solution to interconnect geographically close ASes following the idea to "keep local traffic local." Yet, given that many peers announce tens of thousands of prefixes to hundreds of millions of hosts, we now want to take a look at how strictly this idea is followed through by today's Route Servers. We first use a naïve approach to answering this question: We look at the AS path length (after removing AS Path Prepending). Figure 3.4 shows the Route Server prefixes of different IXPs separated by the number of ASes in their shortest route. We observe that for around half of all prefixes the shortest path contains three or more ASes. This result goes against the "keep local traffic local" idea, as local routes would likely either directly lead to an access/eyeball network or indirectly via a national service provider. However, given that the AS path length is often not a good proxy for geographic distance, we now switch to a more insightful perspective.

Rather than looking at the AS path, we now directly map the visible prefixes to countries and continents using a snapshot of Maxmind's GeoLite2 database [316] from 1st June 2021. While perfect IP-to-geolocation mapping is a long-standing research problem, previous work showed that for various public geolocation databases 99 % of predictions stay within 600 km of the actual location [97]. Similarly, Maxmind claims that for many countries 0 % of predictions are off by more than 250 km [315]. While this large radius might influence the accuracy of country-level predictions, it provides us with near-perfect accuracy for continental predictions as most of our Route Servers have even more distance between their location and the closest continental border. Figure 3.5 shows the Route Server prefixes of different IXPs separated by whether they lead to in-country, in-continent, or out-of-continent ("other") hosts. Notably, there is a small number of prefixes for which the database did not include a mapping ("NA"). Interestingly, looking at host locations provides an even more drastic result than looking at AS paths: Regardless of

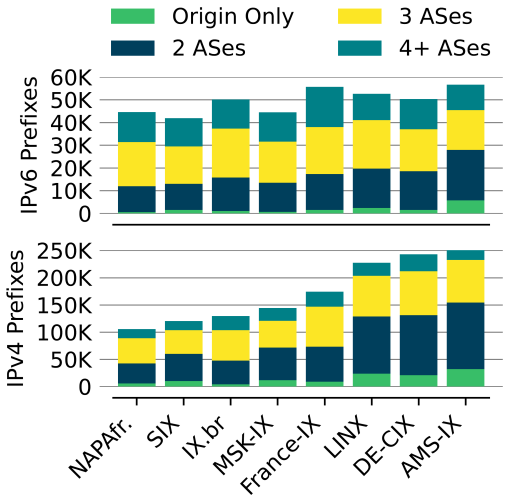


Figure 3.4: Length of shortest AS path per prefix

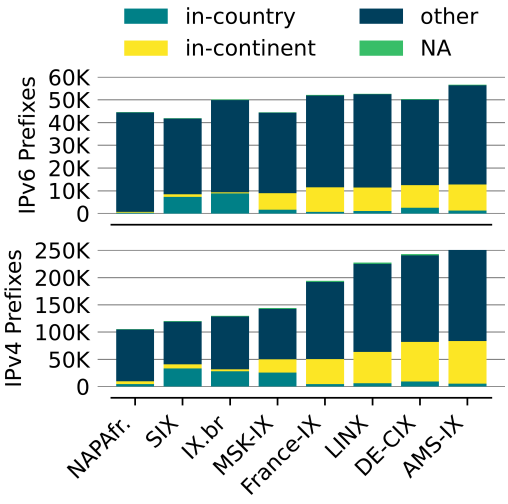


Figure 3.5: Geolocation of prefixes relative to Route Server

the actual Route Server, around two-thirds of all prefixes lead to out-of-continent hosts.

While the growing trend of remote-peering [360] can easily lead to many out-of-continent routes, it is unclear whether it also contributes to the high number of lengthy routes. To better understand whether this correlation exists, we want to compare the path length of each route with the RTT (as a proxy for distance) to its next-hop interface. Hence, we run ping measurements from a server directly connected to the switching fabric of L-IXP towards each member interface.[5] To account for latency inflations due to, e.g., congestion, we repeated those measurements 100 times and collected the minimum RTT towards each interface throughout all runs. Finally, we associate the shortest path of each prefix with the minimum RTT we measured for its respective next-hop interface. Notably, if there was more than one possible shortest path, we picked the one for which the next-hop RTT was the lowest. Figure 3.6 shows for each prefix of a given minimum path length the minimum latency to its next-hop.[6] We observe that there is no strict correlation between the distance of a peer and the length of the routes it provides.

Now that we know that even local peers forward lengthy routes to the route server, the question becomes whether those routes see any traffic. For one of our observed IXPs, we obtained IPFIX captures sampling 1 out of every 10K packets traversing its peering LAN. While we can observe multilateral and bi-lateral peering traffic in this data set, we have no insights into traffic exchanged via private peering established via direct interconnects as it does not traverse the public peering infrastructure. Based on the captured flows between the 1st of June and the 7th of June[7], we calculate the aggregated number of Bytes destined towards each prefix. Figure 3.7 groups Route Server prefix by their shortest path and shows for each prefix (x-axis) the number of bytes (y-axis, logarithmic) relative to the prefix with the most bytes (i.e., we show bytes normalized by the prefix with the maximum byte count, ρ). We observe that 6 % of prefixes reachable via one hop carry at

[5]We neither had probing devices at other peering LANs, nor was our probing device at L-IXP IPv6-enabled at the time of our study.

[6]We explicitly avoid the classification into remote and local peers based on RTT estimates alone given the caveats presented in [360]

[7]We provide details on how we choose this time window in the next section.

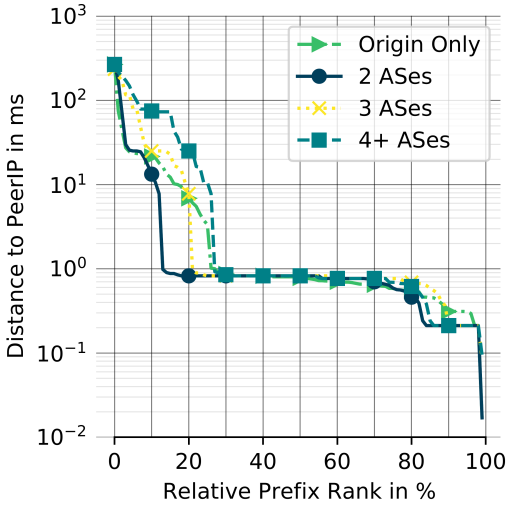


Figure 3.6: Distance to next-hop per prefix, separated by length of shortest AS path

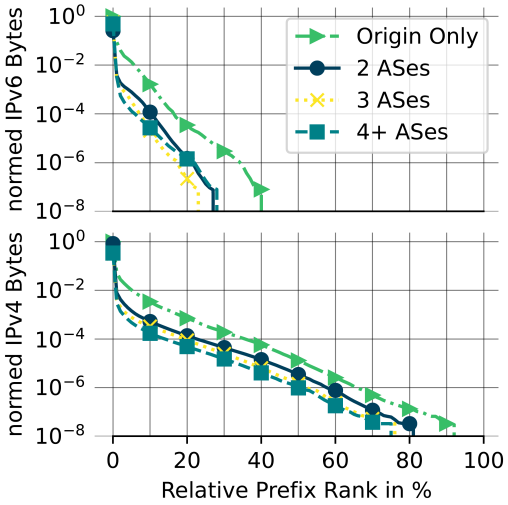


Figure 3.7: peering LAN bytes per prefix, separated by length of shortest AS path

least 1 % of ρ's bytes while only less than 0.5 % of 2 or more hop prefixes carry that much traffic. Apart from the top 6 %, prefixes reachable via two or more hops carry around an order of magnitude less traffic—with only minor differences between two, three, and four or more hops. Finally, we observe that 8, 19, 24, and 25 % of IPv4 (60, 72, 73, and 77 % of IPv6) prefixes with a shortest path of 1, 2, 3, and 4+ hops carry no traffic at all, respectively.

Those observations are likely tied to how long-established IXP members engage with a Route Server: In contrast to new members, long-established members already acquired many bi-lateral peering sessions. It is common that members attribute higher local preference values to such bi-lateral sessions as they often come with Service Level Agreements (SLAs). Hence, long-established members often peer with the Route Server to get an idea of which routes are available at all but only hand-pick routes they actually use based on, e.g., how consistently they are available or how much performance benefit they may introduce. As local preference values only de-prioritize (rather than filtering them) multi-lateral peering routes, Route Servers are also used as automatic fall-back in case a bi-lateral peering session suffers from, e.g., an outage [183, 412].

How Route Server specific are multi-lateral peering routes? Until now, we saw that most Route Servers have very similar characteristics; hence, we now try to understand where the actual difference lies. As a similarity metric, we use the Jaccard distance. The Jaccard distance between two sets of elements, A and B, is calculated as $JD(A,B) = \frac{|A \cap B|}{|A \cup B|}$. In comparison to other common similarity metrics (e.g., the overlap coefficient $OC(A,B) = \frac{|A \cap B|}{min(|A|,|B|)}$), the Jaccard distance also produces small values when A is entirely contained in a significantly larger B, i.e., it not only considers the similarity of elements but also the cardinalities of the sets. For each pair of Route Servers we now compute the Jaccard Distance between prefixes (see, Figure 3.8) and reachable IP addresses (see, Figure 3.9). As the Jaccard index is symmetric, we show results for IPv4 in the top-right triangle and results for IPv6 in the bottom-left triangle.

While we observe that certain Route Server combinations show more overlap than others (e.g. AMS-IX and DE-CIX), the average similarity for IPv4 lies at around 50 % (77 %

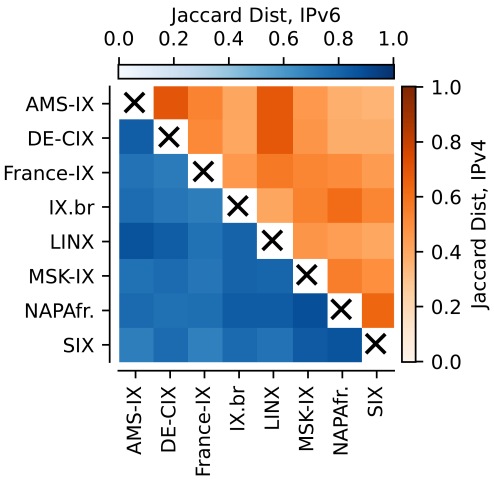


Figure 3.8: Similarity of prefixes between Route Servers

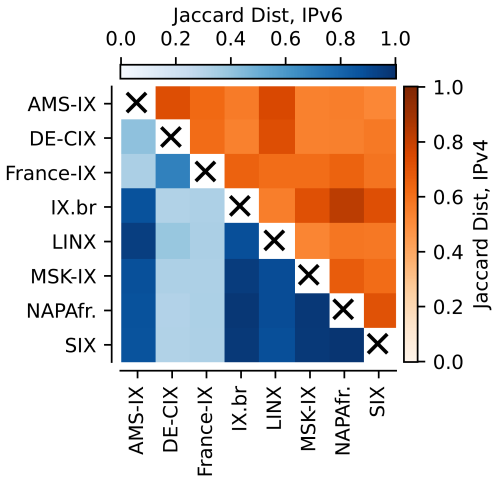


Figure 3.9: Similarity of addresses between Route Servers

for IPv6). As prefixes can be more-specifics of others, it is also unsurprising that the similarity of reachable IP addresses lies roughly 13 % higher for IPv4. While we observe similar behaviour for many IPv6 combinations, we observe that France-IX and DE-CIX are different from the others but similar to each other. We observe that this "clustering" is mainly the result of a single route: 2002::/16 announced by AS6939 (Hurrican Electric). When ignoring this route (see Figure 3.10), the takeaways for IPv6 and IPv4 are consistent.

Finally, we want to know whether ASes with memberships at multiple IXPs share the same routes with the respective Route Servers. Hence, we rerun the same analysis but, this time, focus only on routes announced by the same member ASes at both IXPs (see Figure 3.11). While this comparison shows naturally higher overlap compared to Figure 3.8, we observe that certain Route Server combinations still show a Jaccard distance of less than 70 %; yet those routes barely make a difference for the number of reachable IPs (Figure not shown).

Summary. We observe that the distribution of prefixes across Route Server peers that was presented by Richter et al. [412] is also present in many other Route Servers across the world. In general, we show that the characteristics of routes at various Route Servers are very similar. We observe that the majority of routes at Route Servers lead to out-of-continent destinations—likely a side-effect of the growing remote-peering trend. Surprisingly, we found that most routes at Route Servers contain three or more ASes and that the distance of the peer is not a factor for this phenomenon, i.e., even local peers provide many unattractive routes to the Route Server. Nevertheless, the peering LAN traffic from one IXP suggests that its members primarily use the routes to direct destinations, and mostly rely on the Route Server for failover or analysis purposes.

3.1.4 Inferring Peering Relationships

After we analyzed the routes that are available to newly joined IXP members via multi-lateral peering, we are now interested in the routes that can be obtained by establishing bi-lateral and private peering sessions.

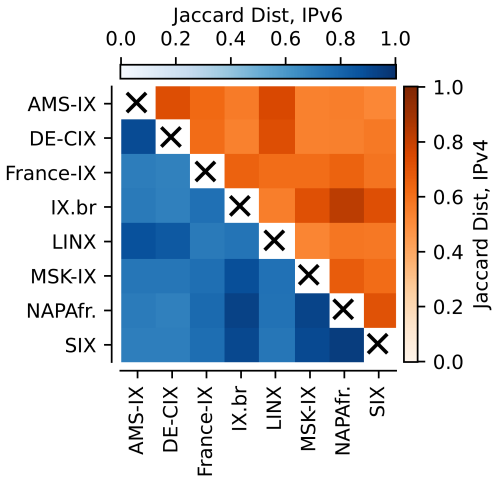


Figure 3.10: Similarity of addresses between Route Servers without HE's 2002::/16 route

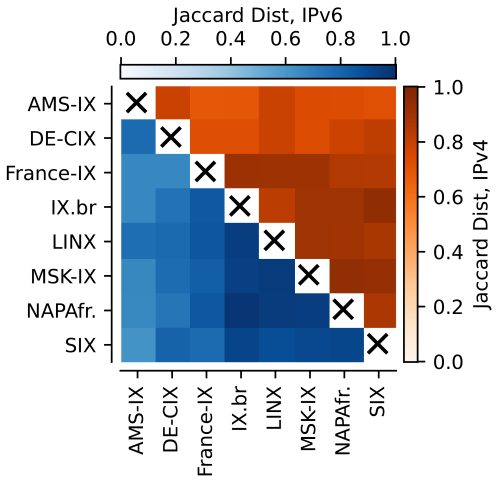


Figure 3.11: Similarity of prefixes between Route Servers for common peers

Similar to the work of Richter et al. [412], we infer bi-lateral peerings (and the prefixes that are announced via them) by observing the traffic that flows through the IXP's peering LAN. As shown by Ager et al., some ASes may "abuse" the peering LAN for additional transit connections to their customers. Given that our reachability analysis might be rather sensitive to the presence of transit relationships [8], we substantially extend the method used by Richter et al. to account for them.

As the inference approach for bi-lateral peerings relies on traffic data, we now limit the scope of our analysis to **one** large European IXP, L-IXP. While the IXP's peering LAN may cover most of the bi-lateral peering agreements, it offers no visibility into the private peerings that happen within the co-located data centers; hence, we rely on carefully selected looking glasses within those data centers to uncover routes that are available via private peering. Notably, this approach does not allow us to accurately distinguish between dedicated private peerings and connections to, e.g., cloud exchanges (as discussed in section 3.1.1).

3.1.4.1 Bilateral Peering

We bootstrap our analysis in a similar way to Richter et al. [412]: Whenever we observe traffic destined towards IP I flowing from A to B, we deduct that the respective covering /24 (or /48 for IPv6) for I must have been announced from B to A. Notably, this approach relies on the assumption that an ASes will *eventually* send traffic to most, if not all, of the prefixes it received from a neighbor. Hence, we first have to understand for how long we need to observe peering LAN traffic before we arrive at a rather static "snapshot."

Picking a reasonable window size. On the one hand, a small window size (e.g., an hour) may underestimate the available routes as not all of them continuously see traffic; on the other hand, a large time window (e.g., a year) is more likely to yield an extensive list, yet may provide an overestimate as certain routes are withdrawn in the meantime.

[8] As customers can potentially send traffic destined for the entire Internet to their transit providers, incorporating such connections would bloat up the set of reachable prefixes.

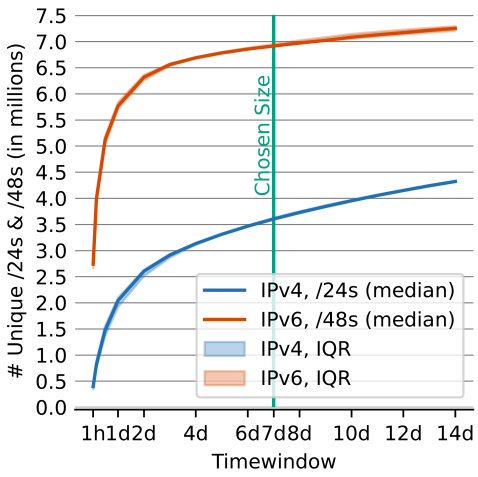


Figure 3.12: Influence of window size on visible prefixes

To get a better sense of what might be a good window size, we test by how much a certain window size would affect the number of /24s and /48s for which we observe traffic. For various window sizes between 4 hours and 14 days, we calculate the prefix counts and then move the window forward by one hour. Using this method, we generate, e.g., 739, 719, 575, and 407 data points for the window sizes 4 hours, 1 day, 7 days, and 14 days throughout the entire May 2021. Figure 3.12 show the median prefixes (y-axis) that we observed for a given window size (x-axis) as well as the Inter Quartile Ranges (IQRs) for IPv4 and IPv6. While the knee of the curve (i.e., the point at which further increases of the window size start to yield smaller improvements) lies at around one and a half days, we observe a continuous, almost linear, increase after a window size of six days. We decided to choose a window size of seven days. While this choice might yield a small number of already withdrawn prefixes, it covers workdays as well as weekend days—which are known to exhibit rather different traffic characteristics [165, 256, 279, 457].

Removing transit sessions. Now that we have some understanding of the routes that are announced between each member pair, we have to isolate and ignore transit sessions as they might substantially inflate the set of reachable prefixes. Perfectly identifying the business relationships of links has been an academic goal for more than two decades. The current state of the art algorithm, ASRank [292], is well-known for its high accuracy when it comes to identifying transit relationships (even in narrow contexts [389]). CAIDA hosts two versions of monthly-updated business relationship information: serial-1 and serial-2. While serial-1 relies solely on routing information (i.e., AS paths), serial-2 contains serial-1's information but is further extended with topology information inferred via additional sources, e.g., traceroute paths that were mapped to AS Paths. As a result, serial-2 contains more relationships but also inherits inaccuracies from its data extensions (e.g., from IP-to-AS mapping [50, 313]). Surprisingly, neither serial-1 nor serial-2 can cover more than 21.2 % or 22.3 % of the 220k+ IPv4 IXP member pairs that exchanged traffic during that period.

Improving Relationship Coverage via Route Server Paths. Whether the ASRank algorithm produces an inference for a given AS link mostly depends on the set of AS paths that it is executed on. Hence, we can improve our inference coverage by providing additional AS paths that 'cross' (i.e., contain two consecutive IXP members) the IXP's

peering fabric. To uncover such paths, we revisit the Route Server of our IXP.

Our main idea is as follows: Our Route Server snapshot contains various routes as well as their respective Route Server redistribution communities, i.e., Route Server specific communities to express the instructions: (1) announce to all neighbor, (2) don't announce to any neighbor, (3) announce to a specific neighbor, and (4) do not announce to a specific neighbor. Notably, instruction (1) and (2) are usually paired with instructions of type (3) and (4) but not with one another. By simulating the redistribution, we can deduce the paths that each IXP member received via its Route Server session(s).

More formally, we construct paths as follows: Let AS A announce some route with AS path (A, p') to the Route Server where p' refers to some (potentially empty) sequence of ASes—we ignore the few routes that contain AS_SETs. A also attaches a set of (potentially large) BGP communities that we translate into the previously explained instructions (1) -(4) . To retrieve the set RP of Route Server peers to which the route is redistributed, we first sort the set of instructions in the order we introduced them [9]. While we set RP to all Route Server neighbors for instruction (1) , we set RP to the empty set for instruction (2) ; if both instruction (1) and (2) are present we ignored the route. Notably, if n either instruction (1) nor (2) is present, we defaulted to instruction (1) . Afterward, we first added and then discarded specific ASes to/from RP according to the instructions of type (3) and (4) . respectively. Finally we constructed paths of the form $(B, A, p'), \forall B \in RP$ which 'cross' the IXP at the link (B, A).

We combine those paths with routes gathered from five days of the rib snapshots from the route collector projects RIPE RIS and RouteViews (i.e., the same data sources that CAIDA uses to produce serial-1 data). For IPv4-related inferences, we use the publicly available ASRank script that is hosted by CAIDA. For IPv6, we apply the necessary changes described by Giotsas et al. [186] to adjust the inference script to IPv6 routig policies. Both scripts require a list of Route Server ASNs for their inference. To generate this list, we extract all ASNs with the type 'Route Server' from PeeringDB. After these steps, our extended relationship data set covers 69.0 % and 63.2 % of traffic-carrying IPv4 and IPv6 links.

Improving Relationship Coverage via Manual Search. At this point, we still have various ASes with limited coverage. Hence, we decided to manually search for additional relationship information. We invested three days of manual relationship look-ups for ASes that either (i) are in the top 30 contributors of unclassified links, (ii) have only less than 10 % of their links covered, or (iii) have more than 10 % of their links inferred to be transit connections.

For our manual search, we mostly relied on entries in PeeringDB (e.g., [380]), RADb/Whois (e.g., [397]), and targeted web searches (e.g., [499]) that clearly described (at least some) relationships of a given ASN— please note that the three given examples are chosen randomly and may or may not belong to members of our studied IXP. For autnum objects in RADb/Whois, we used an approach similar to that described in [292] to infer transit relationships (even though we did not automate the process). We used as-set objects in RADb/Whois with clearly defined names (most commonly, e.g., `AS<XXX>:AS-CUSTOMER(S)`,`AS<XXX>:AS-TRANSIT(S)`,`AS<XXX>:AS-UPSTREAM(S)` or `AS<XXX>:AS-PEER(S)`) to identify relationships. For PeeringDB and the targeted web searches, we searched for exhaustive enumerations of, e.g., providers as part of, e.g., the network infrastructure description. Whenever possible, we differentiated between IPv4 and IPv6 relationships as well as regional relationships (i.e., if a websites described

[9] This order represents a conservative approach—if both the instruction to add AS X and to delete X are present, x will ultimately not be included in the set of Route Server peers.

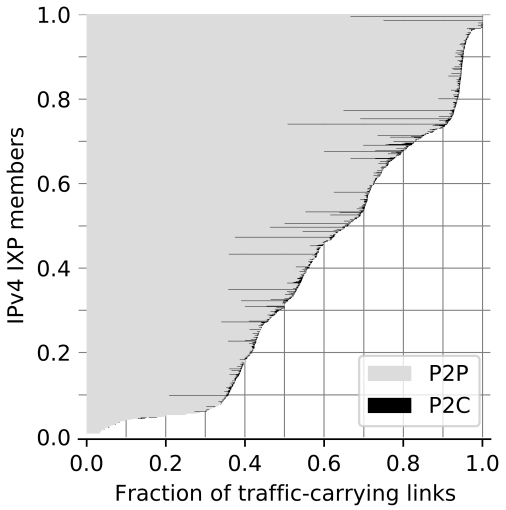


Figure 3.13: Coverage of Relationships for traffic-carrying links (IPv4).

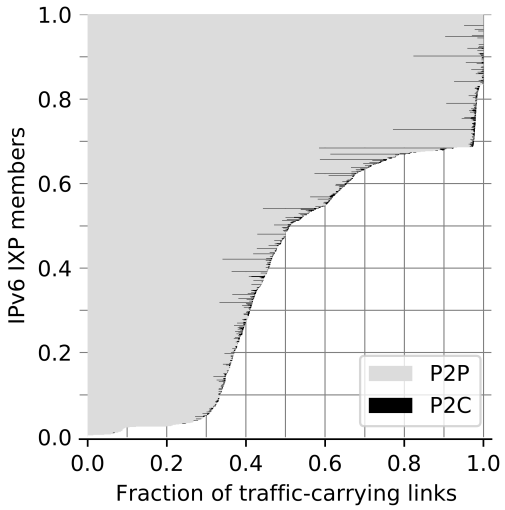


Figure 3.14: Coverage of Relationships for traffic-carrying links (IPv6).

AS X as peer in Europe but as provider in Asia, we noted it as peer giove that our IXP operates in Europe.)

While investigating the relationships for the ASes mentioned above, we observed diminishing coverage improvements; hence, we decided to not extend our manual search beyond them. Notably, whenever an AS explicitly specified its providers and customers but not its peers, we assumed that all remaining links are peering relationships.

Our final set of relationships covers 74.2 % and 65.9 % of traffic-carrying IPv4 and IPv6 links at our IXP. Figure 3.13 (for IPv4) and Figure 3.14 (for IPv6) show the fraction of links for each AS that are inferred to be P2P and P2C relationships. We observe that in both plots our data set covers at least a fourth of all relationships for 93 % of ASes. On median, we cover 66 % of IPv4 and 51 % of IPv6 relationships. While we observe that overall only 1.2 % (IPv4) and 1.5 % (IPv6) of all inferred links have transit relationships, we also observe that these relationships are distributed across almost all IXP members; hence, it is rather the norm than the exception to establish additional sessions with transit providers via the IXP's peering fabric. Beyond its coverage, we are also interested in the filtering impact of our relationship data set.

Figure 3.15 shows the number of available IPv4 and IPv6 norm-prefixes per traffic-carrying, directed[10] AS link. We observe that certain links carry traffic for more than 10^6 norm-prefixes. Yet, when only considering links that our data set classifies as peering links, we filter out all links that carry traffic for exceptionally many prefixes. Hence, we continue our analysis using only the links explicitly inferred as peering links, i.e., we not only ignore those links explicitly inferred as transit links but also those for which we have no inferred relationship.

3.1.4.2 Private Peering

As previously discussed in section 3.1.3, our traffic captures do not contain any private peering connections. We rely on queries to carefully selected Looking Glasses (LGs) to

[10] If A and B exchange traffic in both directions, we treat the links (A,B) and (B,A) separately.

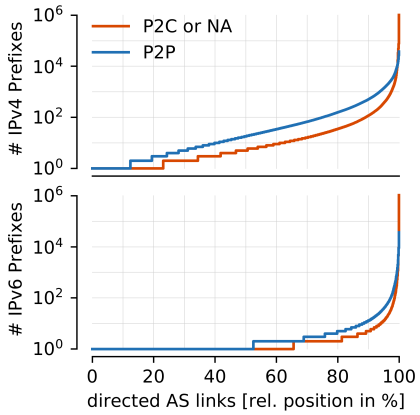


Figure 3.15: Norm-Prefixes per directed AS Link

infer routes available via private peering. To automatically query looking glass interfaces, we write identification and querying interfaces—similar to those described in [182]—for common looking glass utilities including, e.g., HSDN [461], RESPAWNER [317], and COUGAR [124]. To initially find ASes with looking glasses, we rely on PeeringDB [381] as well as various online lists [60, 63, 235, 263, 291, 482]. We first narrow down our selection by removing all LGs from ASes that are not members of our IXP. Afterwards, we removed all LGs that our indentification interface could not map to a LG template. Then, we manually went through the looking glass interfaces of the remaining 63 ASes and validated whether they could look at the routing table of a router that is located within one of the IXPs contiguous colocation facilities—we heavily relied on the naming and excluded all entries for which the location was not exactly matching a colocation name. Finally, after removing LGs requiring captchas, exploring rate-limiting, or explicitly stating 'no automation allowed', we are left with LGs from 17 different ASes to trigger.

Triggering Looking glasses. As looking glasses are usually provided on a voluntary basis from operators to operators, we do not want to abuse them with gazillions of bursty queries. First, we limit the set of norm-prefixes for which we query the LGs to those that are (1) necessary for the analysis in section 3.1.5 and (2) not yet covered by multi-lateral or bi-lateral peering. Second, when a looking glass yields a longest-prefix match rather than an exact match and returns a covering prefix that is likely not a default route (i.e., a routes less specific than /8 and /16 for IPv4 and IPv6, respectively), we no longer query for any other norm-prefixes covered by this less-specific. Third, we waited 39.3 seconds[11] on average between two consecutive queries to the same looking glass. With those safeguards in place, we queried looking glasses as follows:

1. **Querying a LG.** We choose a looking glass in round-robin fashion and performed—depending on the LG utility—either an exact match or, preferably, a longest-prefix match query against it.

2. **Ignoring transit routes.** If the LG returned a route for which the first-hop would be a transit provider to the AS the looking glass resides in, we ignore that route. Similarly, if we can't find a relationship and the first hop is a Tier 1 provider, we also ignore the route (given that it likely represents a transit relationship).

[11] a result of multiple small waits between queries to different LGs in combination with the answer time of the other LGs.

3. **Requiring IXP routes.** To ensure that the route is locally available at the IXP, we ensured that the first-hop AS is also an IXP member.

If no route remains after steps 2. and 3., we wait 2 seconds and then query the next looking glass until we have exhausted our LG list. If one LG returned a non-filtered route we marked the norm-prefix as reachable (and queried the next round-robin-order LG for the next norm-prefix), otherwise we mark it as unreachable.

In total, we were able to uncover 2.33M, 6.73M, and 6.77M IPv4 (3.41B, 3.41B, and 3.45B IPv6) norm-prefixes available via multi-lateral, bi-lateral, and private peering covering 19.8, 57.1, and 57.4 % (37.3, 37.4, 37.8 %) of all routed IPv4 (IPv6) addresses (according to Geoff Houston's Routing Table Analysis Report [218]), respectively. These results provide a real-world calibration for the 70+ % of reachability theoretically calculated by Böttger et al. [71] in 2018.

3.1.5 Route Importance

In this section, we present a qualitative analysis of the uncovered peering prefixes with two different measures of importance: (a) How many domains in a top N ranking are served by transit-free reachable prefixes, and (b) how many of the top destination prefixes of a large eyeball network are reachable without transit. The findings of this section can be summarized as follows:

- For both rankings, around half of the top-100 norm-prefixes can be reached via multi-lateral peering.
- For our traffic-based ranking, nearly all prefixes can be reached via bi-lateral peering with few exceptions that can mostly be reached via private peering.
- For our domain-based ranking, the same holds true for IPv4. For IPv6, we observe that bi-lateral peering has a substantially lower impact. While, in general, more prefixes remain unreachable than for IPv4, most of the top norm-prefixes can be obtained via private peering.
- We observe that the prefixes that remain unreachable even via private peering mostly lead to large Transit and Tier 1 providers.

3.1.5.1 Prefix Rankings

Traffic-based Ranking. To provide a traffic-based importance ranking from an independent source, we use traffic statistics from one of the largest European ISPs. In particular, we collect egress traffic from all the ISP's eyeball source addresses at all edge routers over one day (10th June 2021) at a sample rate of 1:1000 packets. For each destination IP, we sum the number of egress bytes throughout the day, aggregate these values to norm-prefixes, and cluster the top 10k norm-prefixes for IPv4 and IPv6.

Domain-based Ranking. To quantify the importance of IPs with another metric, we obtain a domain-based importance ranking. Thus, we rely on re-computed results from a previous work by Naab et al. [336]. The domain-based norm-prefix top list is generated by picking a common domain top list (e.g., from Alexa [18], Majestic [301], or Umbrella [115]), resolving these domains to as many IPs as possible, and then ranking

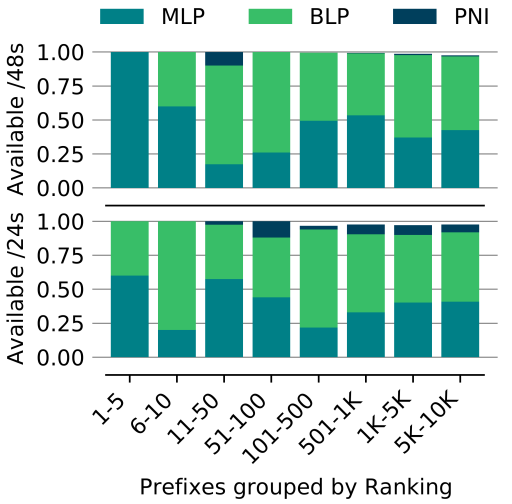


Figure 3.16: Coverage of eyeball-based top-10K prefix ranking

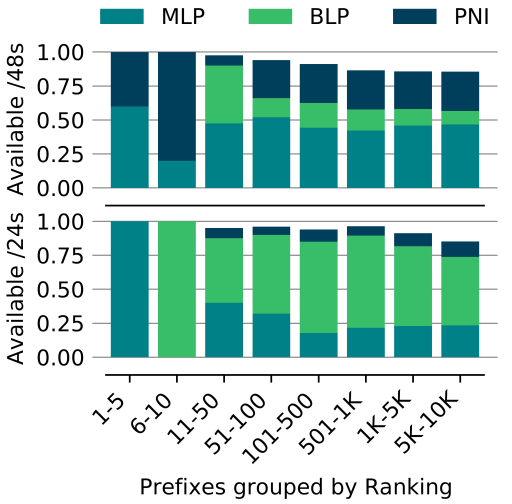


Figure 3.17: Coverage of domain-based top-10K prefix ranking

each norm-prefix by the number of Fully Qualified Domain Names (FQDNs) that can be resolved to an IP.We requested an updated snapshot of the top list from the authors of [336] and promptly received a re-computation from 30th April 2021. We decide to use the Umbrella-based norm-prefix top list because it is the only one from which we can derive 10K IPv4 as well as 10k IPv6 prefixes.

3.1.5.2 Reachability of the Top-10K

Now that we got the domain- and traffic-based top 10 IPv4 and IPv6 norm-prefixes, we can analyze how many of those prefixes are reachable via different peering types.

Traffic-based Ranking. Figure 3.16 separates the top 10k prefixes into different classes based on their respective ranking (x-axis) and shows for each class the fraction of reachable prefixes (y-axis) for IPv4 at the bottom and IPv6 at the top.

In addition, prefixes are colored by the lowest-requirement peering type (requirement and economical costs for PNI > BLP > MLP) they can be reached by (if any). We observe that the top 100 prefixes for both protocols can be fully covered using all peering types.

In general, we observe that only very few prefixes can not be reached. Notably, the vast majority of top-10k prefixes can solely be reached via bi-lateral peering agreements. This result benefits aspiring IXP members who, if they carefully select a few private peering partners, can keep their operational costs minimal.

Domain-based Ranking. Figure 3.17 shows our results for the domain-based top 10k prefixes in the same style as the previous figure. First, we observe that significantly more—especially lower rank—prefixes are unreachable (e.g., approx. 15 % of the lowest 5k IPv4 prefixes are not reachable). Second, we see a drastic shift in patterns for IPv6: The difference between routes available via multi-lateral and bi-lateral peering is almost negligible compared to IPv4. Consequently, IXP members have to rely substantially more on private peering to reach the prefixes with the highest domain counts. Yet, for approx. 15 % of 500-or-lower prefix class prefixes IXP members still have to rely on their transit as they are unreachable via peering.

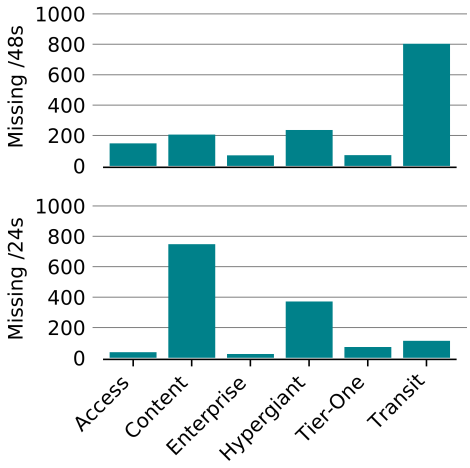


Figure 3.18: Unavailable Prefixes by Origin AS Type.

To reduce their operational costs, members of large IXPs may egress most—if not all—of their high-volume destination traffic via (mostly bi-lateral) peering connections while using their transit to egress low-volume yet domain-heavy prefixes. Notably, between 25 and 50 % of both top-10k prefix lists can be reached via multi-lateral peering—a finding that further highlights the importance of Route Server connections especially for new IXP members.

3.1.5.3 Missing Routes

To get some idea of which routes were not available, we mapped norm-prefixes to ASes via a longest-prefix match on the previously mentioned IP-to-AS data set from CAIDA. We further map each origin AS to a class using CAIDA's AS Classification data set [85].

We further refine the classification using lists of Tier 1 Networks [507] and Hypergiants [73]. Figure 3.18 shows the number of missing norm-prefixes (y-axis) that are originated by the ASes of different classes (x-axis) for IPv4 (bottom) and IPv6 (top). For IPv4, we observe that most of the missing /24 prefixes belong to content providers/hypergiants. In particular, we observe that more than half of the prefixes in both of those classes can be attributed to Amazon's AS14618 and AS16509. Notably, most of the missing prefixes for Amazon do not see any peering LAN traffic (regardless of the business relationship) throughout our measurement period.

As most of these prefixes are unique to the traffic-based prefix ranking, we suspect that our eyeball vantage point has access to routes that are only announced via private peering on dedicated connections, and, hence, remain hidden from the peering LAN. Taking Amazon out of the picture, the most prominent class would be the same as for IPv6: Transit ASes. Notably, the individual contributions made by single ASes are much more uniformly distributed; out of the 61 and 231 total ASes contributing to the IPv4 and IPv6 Transit AS class, the top ASes contribute no more than 21 and 29 prefixes respectively. Further, we observe that the vast majority of the prefixes that belong to Transit ASes are only present in the domain-based top list but not in the traffic-based top list. In summary, our observations suggest that ASes can indeed offload high-volume prefixes to peering links by joining an IXP but they still require transit to reach the heavy tail of (potentially low-traffic) domains.

3.1.5.4 Limitations

Next, we discuss limitations and specifically elaborate on the generalization of our findings. **Multi-lateral Peering:** We analyzed the Route Servers of different IXPs based on separate snapshots generated throughout seven days. Hence, our observations may be biased by sequences of high-frequency updates (as described by Ariemma et al. [36]). Yet, we discussed our results with some of the IXP operators that provided Route Server snapshots, and they told us that they did not observe unusual behavior during the days from which the snapshot was taken. Yet, as many prefixes can only be seen when aggregating updates over some amount of time, a single snapshot might miss unstable routing information. **Bi-lateral Peering:** Our analysis of bi-lateral peering reachability relied on sampled peering LAN traffic data and inferred business relationships. While we used an entire week of traffic data to partially overcome the problem of missing traffic for existing routes, we likely still missed a few routes as (1) they genuinely did not receive any traffic during our observation period or (2) they small amounts of traffic yet the sampling algorithm did not incorporate any of their packets. While we did our best to improve the coverage of inferred business relationships, we can not guarantee for the correctness of the business inference algorithm. While both algorithms were shown to provide high-quality inferences on public data [186, 292], we utilize them in a rather different context which could potentially lead to iimpairments in their performance [389]. **Private Peering:** For the inference of private peering routes, we used a very small set of looking glasses and queried them in a restrictive manner. Especially for our findings regarding the summed reachability, our observations can only be seen as a lower bound. If our number of vantage points would have been significantly higher and we could have triggered queries at a high rate, the amount of private peering prefixes would have certainly increased leading to overall higher estimates for the total achievable reachability. **Regional Importance Bias:** The utilized data sets to infer peering relations and qualify the importance of IPs and prefixes (see §3.1.4 and §3.1.5) are biased towards the European service region. While it is for the conducted analysis required to compare reachability at IXPs and relevance (ISP data set and DNS) in the very same region, it may not necessarily apply to others. As different cultures may have unique eyeball behaviors, a traffic-based ranking for other large eyeball networks around the world may lead to different prefixes especially in the lower part of the top-10k ranking. As address resolution is often location-skewed (e.g., due to DNS load balancing) our domain-based ranking is likely biased towards norm-prefixes primarily used in the European region. While we expect unmatching biases (e.g., comparing American top lists to European IXP) to lower the overall top list coverage based on, e.g., routing policy differences [188], we do not expect that such a comparison would yield considerable differences.

3.1.6 Discussion

Our results suggest that networks that peer at one of the larger IXPs can indeed move most traffic to bi-lateral peerings, yet (especially for IPv6) not all prefixes that serve a high number of domains are reachable via peering. While an assessment of the quality of those available peering relationships (i.e., the capacity and latency guarantees they provide) goes beyond the scope of this work, previous works already hinted at certain obtainable benefits[16], e.g., Schlinker et al. [446] showed that the latencies for 10 % of Facebook's traffic can be decreased by up to 10ms when switching from transit to peering routes.

That many high-volume prefixes can be served via bi-lateral peering at IXPs is strongly

correlated with the observation that Hypergiants—large content providers such as Google, Facebook, or Amazon [73]— interconnect at tens (if not hundreds) of IXPs (see PeeringDB). According to Pujol et al. [393], these relatively few Hypergiants can be responsible for up to 80 % of all ingress traffic of large eyeball networks.

Similar to hypergiants, the routes of many lower-tier networks are also available via peering. To them, broadly announcing their routes allows them to reduce the volume of ingress traffic delivered via some of their transit providers. Over time, such an approach may transform an asymmetric traffic ratio into a symmetric one, and allows these networks to re-negotiate their previous transit providers into a peering relationship.

In contrast, we observe that many of the domain-based top prefixes belong to large transit providers and Tier-1s. To reach those prefixes, IXP members often still have to rely on transit.

But how do those findings relate to different types of networks? **Large networks and hypergiants** already established thousands of peering connections [49] and use sophisticated traffic engineering strategies [447, 518] among those connections. Their egress traffic mapping is already automated to a degree where adding new peers does not pose a challenge anymore which leads to constant growth of their peering edges and continuous dwindling of dependence on their transit connections.

In contrast, **small (access) networks** may rely on a few border routers operated mostly manually by a small group of network engineers. Adding new bi-lateral peers for these networks often poses a challenge in terms of resources and network complexity (operational costs). Hence, despite our findings, many of such networks may only peer with a Route Server and a few carefully selected bi-lateral peers on purpose. To them, the reduced supplier cost that comes with sophisticated peering is often not worth the increasing added operational complexity.

Medium-sized networks (e.g., smaller national service providers) sit in between those two extremes. While many of them have neither automated their egress traffic mapping nor their peer acquisition yet, they are typically run by competent IT staff capable of anticipating how much their network would benefit from a particular peer. The earlier those networks transition from a few expensive yet feature-rich routers to a distributed fleet of cheaper routers (with potentially partial visibility), the sooner they can quickly scale their peering edge allowing them to take full advantage of the opportunities provided by large IXPs.

3.1.7 Conclusion

Throughout this section, we analyzed the routes available via multi-lateral, bi-lateral, and private peering. For multi-lateral peering, we analyzed Route Server snapshots from eight of the world's largest peering LANs and showed that most of their routes lead to out-of-continent locations via three or more AS hops. While remote peering might be a major contributor to the geographic distance of Route Server destinations, we observe that close and distant IXP members alike provide lengthy, unattractive routes to the Route Server. When comparing those findings to peering LAN traffic, obtained through a collaboration with one large IXP, we saw that mostly one-hop routes saw substantial traffic. In fact, we observed that 25 % and 77 % of IPv4 and IPv6 Route Server prefixes with at least four hop long paths see no traffic at all. This indicates that even though Route Servers provide many routes, most IXP members only make use of local routes. Afterwards, we used two heuristic-based methodologies to infer bi-lateral and private

peering routes from the IXP's peering LAN traffic. During our inferences, we carefully isolated transit connections that were established over the peering LAN—a phenomenon previously reported by Ager et al. [15]. Based on our inference, we observe that at least 19.8, 57.1, and 57.4 % (37.3, 37.4, 37.8 %) of all routed IPv4 (IPv6) address space can be reached at our IXP via multi-lateral, bi-lateral, and private peering, respectively. Those results provide practical contrast to the 70+ % reachability theoretically calculated by Böttger et al. [71]. Finally, we show that almost all of the top 10k egress prefixes of a large European eyeball network can be reached via bi-lateral peerings. In contrast, we also find that up to 15 % of top 10k domain-serving prefixes can not be reached via any type of peering at our IXP. Notably, we observe that most of these prefixes belong to large transit and Tier 1 providers.

3.2 AS-level Bias in Vantage-Point Placements

Network operators and researchers frequently use public Internet Measurement Platforms (IMPs) such as RIPE Atlas [433], RIPE RIS [349], or RouteViews [364]. They use the available measurement capabilities and publicly archived data to, e.g., detect routing events and malicious networks [171, 456, 487], analyze the Internet's structure [49, 194, 367], understand and optimize (their own) routing policies [192, 454, 478], or detect outages and performance bottlenecks [183, 458, 496].

IMPs are attractive for Internet measurements as they operate a wide array of globally distributed vantage points. RIPE Atlas hosts around 11,000 measurements probes in 3,300 autonomous systems, RIPE RIS and RouteViews collect routing information from around 300 and 500 ASes, respectively. Even though the number of vantage points is large, visibility in the more than 70,000 globally routed ASes is still partial. It is well-known that IMPs capture incomplete views of the Internet [15, 49, 188, 367, 390] and sometimes offer misleading or incomplete answers for seemingly simple questions [137, 212, 436, 510]. In fact, the incompleteness problem spawned entire branches of research focusing on extending the observed AS topology via other data sources [49, 57, 103, 162, 188] or by adding new, favorable-positioned vantage points to IMPs [121, 194, 282, 435]. While the incompleteness aspect has been extensively studied, it is still unclear how *representative* (wrt. the entire Internet) the view we have through the IMPs is. *Do we have equal visibility to all types of networks? And, if not, how biased are our views?*

In this section, we aim to shed light to this unexplored aspect of IMPs. Capturing representative sets of vantage points is an inherently multi-dimensional problem. While the trend of "hunting for the most AS links" may improve the fraction of observable Internet topology, it may bias IMPs along other dimensions, e.g., network types or geographic placement. To this end, we take first steps towards characterizing and navigating this multi-dimensional landscape of biases and improvements. Our contributions can be summarized as follows:

- We define bias in a multi-dimensional context and present a simple yet generic and easily-extendable framework to quantify the biases in IMPs (§3.2.2).
- We analyze the biases of RIPE Atlas, RIPE RIS, and RouteViews (§3.2.3). Our framework clearly confirms well-known biases, e.g., RIPE RIS is heavily biased towards larger networks and IXPs (previously stated in, e.g., [436]). Yet, it can go beyond these observations, e.g., we show that while networks that peer at many IXPs are over-represented in RIPE RIS, their peering policies are representative

of the Internet's peering ecosystem (as captured by PeeringDB). Also, we study the biases involved in common measurement practices (e.g. RIPE Atlas probes selection, or using individual route collectors).

- Leveraging our framework, we can design methodologies to reduce bias in existing IMPs (§3.2.4). Carefully selecting subsets of IMP vantage points ("subsampling") can lead to significantly more representative measurements. We demonstrate this through a use case, in which we estimate the client latency distribution of a large content delivery network (CDN) through RIPE Atlas measurements: subsampling can lead to a more accrurate estimation than randomly selecting probes, or even than using the entire set of Atlas probes.
- Another way for reducing bias is by extending the current IMP infrastructure (§3.2.5). Using RIPE RIS as an example, we calculate the bias difference each AS would introduce upon connecting to the platform. Since not every AS may be equally easy to acquire as participant in RIS, we collect data from domain experts and try to infer the acquisition complexity per AS. Our findings show that there are many easy-to-peer-with ASes that would reduce RIPE RIS bias.

We deem our work as only the first step towards understanding and mitigating bias in IMPs. There are still many aspects of bias that can be analyzed and even more problems to be studied. To this end, we publicly share our code and tools (§3.2.6) to facilitate further research, and provide a critical discussion about related work (§3.2.7), involved limitations and open research questions (§3.2.8).

3.2.1 Internet Measurement Platforms and Bias: a Primer

In this section, we introduce the concept of bias on a general example (summarized in Table 3.1). Afterwards, we introduce the three major IMPs that we analyze in this section (§3.2.1.1), discuss some of their known biases, and motivate the research questions that our study aims to address (§3.2.1.2).

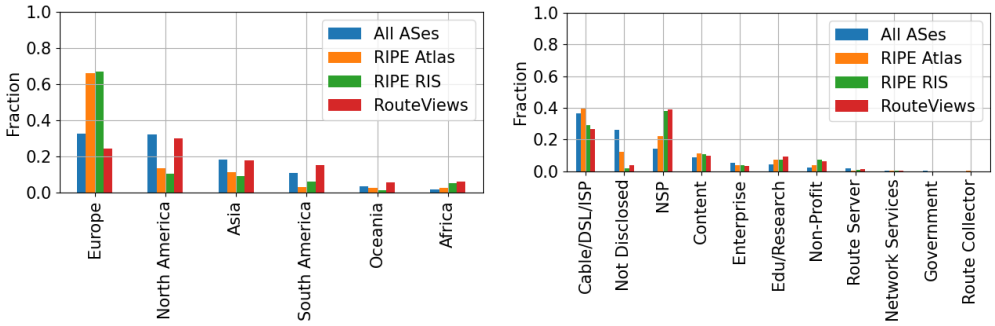
Let us assume a population consisting of 100 people, 50 of which are men and 50 women. If we run a survey with 10 people, of which 8 men and 2 women, our sample is biased towards men. We say that our sample is biased as there is a *difference in the distributions between the entire population and our sample*.

Measuring bias. To *identify* this bias, one could run statistical tests (e.g., Kolmogorov-Smirnov test) to compare the two distributions. To further *quantify* the bias, it is common to measure the *distribution distance* among the population and the sample distributions (e.g., with the Kullback-Leibler divergence metric)

	Men	Women	Country A	Country B
Entire population	50%	50%	70%	30%
Survey sample	80%	20%	80%	20%

Table 3.1: Bias example: population and sample statistics.

Multi-dimensional bias. Let us consider that our survey focuses on the height of individuals. If we compare the distributions of height within our total population to that within our survey sample, we may find that they differ as men (who naturally tend to be around ~7 % taller [374]) are over-represented. Now, let us consider that our



(a) Location (b) Network type

Figure 3.19: AS-Level location (a) and network-type (b) bias for different IMPs.

survey further focuses on the native language of individuals. For this second case, the gender-bias in our sample would not affect our findings. In contrast, the country-bias (e.g., see the right side of the Table 3.1) of our sample, may play a major role. In other words, *different bias dimensions (e.g., gender or country) may affect our measurements findings differently, depending on how they relate to the insights we want to gain.*

3.2.1.1 IMPs: RIPE Atlas, RIPE RIS, and RouteViews

We provide a brief overview of the three major IMPs on which we focus.

RIPE Atlas [433] is a platform that hosts more than 11,000 measurement "probes" in more than 3,000 ASes. Probes support a fixed set of measurement types (e.g., ping, traceroute, DNS). Users can select sets of probes and execute measurements (e.g., a traceroute towards a target IP), under some rate-limits.

RIPE RIS [349] and RouteViews [364] are two global platforms that host "route collectors", which are dedicated devices that passively receive, dump, and publicly archive the routing information from their peering networks. Most route collectors are located at large IXPs such that they can quickly establish many sessions over the IXP's peering LAN. The "multi-hop"-enabled route collectors may establish indirect sessions with remote ASNs. In total, RIPE RIS and RouteViews host 27 (of which 3 multi-hop) and 36 (20 multi-hop) route collectors with more than 500 and 300 peer ASNs, respectively. A peering ASN may provide feeds for the entire routing table ("full feed") or only a part of it.

3.2.1.2 Known IMP Biases & User Awareness

Location bias. A glance at the map with the locations of RIPE's infrastructure (see [347] for Atlas probes and [3] for RIS route collectors) reveals a higher density of the infrastructure in Europe, which is in imbalance with the spread of ASes around the world, i.e., there is location bias in RIPE Atlas and RIPE RIS. On the contrary, as it can be seen in Fig. 3.19(a), the RouteViews project has route collectors deployed in more representative locations around the world.

Topological bias. Route collectors (RIPE RIS and RouteViews) are biased towards larger core networks and at Internet eXchange Points (IXPs) [436].

Bias awareness. Neither the location-based nor the topological bias are new to expert users. However, not even expert-users might be able to accurately judge the extend of different biases on different IMPs, e.g., while RIS and Atlas have substantial location bias, this bias is almost negligible for RouteViews. Similarly, other biases along (less prominent) dimensions, such as the network type (see Fig. 3.19(b)), might be even harder to judge. A questionnaire related to the topic of this section that we ran supports the fact that not all users are aware of biases: out of the 50 questioned operators and researchers, only 26 (52%) consider IMPs to be biased, while 28% consider that there is no bias (or, probably not), and 20% "do not know". *This lack of (or, partial) awareness motivates our study to comprehensively quantify the bias in IMPs.*

3.2.2 Data and Methodology

Similarly to the example of §3.2.1 where people are characterized by two features (gender and origin country), the IMPs can also be characterized by a multitude of features, such as, location, connectivity, traffic levels, etc.. Each characteristic/feature can be considered as a dimension, and the bias can be calculated over each dimension. Then, depending on the measurement use case, all of some of the dimensions can be taken into account, depending on their relevance (see §3.2.1).

In this section, we first formally define the bias and the metrics to quantify it (§3.2.2.1). Then, we present the data we use to retrieve characteristics for the IMPs (i.e., "bias dimensions") and the taxonomy we use in the section (§3.2.2.2).

3.2.2.1 Quantifying Bias: Definition and Metrics

Definitions: Let P be the distribution of a characteristic (e.g., network size) within a set of networks $\mathcal{N}$. If the characteristic takes K distinct values, its distribution is $P = [p_1, ..., p_K]$, where p_i is the probability of a network having the i value (e.g., p_{Europe}=0.32 for the entire population of ASes; see Fig. 3.19(a)); formally, $p_i = \frac{1}{|\mathcal{N}|} \sum_{j \in \mathcal{N}} I_{j \to i}$, where $I_{j \to i}$ an indicator function that is 1 if the network j has the characteristic i, and $|\mathcal{N}|$ the size of the set $\mathcal{N}$.

Also, let a subset of networks $\mathcal{V} \subset \mathcal{N}$, and Q be the corresponding distribution within the set of networks $\mathcal{V}$. We define the bias (of the set $\mathcal{V}$ wrt. to the set $\mathcal{N}$) as the distance between the distributions P and Q.

Identifying bias: If the distance between P and Q is statistically significant, then there is bias. There are several statistical tests that could be applied. We use the Kolmogorov-Smirnov (or, KS-test), which is a nonparametric test that compares two distributions (two-sample KS-test), and answers "what is the probability that P and Q are drawn from the same distribution?".

Bias Metrics: There exist several metrics to quantify the distance between two distributions. A common metric is the Kullback–Leibler (KL) divergence:

$$B_{KL} = \sum_{i=1}^{K} p_i \cdot \log\left(\frac{p_i}{q_i}\right) \tag{3.1}$$

The KL-divergence takes values in $[0, +\infty]$, where the higher the value the more the two distributions differ. In the work, we use a bounded version of the KL-divergence that

takes values in $[0, 1]$ [440, 477][12], and we call it the *bias score*.

For example, in terms of the location distributions depicted in Fig. 3.19(a), the bias score for RIPE Atlas and RIPE RIS is $B_{KL} = 0.06$ and $B_{KL} = 0.07$, respectively, while for RouteViews, which follows a similar distribution to the entire population, the bias score is $B_{KL} = 0.01$. For the network type (Fig. 3.19(b)) the bias scores for RIPE Atlas, RIPE RIS, and RouteViews are 0.03, 0.12, and 0.09, respectively, clearly highlighting the higher bias in the route collector projects.

Remark: We tested other common metrics (e.g,. Total Variation) for the bias score as well, and while the actual values of each metric are different, the qualitative findings of this section remain the same.

3.2.2.2 Bias Dimensions: Data and Categories

Data sources We focus on the characteristics of IMPs at an AS-level (e.g., two RIPE Atlas probes in the same AS have the same AS-level characteristics). Yet, our methodology and analyses are extensible and applicable to a more fine-grained level (e.g., per monitoring device, such as at a vantage point level or router level; see also the discussions in §3.2.6 and §3.2.8). We compile a list of characteristics for each ASes from the following data sources: CAIDA's AS-rank [88] and AS-relationships [87] datasets, PeeringDB [90, 381], Internet Health Report (AS-hegemony) [172, 231], and bgp.tools [65].

Remark: Our choice for AS-level granularity is twofold: data availability and scope. Specifically, at the AS-level there are several public datasets, while at at a finer granularity there is scarce information; this would limit the generality of our analysis in terms of bias dimensions. For a finer granularity, one would need to conduct measurements and analyses to collect or infer the needed data, which can be very useful for several use cases, but is out of the scope of this section that aims to give a first comprehensive characterization of bias in the IMPs. We deem it as the beginning of a research thread, and discuss more about its limitations and potential future directions in §3.2.8.

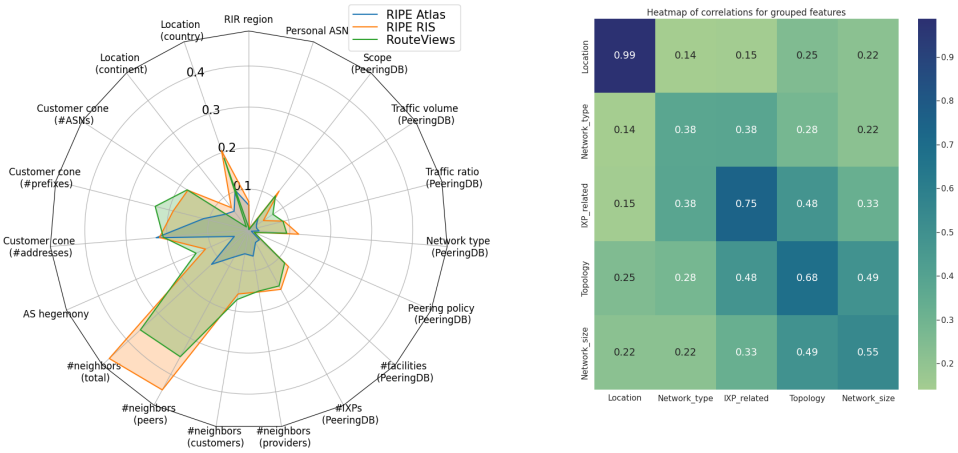
"Vantage Points (VPs)": Since we study bias at an AS-level, in the remainder, we will not differentiate between different probes in RIPE Atlas that are hosted in the same AS, or between different peers of RIPE RIS and RouteViews with the same ASN. And, for brevity, we will refer to the ASes that host RIPE Atlas probes or provide feed to RIPE RIS / RouteViews as "vantage points" or VPs.

Dimension categories. From the datasets we select a number of characteristics that are more relevant to the concept of bias and group them in the categories:

- ***Location***: RIR region; Country; Continent
- ***Network size***: Customer cone (#ASNs,#prefixes,#addresses); AS hegemony
- ***Topology***: #neighbors (total, peers, customers, providers)
- ***Interconnection (IXP-related)***: #IXPs; #facilities; Peering policy
- ***Network type***: Net. type; Traffic ratio; Traffic volume; Scope; Personal ASN

Remark: Our methodology is generic and more characteristics can be included or grouped differently. We only use this taxonomy to facilitate the discussion throughout the section (i.e., to refer to multiple dimensions under a single term). It does not affect any of the results, which we present in detail for all dimensions.

[12]We substitute $q_i \to (1-w) \cdot q_i + w \cdot p_i$, with $w = 0.01$ and normalize with its upper bound $\log \frac{1}{w}$, to get $B_{KL} = \frac{1}{\log \frac{1}{w}} \cdot \sum_{i \in \mathcal{K}} p_i \cdot \log\left(\frac{p_i}{(1-w)\cdot q_i + w \cdot p_i}\right)$.



(a) Bias in IMPs

(b) Correlation between dimensions

Figure 3.20: Radar plot showing the multi-dimensional bias in IMPs.

3.2.3 Analyzing IMP Bias

In this section, we study the biases in RIPE Atlas, RIPE RIS, and RouteViews. Figure 3.20(a) shows a radar plot with bias scores for all dimensions. The colored lines—and their included area—correspond to the bias metric of a given IMP along a given dimension, e.g., the bias score for RIPE RIS (orange line) in the dimension "Location (country)" is 0.2. Larger bias scores (i.e., farther from the center) correspond to more bias, e.g., in the dimension "Location (country)" RIPE RIS is more biased than RIPE Atlas (blue line).

Remark: As knowing the overall distributions of a characteristic may help to better understand the bias along a certain dimension, we provide detailed distribution plots (i.e., similar to those in Fig. 3.19) for all characteristics in Appendix A.2. Based on Figure 3.20(a), we make the following observations:

- While the bias of IMPs differs significantly by dimension, RIPE Atlas is substantially less biased than RIPE RIS and RouteViews along most dimensions.
- RIPE RIS and RouteViews have significant topological bias (e.g., number of neighbors/peers) as most of their collectors are deployed IXPs, where ASes establish many (peering) connections [390].
- RouteViews and RIPE RIS are also quite biased in terms of network size ("Customer cone" dimensions), since route collectors peer with many large ISPs. While having feeds from large ISPs may be desired in terms of visibility, users still should be aware of it since it may lead to biased measurements.
- In most IXP-related and network type dimensions (that correspond to data mainly from PeeringDB), all platforms have relatively low bias; with an exception of RIPE RIS and RouteViews that are biased in terms of number of IXPs/facilities the monitors are connected to.
- There are small differences between RIPE RIS and RouteViews. RIPE RIS is more biased in terms of topology (number of neighbors, total and peers), whereas Route-

Views is more biased in terms of network sizes ("Customer cone" and "AS hegemony" dimensions).

- We applied the KS-test for all platforms and dimensions. In almost all cases, the KS-test rejected the null hypothesis that the IMPs vantage points follow the same distribution as the entire population of ASes. The only exceptions were the "Personal ASN" dimension for all IMPs, and the "RIR region" and "Location (continent)" for RouteViews (where bias scores are less than 0.01).

Figure 3.20(b) shows the correlations between dimensions (grouped as in §3.2.2.2; values correspond to averages among groups) for the entire population of ASes. As expected, dimensions in the same category are correlated. Also, topology dimensions are significantly correlated with network size and IXP-related dimensions. Nevertheless, comparing with Fig. 3.20(a), we can see that correlated dimensions do not necessarily share similar bias scores. This highlights that a multi-dimensional bias exploration as in Fig. 3.20(a) can give a more detailed view.

Beyond this basic analysis, we conduct three similar analyses deepening our understanding of different IMP aspects.

Combining RIS and RouteViews: Using data from both RIPE RIS and RouteViews is common (e.g., via CAIDA BGPStream [371]); hence, we analyze the combined bias in Fig. 3.21(a). When considering vantage points from both projects, the bias slightly decreases in most dimensions. Interestingly, there are some exceptions, e.g., number of neighbors (total and peers), where it would be preferable—in terms of bias—to use only feeds from RouteViews.

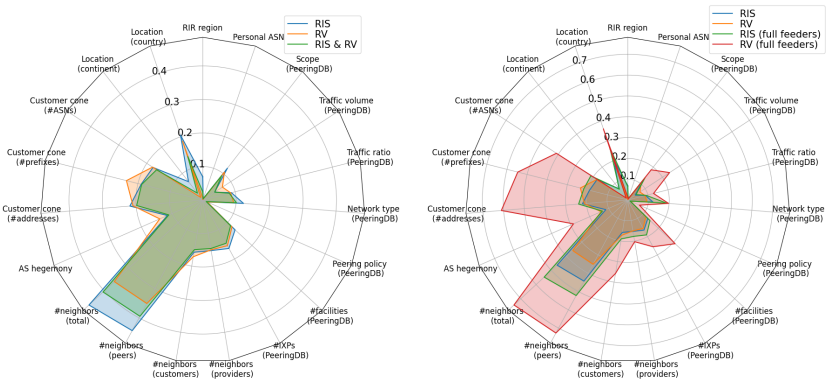
Full vs. all feeds: Only 240 and 70 peers of the RIPE RIS and RouteViews peers provide feeds for the entire routing table ("full feeds"), respectively. Figure 3.21(b) compares the bias of only feed peers against the entire IMPs. For RIPE RIS the increase in bias is small, whereas for RouteViews the set of full feeds is significantly more biased. In fact, while RIPE RIS is on average more biased than RouteViews, the opposite becomes true when considering only full feeds.

IPv4 vs IPv6 vantage points: Figure 3.21(c) compares the set of ASes hosting IPv4, IPv6, and any RIPE Atlas probes. The set of networks hosting IPv6 probes is slightly more biased than networks hosting IPv4 probes in most dimensions. The only exception is the #addresses in customer cone, which is mainly due to the differences in the IP space between the two versions. In RIPE RIS (not depicted in the plot), the differences between IPv4 and IPv6 peers is negligible.

3.2.3.1 Analyzing Improvement Potential

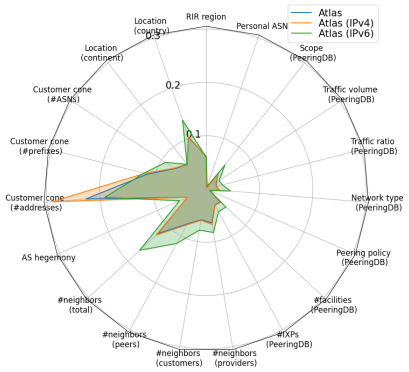
Now that we have a basic understanding of the current biases in IMPs, we want to compare the current state to an (hypothetical) case, where vantage points are randomly deployed among all types of networks, locations, etc. This comparison (i) provides a better understanding of the potentially avoidable IMP bias, and consequently (ii) reveals room for improvement (under practical limitations).

Random sampling from the entire population is an unbiased process. A sufficiently large random sample would lead to zero bias. Yet, small samples tend to be biased especially for characteristics with large variance. We treat the bias score that can be achieved via random sampling as a non-biased baseline.



(a) RIS & RV

(b) Full feeds



(c) Atlas: IPv4 vs IPv6

Figure 3.21: Radar plot showing the multi-dimensional bias in IMP variations.

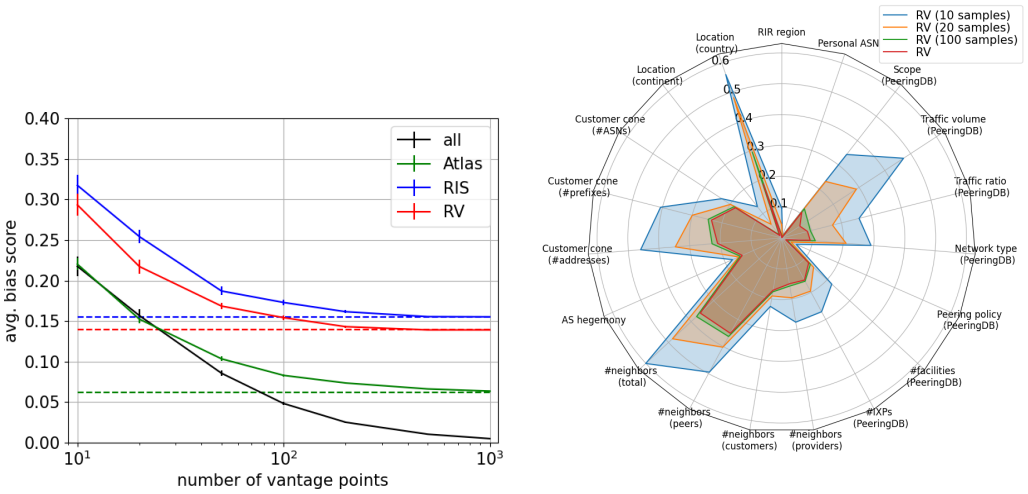
Table 3.2 compares the average bias over all dimensions[13] of the IMPs against that of a random sample with the same number of vantage points (e.g., in the case of RIPE RIS we consider random samples of size $|\mathcal{V}|$=539). We repeat our random sampling 100 times and report the average bias. We observe that *with the same number of VPs as in the current IMPs, a random sample of ASes would have on average (almost) no bias*. This indicates that

Platform (#vantage points)	Atlas (3391)	RIS (539)	RV (340)	RIS & RV (762)
Platform bias	0.06	0.16	0.15	0.14
Random sample bias	0.00	0.01	0.01	0.01

Table 3.2: Bias of IMPs vs. random sample of vantage points.

the "limited" number of vantage points is not the root cause of bias (which is mostly due to the deployment strategies; see §3.2.1.2) . In fact, we show later that adding few well-chosen VPs can drastically reduce the overall bias (§3.2.5.2) and that very low-biased IMP subsets can be selected via subsampling (§3.2.4).

[13]There infinite options of combining bias scores of different dimensions. In this section, we consider



(a) (b)

Figure 3.22: Effect of IMP subsampling on (a) average and (b) multi-dimensional bias.

Bias vs. number of vantage points While the current set of VPs is clearly not optimal in terms of bias, we wonder how bias changes when we only use a smaller random set of VPs (e.g., measurements with few Atlas probes due to rate/credit limits, or collecting feeds from a subset of route collectors peers due to the large volumes of data [19]).

Figure 3.22(a) shows the average bias for different sample sizes drawn randomly from either the entire population of ASes ('all') or one of the three IMPs. Lines correspond to averages over 100 sampling iterations, and errorbars indicate 95% confidence intervals. For ease of comparison, dashed lines correspond to the bias values of using the entire infrastructure (i.e., the values in Table 3.2). We observe that: (1) the bias decreases with the sample size (as expected), (2) random sampling has always lower bias (for the same number of VPs), and (3) even for very small sample sizes ($\geq$20 VPs), random sampling has lower bias than the *entire* sets of RIPE RIS and RouteViews VPs (see dashed lines), while the same holds for RIPE Atlas for $\geq$40 VPs.

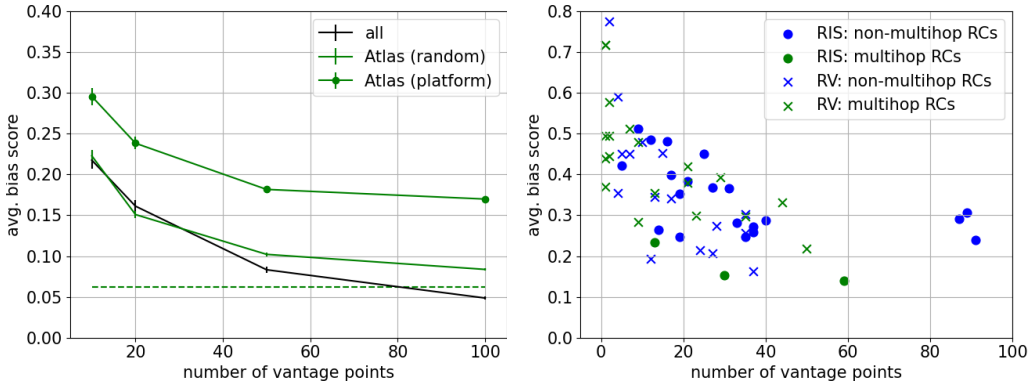
For a deeper inspection of the bias in smaller sets of VPs, Fig. 3.22(b) presents the bias of random samples of RouteViews VPs of sizes 10, 20, and 100 (similar results hold also for RIPE RIS and Atlas). We can see how the bias decreases in all dimensions for larger subset sizes. Yet, the bias does not decrease linearly, e.g., in network type dimensions the relative increase in bias for small subsets is much larger than in topology dimensions.

3.2.3.2 Bias in Common Measurement Practices

In this section, we briefly analyze the bias involved in common VP selection methods that users follow in practice.

RIPE Atlas probe selection algorithm. RIPE Atlas users can either select specific probes to use in their measurements or not specify them (which is is the default choice; with

averaging as an intuitive choice, and discuss other options in §3.2.8.



(a) Atlas probe selection

(b) Bias vs. #monitors per RC

Figure 3.23: Effect of subsampling on average bias for (a) atlas probes and (b) route collectors.

parameters 10 probes from "worldwide locations"[14]). In the latter case, RIPE Atlas has an automated algorithm to assign probes to a measurements, which prioritises probes with less load over more loaded probes, which makes the probe selection procedure not equivalent to true random sampling.

In Fig. 3.23(a) we study how the RIPE Atlas selection algorithm, "Atlas (platform)", performs compared to random sampling from either all RIPE Atlas probes, "Atlas (random)", or from all ASes ("all"); the values for these latter cases are the same as in Fig. 3.22(a)). We considered the sets of probes that the RIPE Atlas platform returned when we initiated measurements with parameters `type="area"` and `value="WW"`. Lines correspond to averages over 100 sampling iterations, and errorbars indicate 95% confidence intervals. We observe that *when using the RIPE Atlas algorithm for selecting probes, "Atlas (platform)", then the bias is significantly higher compared to selecting randomly probes, "Atlas (random)"*. In fact, the bias is almost two times higher. This indicates that even with the existing infrastructure, users could decrease bias by 50% by not depending on the built-in probe selection process, but select random probes themselves.

Feeds from a single Route Collector (RC) may be used in cases that there are processing limitations (e.g., in terms of real-timeness or storage) due to the large volume of data, see [19, 36, 193]. Figure 3.23(b) presents the average bias score per RC (i.e., the bias of the set of VPs that peer to a RC) in relation to its number of VPs. Overall, there is a clear (negative) correlation between the number of VPs and the bias score of a RC. Nevertheless, the size of a route collector does not predict its bias as (1) the three RCs of RIPE RIS (rrc01, rrc03, rrc12) that are significantly larger (>80 members) than the rest of RCs, are not less biased (in fact, there are several smaller RCs with lower bias) and (2) there are several medium-size RCs (and even some with only 10-20 monitors) that have relatively low bias. For RIPE RIS, the three multihop RCs (rrc00, rrc24, rrc25) are less biased than most of the non-multihop RCs (which are deployed at IXPs).

Summary of main takeaways: (1) Our framework can easily detect bias and finds that RIPE RIS and RouteViews are substantially more biased than RIPE Atlas; (2) if IMPs would choose VPs entirely random, their current set of VPs would be very close to an ideal sample; and (3) common practices to limit the number of VPs yield higher bias than simple random samples from IMPs.

[14] `https://atlas.ripe.net/docs/udm/#probe-selection`

3.2.4 Reducing Bias via Sub-sampling

The bias of the IMPs is due to the fact that some types of networks are under-represented and others over-represented (e.g., networks in Asia and Europe, respectively, for RIPE Atlas; see Fig. 3.19(a)). To decrease the bias, we need to have a balanced representation of all network types. To this end, we can either (i) deploy new VPs to the under-represented network types or (ii) use only a subset of the existing monitors whose types are over-represented. In this section, we study the latter option, which is immediately applicable and does not involve any extra costs. We study aspects of extending the IMPs in §3.2.5.

3.2.4.1 Sub-sampling: Problem and Algorithms

We consider the problem of selecting a subset of VPs in the IMPs (subsampling), whose bias is lower than the bias of the entire set of VPs; or, more formally:

$$\min_{\mathcal{S}\subseteq\mathcal{V}} B(\mathcal{S}) \tag{3.2}$$

where $\mathcal{S}$ is a subset of the VPs of a IMP, $\mathcal{S} \subset \mathcal{V}$, and $B(\mathcal{S})$ is its bias score, which we want to be less than the bias of the entire set of VPs in the IMP ($B(\mathcal{S}) < B(\mathcal{V})$). Optionally, we may impose constraints, e.g., to require a given size of the subset ($|\mathcal{S}| \leq k$).

Problem complexity. The above problem is combinatorial, and relates to the best subset problem with cardinality constraints, which is NP-hard [59]. Moreover, it is straightforward to show (through counter-examples) that it lacks the properties of *monotonicity* and *submodularity*, which would allow to design approximation algorithms with performance guarantees [266]. Informally, the complexity of the problem is due to the fact that adding a VP in a set could either decrease or increase bias (no monotonicity). And, since the bias score is calculated over distributions of characteristics for the entire set of selected VPs (see Equation 3.1), having two VPs can be equally, less, or more than the sum of effects of adding each VP individually (no submodularity).

Greedy algorithm. We design a heuristic algorithm to select a subset of VPs with low bias (Algorithm 1). The algorithm starts by considering the entire set of VPs in the IMP $\mathcal{V}$. Then, for every VP $v \in \mathcal{V}$, it calculates the resulting bias score if we remove this VP from the IMP, i.e., the bias score of the set[15] $\mathcal{V}\backslash\{v\}$ (see lines 2–4). And, it removes from the set $\mathcal{V}$ the VP v that would decrease the most the bias score, i.e., $\arg\min_v B(\mathcal{V}\backslash\{v\})$ (lines 5–6). It repeats the above process for the updated set $\mathcal{V}$ by removing one VP at each iteration, until the remaining set is of the desired size k.

The complexity of the algorithm is $O\left((|\mathcal{V}|-k)\cdot|\mathcal{V}|^2\right)$, or $\approx O\left(|\mathcal{V}|^3\right)$ for small k, since the loop in lines 1–7 is executed $|\mathcal{V}|-k$ times, and the loop in lines 2–4 is executed $|\mathcal{V}|-i$ times at the i^{th} iteration (i.e., complexity $O(|\mathcal{V}|)$, and the calculation of bias in line 3 requires the calculation of the distribution P (probabilities p_i in Equation 3.1) which is of complexity $O(|\mathcal{V}|)$.

Simple sorting algorithm. We also study the performance of a simpler algorithm (Algorithm 2), which initially calculates for every VP the resulting bias score if the VP is removed from the set $\mathcal{V}$, and then it removes (without further calculations) the $|\mathcal{V}|-k$ vantage points that correspond to the lowest bias scores. The complexity of the algorithm is $O\left(|\mathcal{V}|^2\right)$, which is significantly lower than the greedy. However, it is expected to perform worse than Algorithm 1, since it does not take into account the combined effects of removing multiple VPs.

[15] $A\backslash B$ denotes all elements in a set A, except for those that belong also to a set B.

Algorithm 1: **Greedy**

Input: $\mathcal{N}$ (population), $\mathcal{V}$ (vantage points), k (sample size)

while $|\mathcal{V}| > k$ **do**
  for $v \in \mathcal{V}$ **do**
    $B_v \leftarrow calc\_bias(\mathcal{N}, \mathcal{V} \backslash \{v\})$
  end for
  $v \leftarrow \arg\min B$
  $\mathcal{V} \leftarrow \mathcal{V} \backslash \{v\}$
end while
Return $\mathcal{V}$

Algorithm 2: **Sorting**

Input: $\mathcal{N}$ (population), $\mathcal{V}$ (vantage points), k (sample size)

for $v \in \mathcal{V}$ **do**
  $B_v \leftarrow calc\_bias(\mathcal{N}, \mathcal{V} \backslash \{v\})$
end for
while $|\mathcal{V}| > k$ **do**
  $v \leftarrow \arg\min B$
  $\mathcal{V} \leftarrow \mathcal{V} \backslash \{v\}$
end while
Return $\mathcal{V}$

Figure 3.24: Subsampling algorithms

3.2.4.2 Sub-sampling Efficiency

Figure 3.25(a) shows the bias score (y-axis) for subsets of the IMPs selected by the greedy Algorithm 1 (continuous lines) or the simple sorting Algorithm 2 (dashed lines) for varying set sizes k (x-axis). The bias score values at the rightmost part of the curves correspond to the bias score of the entire set of VPs $\mathcal{V}$. As we move on the left of the x-axis, the subsampling algorithms remove VPs from $\mathcal{V}$. In all cases, the subsampling with the greedy algorithm has a similar behavior: the bias score decreases as we remove VPs (i.e., moving to the left of the x-axis) to a minimum point, and then it increases again as the subset sizes become small. The minimum is achieved at samples sizes k that are one order of magnitude less than the size of the entire IMP $|\mathcal{V}|$ (note the log scale of the x-axis).

RIPE RIS and RouteViews: the sweet spot of 50 peers. RIPE RIS and RouteViews have a similar behavior, with sample sizes of around 50 VPs achieving the lowest bias score values of less than 0.04, which is *four times lower than the bias score of the entire VPs sets*. Comparing the curves with this of random sampling in Fig. 3.22(a), we see that for sample sizes $k < 100$ VPs, the subsampling algorithm can select sets that have lower bias than what a random sample (of the same size) from the entire population of networks has.

RIPE Atlas: almost zero bias with a few hundreds of probes. In the case of RIPE Atlas the bias score is less than 0.01 (i.e., six times lower than the entire set of Atlas VPs) for subsets of a few hundreds of VPs ($84 \leq k \leq 976$). The minimum is achieved for around 300 VPs, and the greedy algorithm performs better than random sampling for samples smaller than ∼1,000 VPs.

Greedy vs. Sorting. Finally, we compare the performance of the greedy algorithm (continuous lines) against the simple sorting algorithm (dashed lines). For large subset sizes ($k > 200$ for RIPE RIS and RouteViews, and $k > 2000$ for Atlas) the performance of both algorithms is similar. However, as the subset sizes decrease, the sorting algorithm performs worse, since it does not take into account in its decisions the combined effect of removing more than one VPs.

And, while for RIPE RIS and RouteViews the minimum achieved bias score by the sorting algorithm (0.05 and 0.04, respectively) is not far from the corresponding of the greedy algorithm (25%–30% higher), in the case of RIPE Atlas the sorting algorithm achieves at best a 3 times higher bias score and for a much larger subset size k (more than 1000

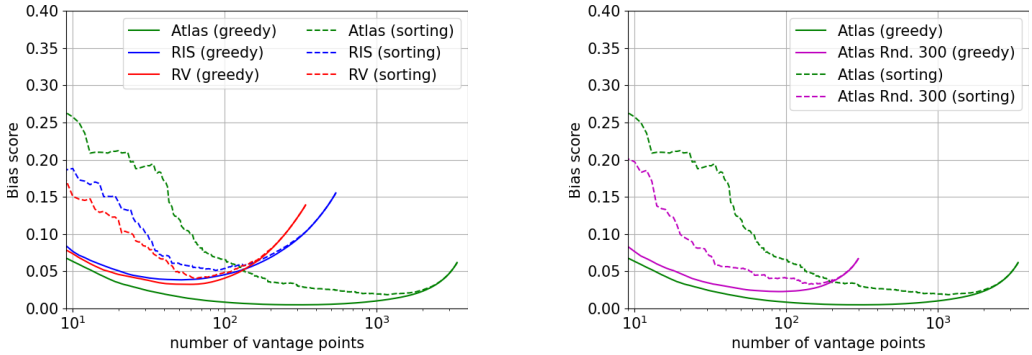


Figure 3.25: Bias scores for greedily and optimally chosen subsets of RIPE Atlas, RIPE RIS, and Routeviews (a) with a comparison to Atlas' random choice algorithm (b).

VPs). Nevertheless, the computation complexity of the sorting algorithm is much lower[16]. Hence, there is a trade-off between computational complexity and performance, allowing for a need-based algorithm choice (e.g., real-timeness or resources).

Remark: The subsampling algorithms can be applied to any set $\mathcal{V}$. For example, Fig. 3.25(b) shows the results for a scenario where the algorithms received as input a random set of 300 RIPE Atlas VPs ("Atlas Rnd. 300"; purple lines). The greedy algorithm efficiently decreases the bias score from a value of 0.07 (for the entire random sample of 300 VPs) to a value of 0.02 (achieved at the subset of 90 VPs). This is worse than applying the subsampling on the entire set of Atlas VPs (green lines), however, it comes at a much lower computational complexity.

3.2.4.3 Sub-sampling in the Wild: a Use Case

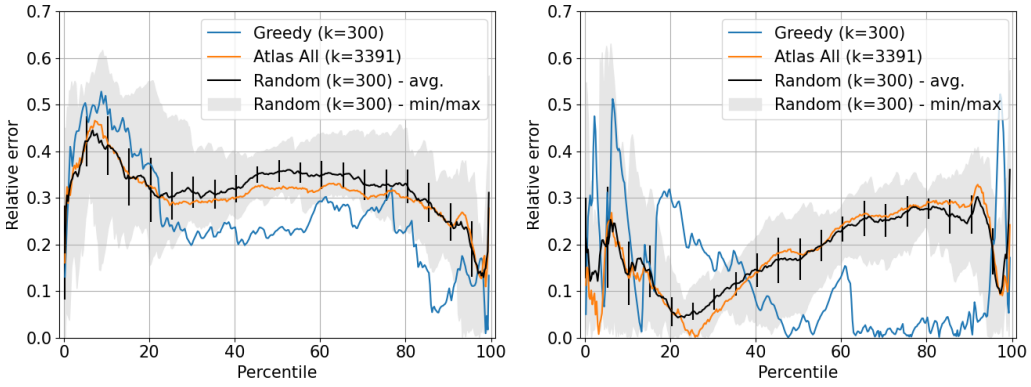
In this section we demonstrate through a use case how subsampling can improve measurement methodologies.

Use case: estimation of latency distribution. We consider a measurement scenario where the goal is to estimate the distribution of the latency between a network (e.g., a content distribution network, CDN) and its client networks. A network may already have a custom measurement system to monitor the latency between all its customers. However, in some cases having a system like this may not be possible, and we may need to rely on IMPs (e.g., ping measurements). For example, we may want to test a routing configuration that does not serve traffic, or evaluate the expected performance of a non-existing deployment [322].

More specifically, let ℓ_i be the latency between the network and a client network i, and L be the distribution of the latency values ℓ_i for all client networks $i \in \mathcal{N}$. Also, let us consider measurements from a subset S of the infrastructure, $S \subseteq \mathcal{V}$, and L_S the distribution of the latency values for the networks $i \in S$. We want to investigate whether sets S of lower bias results to a distribution L_S that better approximates the ground-truth distribution L.[17]

[16] In our experiments the greedy algorithm needed significantly more time, e.g., several minutes for RIPE RIS and RouteViews, and hours for RIPE Atlas.

[17] Note that our goal is simply to investigate the role of the set S in the accuracy of a simple estimator



(a) RIPE Atlas measurements

(b) CDN measurements

Figure 3.26: Relative error between the actual and measured RTT distribution for different IMP subsets.

Experiment setup. We conduct our experiment from a large production CDN with multiple points of presence and global coverage.

- *Groundtruth latency data:* We collect client latency data in the form of Roud-Trip-Times (RTTs) for all client ASes from the CDN monitoring infrastructure using TCP socket estimates, similar to [446]. For each client AS, we collect multiple latency samples (from connections to different hosts in the same AS) and calculate the median[18] latency value ℓ_i per AS i.
- *RIPE Atlas measurements:* We conduct ping measurements from all RIPE Atlas probes to the same global CDN infrastructure as measured in the previous step, and collect the RTT values. For ASes hosting more than one probes, we calculate the median latency, which we denote as $\ell_i^{'}$ for an AS i.

We calculate the latency distribution of the ground-truth data L from the CDN measurements over all ASes, i.e., $\{\ell_i : \forall i \in \mathcal{N}\}$. We want to investigate how accurately we can estimate L with the RIPE Atlas platform. Hence, we consider the following subsets S of the RIPE Atlas platform:

- *Atlas All:* $S = \mathcal{V}$, i.e., all ASes that host at least one RIPE Atlas probe.
- *Greedy (k=300):* The set S selected by Algorithm 1 for size set $k = 300$, i.e., the subset with the minimum bias score (see §3.2.4.2 and Fig. 3.25(a)).
- *Random (k=300):* A set S consisting of a random sample of 300 Atlas probes. For statistical significance, we consider 10 different sets.

For each set S, we calculate two latency distributions: The *actual measurement distribution* $L_S^{'}$, which consists of the RTT values from the RIPE Atlas measurements $\ell_i^{'}$, $i \in S$, and the *simulated measurement distribution*, L_S, which consists of the values of the CDN measurements ℓ_i, $i \in S$. The reason for this additional consideration is to isolate any noise due to methodological differences between RIPE Atlas and CDN measurements per AS, i.e., differences between ℓ_i and $\ell_i^{'}$.

Results. Figure 3.26 presents the relative error (y-axis) between the measured latency

L_S. Designing a fine-tuned methodology for accurately estimating the latency distribution from a set of measurements is out of the scope of this section.

[18]Considering other values (e.g., averages or minimum) did not affect our findings.

Measurements	Greedy (k=300)	Atlas All	Random (k=300)
RIPE Atlas	26%	31%	32%
CDN	12%	18%	18%

Table 3.3: Average relative error (over all percentiles) in latency measurements.

distribution $L_S^{'}$ and the groundtruth distribution L for all percentiles (x-axis). For example, if the 30^{th} percentile of L_S is t sec. ($L_S(30) = t$) and the corresponding percentile for L is $L(30) = 0.8 \cdot t$ sec., then the relative error is $\frac{|L_S(30)-L(30)|}{L(30)} = \frac{|t-0.8 \cdot t|}{0.8 \cdot t} = 0.25$.

We can see that the relative error for the subset selected by the greedy algorithm (blue lines) is lower than the relative error of the entire set of Atlas VPs (orange lines) and the average relative error of the randomly selected sets (black lines) in the majority of the percentiles. For low percentiles (e.g., less than 20%) the subsets selected by the greedy show a higher relative error, however, due to the fact that these percentiles correspond to latency values of a few milliseconds, small differences in the measured latencies lead to high relative errors (see the higher variability of the random results indicated by the gray area). Taking into account the accuracy over the entire distribution (i.e., average accuracy over all the percentiles), the subsampling achieves almost one third lower errors than using the entire set of Atlas probes or random sets of them (see Table 3.3).

The same trends holds both for the RIPE Atlas measurements (Fig. 3.26(a)) and the CDN measurements (Fig. 3.26(b)); in the latter, the errors are lower due to the isolation of the "noise" from the different measurement methodologies.

These results indicate that subsampling for sets of lower bias can improve measurement methodologies. Further refinements in the subsampling per use case, e.g., identifying the bias dimensions that relate to latency measurements and taking only them into account, could lead to higher accuracy (see §3.2.8).

Summary of main takeaways: (1) We extended our framework with subsampling methods that allow users to choose low-bias subsets of the VPs of an IMP; (2) While choosing sets of ~50 vantage point from RIPE RIS and RouteViews can decrease the bias 3.25-fold, greedily choosing a set of ~300 RIPE Atlas vantage points can almost entirely eliminate bias; (3) A first in-the-wild test of our subsampling methods showed that its selection of RIPE Atlas probes can improve the prediction of the global latency distribution for a large anycast CDN.

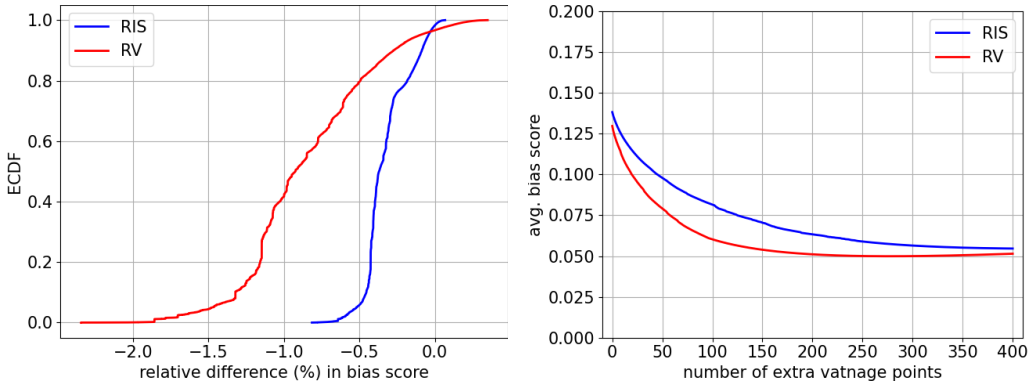
3.2.5 Extending the Platforms

The bias of IMPs can be decreased by adding new participants from under-represented dimensions. In this section, we take a closer look at the effectiveness (§3.2.5.1) and complexity (§3.2.5.2) of adding vantage points.

3.2.5.1 Selecting New Vantage Points

As our framework can calculate bias for arbitrary sets of VPs, we can calculate the difference in bias that a given VP v would introduce by adding it to the existing set of VPs of a platform $\mathcal{V}$, i.e.,

$$B(\mathcal{V} \cup \{v\}) \leftarrow calculate\_bias(\mathcal{N}, \mathcal{V} \cup \{v\}) \tag{3.3}$$



(a) (b)

Figure 3.27: Relative difference in bias score when extenting RIPE RIS and RouteViews.

This problem shares the same hardness and complexity properties as the subsampling problem we discussed in §3.2.4.1. Hence, we again may rely on either a sorting or a greedy strategy: The algorithm sorts VPs based on how they affect bias (Eq. 3.3) and either (1) selects the k VPs that decrease the bias score the most (sorting algorithm) or (2) greedily chooses the v that minimizes the IMP bias (i.e., $v \leftarrow \arg\min B(\mathcal{V} \cup \{v\})$) in each iteration and adds it to the set of all VPs $\mathcal{V}$. Despite its similarity to the subsampling variant, this greedy algorithm has a much higher complexity of $O\left(|\mathcal{N}| \cdot |\mathcal{V}|^2\right)$. Since the algorithm has to investigate $\mathcal{N} \backslash \mathcal{V}$ vantage points in each iteration and $\mathcal{N}$ is multiple orders of magnitude larger than $\mathcal{V}$, the algorithm effectively runs in $\gg O\left(|\mathcal{V}|^3\right)$.

Extending RIPE RIS and RouteViews. For RIPE RIS and RouteViews, we calculate the bias difference for every[19] AS not in IMPs (using Eq. 3.3), and present the distribution of these differences in Fig. 3.27(a). While few ASes would further increase the bias (i.e., values larger than 0.0), the vast majority of ASes would decrease the bias (i.e., negative values); this shows that there are many good options for extending the route collector projects. We further observe that certain ASes are especially beneficial as they would decrease the bias score by more than 2% for RouteViews or almost 1% for RIPE RIS.

Yet, substantially decreasing the bias of a platform requires greater efforts than connecting a single AS. Using the sorting algorithm, Fig. 3.27(b) shows how the bias for these platforms reduces with increasing amounts of new ASes. The bias in both RIPE RIS and RouteViews can decrease to a value of 0.05 with a few hundreds of extra VPs (which is close to the minimum value achieved by subsampling; see Fig. 3.25(a)), while RouteViews would need less extra VPs compared to RIPE RIS to reach a certain bias score value.

3.2.5.2 Acquisition Complexity of ASes

While certain ASes could substantially reduce the overall bias, not all of them might be equally eager to join a project. Some of them might have security, privacy, or communication policies that entirely impede connecting them. In contrast, some ASes (e.g., personal-use ASes) often join projects within hours of first contact. In the following, we study the complexity/cost of adding an AS as a VP, and how this relates to its impact

[19] In this analysis we omit stub ASes, i.e., edge networks having only one neighbor AS and connecting to the Internet through this single upstream provider.

on the bias score. This acquisition complexity can play a crucial role in practice, and it should not be overlooked when considering vantage point placement.

The complexity for peering with an AS depends on various aspects (operational, legal, etc.). And, to our best knowledge, this dependence has not been publicly reported or studied. Since peering coordinators of route collector projects may manually explore those aspects, we (i) ran a targeted questionnaire among 4 experts involved in 3 route collector projects to record existing domain knowledge, and (ii) build a model to infer the peering complexity for all ASes.

Approach overview. First, we compile a set of labels, based on various datasets, that characterize an AS (with each AS potentially having multiple labels). Then, we asked the peering coordinators to assess—based on their experience—the peering complexity associated with each label. Finally, since each AS is associated with some labels, and each label is mapped to a complexity score, we infer a complexity score for every AS.

Characterizing ASes. ASDB [528] combines a multitude of databases and classifiers to associate a set of labels to an AS (e.g., "ISP", "Research and Education", or "Government and Public Administration"). The ASDB dataset characterizes more than 100K ASes with one or more labels (out of 86 potential labels[20]). Moreover, we extend the ASDB data with the following labels:

Community-support: ASes that are involved in IMPs (by April 1st, 2022), by (i) hosting an active RIPE Atlas probe, or (ii) peering with a RIPE RIS or RouteViews collector, or (iii) participating in the NLNOG Ring project [414].

Education: ASNs for which the PeeringDB [381] `info_type` field in the `net` record is `Educational/Research` [90], or which are operated by a national research and education network, NRENs (we manually mapped the organization names from Wikipedia [508] to ASNs using PeeringDB, ipinfo.io [234], bgpview.io [66], bgp.tools [64], or their websites).

Isolario-peer: ASes that previously peered with the Isolario route collector project [244] that terminated service on Dec. 31st, 2021. While the MRT data files are no longer publicly available, we extracted the list of peer ASes for each Isolario route collector on June 21st, 2021.

Personal-use: A list of ASes that belong to (mostly) individuals (e.g., as hobby projects, or for testing or research) [65].

Point-of-contact: We extract from PeeringDB [90], all ASNs whose `role` field in the `poc` record is either set to `Maintainance`, `NOC`, `Policy`, or `Technical` as we expect other roles, e.g., `Abuse`, to not handle peering requests.

State-owned: ASes that are owned (at least partially) by states, based on the list published by Carisimo et al. [93].

Questionnaire & Complexity Scores. We asked the participants to judge, based on their prior experiences, how much each characteristic/label may influence the ease of peering on a scale from -3 (prevents peering) to +3 (easily peers upon request). Table 3.4 summarizes the answers. While we observe the +3 option twice, we only observed the -3 option once. In general, all participants tended to agree in their answers (with an exception for the "Point-of-contact" label).

For each label, we take the mean value across all questionnaire answers. For each AS, we

[20] ASDB assigns primary and secondary labels. For simplicity, we counted combinations of these two label classes.

Label	min	mean	max
Community Groups & Nonprofits	±0	+1	+2
Computer & Info. Techn.	±0	+0.25	+1
Edu. & Research	±0	+0.25	+1
Finance & Insurance	-1	-0.25	±0
Gov. & Public Admin.			
→ Military, Security, ...	-3	-0.75	±0
→ Other 2nd labels	-1	-0.25	±0
Services			
→ Law, Business, ...	-1	-0.25	±0
→ Other 2nd labels	±0	±0	±0

Label	min	mean	max
Community support	±0	+0.75	+2
Education	+1	+1.75	+2
Isolario-peer	±0	+1.75	+3
Personal-use	±0	+2	+3
Point-of-contact	-2	+0.75	+3
State-owned	-2	-1	±0

Table 3.4: Summary of questionnaire answers.

merge the different labels as follows: if there is a -3 or +3 label, we set the complexity score to -3 or +3 (preferring -3 over +3), respectively, otherwise we take the mean across all labels; this approach ensures that our final values always comply with the strong indications provided by the experts. Finally, we divide this score by 3 to normalize it.

Complexity scores for current and future ASes. Figure 3.28(a) (black curve, "All") shows the distribution of acquisition complexity scores across all ASes: approximately 80 % of ASes are equally hard to peer with, while the remaining 20 % split almost equally between ASes that are substantially harder/easier to peer with. Comparing with the distributions of ASNs that are part of IMPs, we can see that all platforms are biased towards easier-to-connect ASes; this trend is stronger for RIPE Atlas as only ~17 % of probe hosting ASes have equal or lower complexity scores than the median AS (i.e., ~0.1).

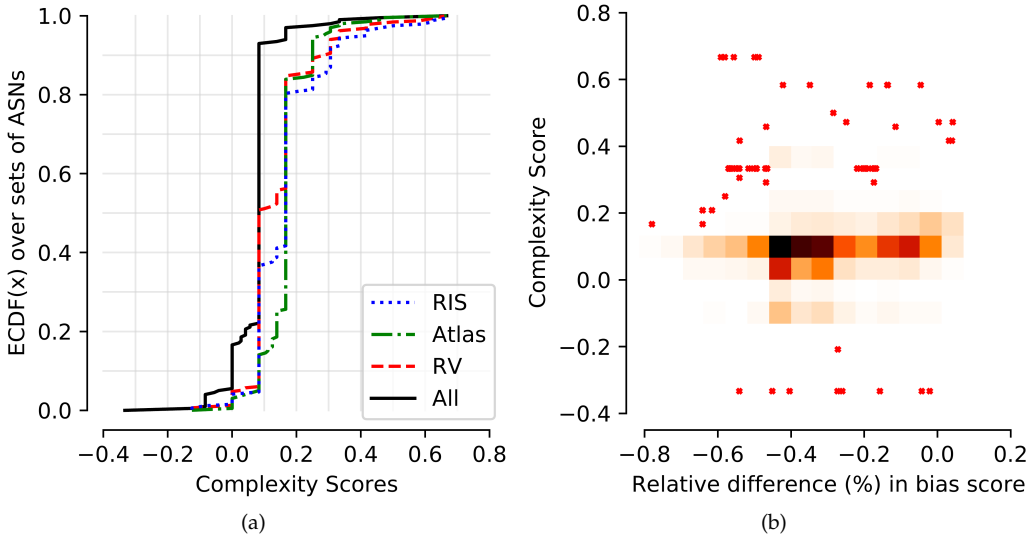
Does this mean that decreasing the bias can only be achieved by convincing hard-to-connect ASes? Figure 3.28(b) shows a heatmap that compares these two dimensions for the RIPE RIS platform. We can observe that there is no strong correlation, i.e., there is no pattern indicating that networks that could decrease bias are harder to connect to. Moreover, there are several ASNs that are ideal candidates to extend RIPE RIS (top left part of the heatmap), since they could significantly decrease bias and are easy to connect. For example, AS132139 —a cloud on-ramp provider[21] with three upstreams that mainly operates in Hong Kong, is registered under APNIC, and announces few IPv4 and IPv6 prefixes—would be a great choice for extending RIPE RIS as it has labels that indicate a low peering complexity (0.66) and also reduces the bias drastically (-0.59%).

Summary of main takeaways: (1) Adding extra a few hundreds VPs to IMPs can decrease their bias by 50%; (2) While ~10 % of ASes are easy to acquire as new VPs, another ~10 % of ASes need substantial effort to be connected; (3) There is a very clear set of extension candidates that can easily be connected and substantially reduces bias.

3.2.6 Open Data, Code, and API

To facilitate users and further research and analyses, we provide data, code, and tools to calculate and visualize the bias in a set of networks.[22]

[21] An AS that mainly provides transit to all major cloud providers / hypergiants.
[22] To ensure anonymity, a link to these resources will be provided at a later stage.



(a) (b)

Figure 3.28: Relation between acquisition complexity and bias scores.

Data. The data we aggregated from different sources are provided as a table with rows corresponding to ASNs and columns to network characteristics (see §3.2.2.2).

Code. We open-source the code for calculating the bias. The method receives as input (i) the data table, (ii) a set of ASNs that are considered the "population" $\mathcal{N}$, (iii) a subset $\mathcal{V}$ of the population, whose bias we are interested in, (iv) the set of characteristics that will be taken into account.

Visualizations. We provide visualizations of the bias data, namely, radar plots (as in Fig. 3.20(a)) and the detailed distributions per characteristic (CDF plots or histograms, as in Appendix A.2) for all platforms.

Open API. To further facilitate access to data and methods, we provide an API that receives a set of ASNs and returns their bias score per dimension.

3.2.7 Related Work

Topological bias of route collectors. When analyzing the Internet's topology, route collectors often miss many interconnections of CDNs [49], at IXPs [15, 188, 390], or due to complex routing setups [367]. While it is hard to remove these biases, many works tried to understand the importance of certain biases for their work by analyzing how their results would change when using only subsets of the available infrastructure, e.g., [251, 292, 312, 455].

Biases and usecases. While Roughan et al. argued in 2010 that route collectors are biased towards larger core networks and IXPs [436]. Chung et al. [108] saw no substantial differences when comparing their view on the longitudinal deployment of route origin validation with that of Akamai gathered from an order of magnitude more monitors[23].

[23]The study only analyzed prefix-origin pairs that were visible by the route collectors. It remains unclear whether this result would change when also considering Akamai's privately received BGP announcements.

This highlights that biases might be use-case dependant—a fact further supported by the work of Cittadini et al. from 2014 which showed that route collectors have different biases for topology analysis and iBGP policy inference [121]. In 2009, Heidemann and Papdopoulos argued that Internet measurements, in general, are biased by various (sometimes unknown) factors such as traffic volume, user populations, or topology [206], which is further supported by a serious of exemplary experiments conducted by Bush et al. [82].

Bias in RIPE Atlas. In 2015, Bajpai et al. showed that the distribution of Atlas probes to ASes is heavy-tailed and also analyzed the network type distribution of probe hosting ASes (without comparing it to the overall type distribution) [52]. A later study by Bajpai et al. in 2017 further found that 91 % of RIPE Atlas probes are located in the RIPE and ARIN region and that the number of probes is not representative for the number of Internet users in countries such as Japan [53].

Extending current platforms. In 2008, Roughan et al. model the topology discovery problem as an extension of the simple capture-recapture model that is frequently used in biological research to estimate the population of a species via K random yet comparable samples. They argue that their model can also be used to estimate beneficial route collector peers and they estimate that significant global link coverage can be reached with fewer than 700 peers [435].

In 2012, Gregori et al. observed that customer ASNs see most—if not all—of their provider's routing information. Based on this observation they introduced a metric named p2c-distance that counts the number of transit relationships an update has to travel between two ASNs. Their approach defines the optimal route collector set as the minimal number of ASes needed such that each AS in the entire Internet has at most a p2c-distance of N (they practically used 2) to at least one feeder ASN. They solve a slightly modified minimum set cover problem to compute this set of ASNs and find that multi-homed stub ASNs are most valuable as new collector feeds [194].

In 2022, Leyba et al. analyzed AS topology from a probabilistic standpoint. They accumulated different observations over several time periods, and assigned to each link a probability of existence. Their model show the most uncertainty for links in Israel, Egypt, Georgia, Bulgaria, and Iceland, suggesting that connecting ASNs in these countries can be beneficial for topology discovery [282].

Subsampling IMPs. Two very recent studies [19, 34] considered subsampling of RIPE Atlas and RIPE RIS vantage points, respectively. [34] calculates a similarity matrix between Atlas probes based on measurements, and proposes a method to select subsets of probes that are dissimilar. Similarly, [19] calculates VPs similarities based on topological characteristics, and applies a clustering algorithm to select a set of dissimilar of VPs aiming to achieve a good tradeoof between volume of information (i.e., less VPs) and observability of the AS topology.

3.2.8 Conclusion

This work aims to be the first effort for a systematic and comprehensive characterization of bias in IMPs, by providing a framework to quantify bias (metrics, data, code, etc.) and an analysis of popular IMPs. Being aware about the existence of bias and its "flavors" (e.g., how much and at what dimensions) can help the users of IMPs to carefully interpret the results of their measurements, and avoid pitfalls or wrong generalizations that may appear due to the bias.

Moreover, our findings and tools (data, code, API) can further help users to fine-tune their measurements (e.g., select a set of vantage points), and provide useful insights to IMP operators for extending their platforms. We see several promising messages in our results towards these directions.

We deem our work as an initial (but, necessary) step towards a complete understanding of bias in IMPs and its impact on user measurements. There are many research directions and improvements that would need a more extensive investigation and can be addressed in future work. In the following we provide a critical discussion for some of these directions, in relation to our work:

AS-level granularity. We conducted our analysis at an AS-level, because the majority of data sources provide data at this granularity. It is straightforward to generalize our framework to a more fine-grained level (all methods, metrics, etc., directly apply). For example, if we have available data per prefix[24], then we can consider as our "population" all the routed prefixes, and as "sample population" the prefixes that contain the IP addresses of the RIPE RIS / RouteViews peers or the RIPE Atlas probes. Our methods would then simply take as input a matrix with rows the prefixes (instead of the ASNs) and columns the prefix characteristics (instead of the AS characteristics).

Several use cases could benefit from such a more fine-grained granularity. However, the challenging part is the data availability. To extract even a single characteristic at this granularity, we may need extensive measurements and analyses. For example, a custom method is needed to infer per-prefix locations [511], while to infer customer cones per-prefix could lead to incomplete data since aggregating measurements from VPs in different prefixes would not be possible.

Dimensions of bias (per use case). Not all dimensions of bias may be relevant to a measurement study. For example, any bias in the "peering policy" dimension may not affect latency measurements (this is just a conjecture), whereas it is probable to affect BGP hijacking detection measurements. Identifying which dimensions are important per use case, could improve our understanding of bias and its role. However, this requires a per case analysis, since there are many different measurement use cases with a wide range of scopes and objectives.

Our framework considers bias per dimension, allowing for various use cases. Specifically, while in our work we decrease the *average* bias score along all dimensions (§3.2.4 and §3.2.5), one can define different aggregation ways (i.e., other than averaging). More formally, if $B_i(\mathcal{S})$ is the bias score of a set $\mathcal{S}$ along a dimension i, then the overall bias score can be any function f of the individual scores: $B(\mathcal{S}) = f\left(B_1(\mathcal{S}), B_2(\mathcal{S}), ..., B_K(\mathcal{S})\right)$. Some examples could be: (i) a weighted average, where the importance of each dimension would be captured by a weight w_i (which can be tuned per use case), or (ii) the "stricter" case of $\max_i B_i(\mathcal{S})$ that captures the "worst case" of bias, or (iii) $B(\mathcal{S}) = 1 - \prod_{i=1,2,...,K}\left(1 - B_i(\mathcal{S})\right)$ that aims to achieve a "balance" among all dimensions.

Impact of bias. In the CDN use case, bias is responsible for a 6% in the latency estimation error (see randomly selected vs. low-bias sets of Atlas probes; Table 3.3). Another study [455], has shown that estimating the impact of a hijack with RIPE RIS leads to a 10% higher error than custom measurements to random ASes. Knowing the impact of bias in a use case can help us build new methods (or even inform us to not focus on it, if the impact is small). However, as in our above discussion, this would need a per case analysis.

[24]Some ASes consist of many—sometimes globally distributed—routers that make independent decisions, which can be captured at a prefix level.

Accuracy, completeness, and bias in ground truth data. The input to our framework (i.e, the AS characteristics) is from public datasets. And, some of them are known to suffer from inaccuracies (e.g., country information per ASNs), incompleteness (e.g., only 25% of ASNs have records in PeeringDB), or even biases (e.g., data inferred based on measurements from the existing –biased– platforms, such as, customer cones, topology, etc.). Improving the datasets would be beneficial, in general, and for the quantification of bias, in particular, since they could reveal further insights[25]; nevertheless, this is an orthogonal task.

Generalization of our framework: beyond IMPs. The population (i.e., set $\mathcal{N}$) does not necessarily be the entire population of ASes; depending on the use case, it can be the set of clients of a network, or a set of networks with a given characteristic (e.g., all ASes in a continent, or non-stub ASes), etc. Similarly, the selection of the subset $\mathcal{V}$ may not be limited to IMPs; e.g., it can any arbitrary set of networks that can be measured. In this way, our framework can be used to quantify biases in setups other than IMPs; one just needs to change the input sets ($\mathcal{N}$ and $\mathcal{V}$) in our methods, while the data remain the same.

3.2.9 Ethics

We carefully designed this study to protect user privacy and ethical research.

Questionnaires. Throughout this study, we circulated two *anonymous* questionnaires. We do not collect any personal data, and we follow the basic principles of ethical research (Respect for Persons, Beneficence, Justice, etc.), and the guidelines for collecting anonymous data through online surveys (informed consent, comprehension, voluntariness, returning benefits, minimization of risks, etc.). To minimize the risk of identifying individuals, we will neither publish the raw or preprocessed versions of individual answers.

AS characteristics. (i) The complexity score we inferred for each AS indicates the chances that an AS peers with route collectors, and (ii) the bias score is calculated for a set of networks and indicates how representative this set is wrt. the entire population of ASes; none score characterizes in any other way an AS.

CDN measurements. In our use case (§3.2.4.3), we collaborated with a large CDN with strict policies to retrieve anonymized performance statistics. The individual data points presented in this section were entirely produced on servers at the CDN's premise, and do not carry any per-individual or per-network information.

3.3 AS Business Relationships and Evaluation Bias

The Internet consists of many autonomous systems that exchange reachability information (also known as routes). Which routes are made available to a neighbor often depends on business relationships. While actual business relationships are rather complex [180, 185], we often categorize them into three different types: (1) Provider-to-Customer (P2C), (2) settlement-free Peer-to-Peer (P2P), and (3) relationships between ASes that belong to the same organization called Sibling-to-Sibling (S2S).

[25]The main insights of this section are not expected to deviate significantly, since we have not identified any counterintuitive findings in our analysis

Many researchers rely on accurate relationship information for (1) simulations of routing incidents [319, 331, 456], (2) IP-to-AS mapping [220, 313], or (3) network (resource) management [268, 492]. Yet, there is no organisation or entity that can provide authoritative knowledge for those relationships. Over the last two decades, this lead to a large corpus of research focusing on inferring relationships from, e.g., routing information [167, 175, 185, 186, 248, 251, 292].

Yet, there are two major problems that those inferences suffer from: (1) limited visibility into the Internet's AS interconnection graph and (2) lack of ground-truth validation data. The **visibility problem** is a well-known challenge in Internet topology research [15, 99, 195, 365]. While various partial solutions have been proposed (e.g., using data plane information [49, 103, 162], routing policy databases [57], or BGP community encodings at IXP route servers [188]), it is still a challenge to generate a comprehensive AS-level typology that also captures, e.g., private network interconnections [512].

The **lack of ground-truth validation data** has been pointed out as a challenge many times (e.g., [175, 292, 483]), yet recently proposed and evaluated algorithms (see, [248, 251]) rely entirely on "best-effort" validation data compiled from BGP communities—a technique initially introduced and used (among others) by Luckie et al. [292].
To better understand the implications of this trend, this section focuses on the basic question: *How good is our "best-effort" validation (data)?* In particular, our work makes the following contributions towards answering this question:

- **Bias Analysis.** We analyze to which degree the geographical and topological biases within the sets of inferred and validated relationships match (§3.3.4). We uncover significant mismatches: While the "best-effort" validation data covers 31 % of all links between ASes in the ARIN region, it only covers less than 1 % of links in the LACNIC region. Yet, both regions contain roughly 15% of the inferred relationships.
- **Implication analysis.** We analyze how such bias mismatches may affect classification correctness for three (ASRank [292], ProbLink [248], and TopoScope [251]) classification algorithms[26] and uncover substantial drops in precision for certain groups of peering links (§3.3.5). In particular, we observe that the near-perfect precision of 96-98 % for the entire validation data set drops by 14-25 % (depending on the algorithm) for peering relationships between Tier-1 and transit providers.
- **Future outlook**: We discuss, in-depth, different approaches for compiling less biased and more complete validation data sets (§3.3.6) and highlight (1) the need for active discourse with operators and (2) how the routing ecosystem's continuous change can be exploited to over-sample validation data.

To allow for the reproduction of our results and to facilitate the analysis of future validation efforts, we make our research code publicly available via:

`https://gitlab.mpi-klsb.mpg.de/lprehn/imc2021_breval`

3.3.1 Why Should We Care about Bias?

Biases commonly arise in all forms of classifications—whether one looks at face detection [81], patient treatment [385], or criminal behavior [369]. While those disciplines

[26]While we would have also analyzed UNARI [167], the authors do not provide publicly available artifacts.

may have stronger social impacts, the correctness of business relationships may have far-reaching and unintended consequences when studying the Internet's routing ecosystem. For instance, Müller et al. [334] recently proposed an algorithm that relies on the inferred relationships between Internet Exchange Point (IXP) members to identify spoofed packets (i.e., packets with a forged source address). The misclassification of a P2C as a P2P relationship could potentially result in many packets being falsely flagged as spoofed. If an IXP would publicly disclose, e.g., the number of spoofed packets per member, the reputation of certain members could sustain damage.

Yet, how did bias affect this example? IXPs are often built with the intention to keep local traffic local [15], i.e., they connect ASes within the same **geographical region**.[27] As most geographical regions have their own operator meetings, conferences, and communities—e.g., RIPE [425], NANOG [337], or APRICOT [35]—that release different recommendations on how to operate certain types of networks, the best practices for routing can differ among regions (and IXPs). For instance, Marcos et al. [312] recently reported that the usage patterns for AS path-prepending (a commonly used traffic engineering technique) vary strongly by region and over time. Similarly, **topological biases** can arise from how ASes of different sizes or locations within the Internet's hierarchy select their peering policies [288].

In summary, features such as the geographical or topological positioning of a network can greatly influence the routing decisions taken by its operators. This may become important when relationships are explicitly or implicitly[28] used in narrow contexts, e.g., only between members of an IXP. In such a case, the correctness estimates that were obtained from a potentially larger base of relationships may provide a false sense of safety which may result in economical consequences (as in the example above).

3.3.2 Background

In this section, we first give a brief introduction to selected[29] relationship inference algorithms, then provide details on previously used techniques for obtaining validation data, and finally summarize the already-known sources of bias in validation data.

3.3.2.1 Classification Algorithms

Lixin Gao was the first to describe the Internet as a strict hierarchy in which customers receive transit from the providers "above" them and redistribute routes according to economically incentives [175]. Based on this hierarchy, she described the notion of a "valley-free" path—a path that travels strictly upwards, then to at most one AS of the same height, and then strictly downhill. Using this property, her proposed algorithm tries to maximize the number of valley-free paths.

Rather than maximizing the number of valley-free paths, more recent algorithms often first determine the clique of provider-free ASes at the "top" of the hierarchy and then iteratively infer relationships. In 2013, Luckie et al. [292] proposed ASRank—one of the most-used classifiers till today. ASRank utilizes AS-triplets, a new metric called "transit-degree", and an extensive list of heuristics to classify relationships. Giotsas et al. later modified the ASRank algorithm to adapt it to the IPv6 routing ecosystem [186].

[27] usually only a small fraction of ASes connect remotely [94].

[28] e.g., while using bdrmapit—a tool to map IPs to routers and ASes that relies on relationship inferences—on paths obtained from a limited number of vantage points

[29] based on significance to our work and recency.

In 2014, Giotsas et al. used routing information, IP paths, and geolocation data to infer two more complex types of AS relationships: partial-transit and hybrid relationships [185]. If a provider exports routes towards its customers and peers but not towards its own providers, then the provider and customer have a partial-transit relationship. Further, two ASes have a hybrid relationship if their observed relationships differ throughout various Points of Presence (PoPs).

In 2019, Jin et al. proposed ProbLink—a meta-classifier that builds upon an initial classification (e.g., from ASRank) [248]. The algorithm assigns a probability to each link to be of a certain type based on, e.g., the relationships of other nearby links, refines the selected relationship based on the highest probability, and iterates those two steps until convergence. UNARI [167] takes the idea of probability one step further and produces a measure of certainty for each link type as its outcome. TopoScope [251]—as the newest classification algorithm—applies machine learning techniques on a large set of link features to perform its classification. Notably, this algorithm also predicts additional AS links that, despite note being visible, might exist.

3.3.2.2 Validation Data

Compiling a set of ground-truth labels is crucial to properly evaluate any classification algorithm. Yet, this step has proven to be rather difficult for AS relationships. Before Luckie et al [292], only the works by Gao [175] and Dimitropoulos et al. [150] presented validation data from a Tier-1 and via operator surveys, respectively.

In 2013, Luckie et al. compiled their validation data from (1) directly reported relationships (e.g., by operators through a web interface), (2) relationships extracted from routing policies encoded in WHOIS databases (more specifically, inside their `autnum` records) via the Routing Policy Specification Language (RPSL), and (3) relationships extracted from BGP Community encodings within the Internet Routing Registry (IRR) databases or public documentation (e.g., ISPs that host such encoding on their website).

While relying on multiple databases allows for frequent re-computation of validation data, the sources (2) and (3) suffer from a set of well-known challenges. Most records within the WHOIS databases are added and maintained voluntarily, hence, some records get stale (i.e., become inconsistent with publicly visible routing information) over time [99].

While the same may be true for the publicly documented BGP community encodings, those, in addition, suffer from ambiguity problems. Simply put, BGP communities are just colon-separated value pairs[30] [96] that can be tagged onto routes. Which information is encoded into/decoded from a specific BGP community depends on the AS that sets/reads it. Ambiguity is introduced when a single BGP community represents different meanings to (potentially overlapping) sets of ASes, e.g., while the BGP community `3356:666` could be recognized as an attempt to blackhole a route [264], AS 3356 (Level3/CenturyLink/Lumen) uses it to tag peering routes [398].

Despite those challenges, the data compiled by Luckie et al. presents the first extensive source of validation information. Recent classification efforts rely solely on recomputations of their third data source—relationships from BGP communities [167, 248, 251].

[30] or triplets, see large BGP communities [207].

3.3.2.3 Existing Insights into Validation Bias

Hard-to-Infer Links. Jin et al. [248] reported on sets of links for which it is challenging to infer them correctly. They describe those "hard" links as links with at least one of the following characteristics: (1) node-degree < 100, (2) observed by $50 - 100$ vantage points, (3) neither incident to a vantage point nor a clique AS, (4) stub links for which there is no triplet containing two consecutive clique ASes, and (5) links for which a simple top-down classification results in a conflict. They further showed that even sophisticated algorithms like ASRank wrongly infer many of the relationships for hard links and that the validation data set is skewed towards links for which it is easy to infer them correctly.

Clique & Vantage Point Links. Luckie et al. [292] show that for their 2014 validation data set links incident to a clique AS are over-represented while links between stubs and non-clique ASes are under-represented. They also note that this disparity is mostly due to the significant biased introduced by the community-based data set—the validation data that has been used for the more recent validations. Similarly, they report that the community-based data set over-represents links incident to a vantage point over those only remotely visible.

Complex Relationships. As discussed in §3.3.2.1, AS relationships can differ based on the PoP the link is observed at. Giotsas et al. [185] reported that their improved algorithm exposed around 1k relationships as hybrid and around 3k relationships as partial-transit. As the inference of such relationships can be ambiguous, they should be handled separately during the validation process.

3.3.3 Obtaining & Cleaning Data

In this section, we first describe how we obtain validation and inference data (§3.3.3.1). Afterward, we take a closer look at the validation labels and identify entries that either need to be removed or handled carefully (§3.3.3.2).

3.3.3.1 Obtaining Validation Data & Inferences

Validation Data. While ASRank's validation data from April 2013 is publicly available at [87], ProbLink and TopoScope do not contain validation data in their public repositories [249, 250]. Upon request, we received the same validation data for both tools—12 snapshots unequally spread between January 2014 and April 2018. Each snapshot was generated using the community-based relationship extraction method described by Luckie et al. [292] for their ASRank validation.

Inference Data. The monthly generated inference snapshots that are publicly available for ASRank, ProbLink, and TopoScope only overlap throughout 2019. As this period is not covered by any of our validation snapshots, we requested (and promptly received) an inference snapshot for April 2018 generated by ProbLink. To produce comparable results for all three algorithms, we continue using the inference and validation snapshots for April 2018 throughout the remainder of the section (unless explicitly specified otherwise). Notably, we use the term "inferred links" to refer to all AS links visible in the ASRank data set for April 2018.

3.3.3.2 Label Quality & Treatment

Spurious Labels. When taking a first look at the validation data, we notice 15 AS relationships formed with AS 23456. This AS is also known as "AS_TRANS" and is exclusively used to represent 32-bit ASNs for devices that only support 16-bit ASNs; therefore, AS_TRANS does not represent an actual network and hence can not have any business relationships. We further find 112 relationships involving reserved (e.g., for documentation or internal use, see [229]) ASes that should neither be publicly routed nor be used to validate business relationships.

Ambiguous Label Treatment. As briefly discussed in section 3.3.2, two ASes can have different relationships based on the PoPs they interconnect at [185]. In April 2018, the received validation data contains multiple labels for 246 relationships involving 233 different ASes. Arguably, those entries should be ignored for validation unless the classification algorithm explicitly infers or handles them; otherwise, it is ambiguous whether a simple relationship prediction is correct. Interestingly, we find that those validation entries are handled very differently in practice. If we treat an entry with multiple labels as P2P if it *starts* with P2P and otherwise as P2C, the number of P2P and P2C links in the validation data for 2017 and 2018 matches *exactly* those reported in the Toposcope paper [251]. We observe a similar match for the numbers reported for 2017 in the work by Jin et al. [248] if we treat an entry with multiple labels *always* as P2C.

Sibling Labels. Sibling (S2S) relationships represent links between two ASes that belong to the same organization and, hence, can use their resources interchangeably. When applying CAIDA's AS-to-Organisation data set [219], we find that 210 relationships in our validation data set and 2800 of the inferred relationships are actually sibling relationships and should be ignored during the validation process (unless specifically handled by the classification algorithm).

3.3.4 Is our Validation Data Biased?

Regional Imbalance. As briefly discussed in section 3.3.1, how an AS routes traffic may depend on its geographic region. To analyze regional bias, we first map each ASN to a geographic service region using IANA's list of initial ASN assignments [229] and then refine the mapping based on the daily delegation files published by the Regional Internet Registries (RIRs) [12, 32, 46, 278, 431]. We abbreviate AFRINIC, APNIC, ARIN, LACNIC, and RIPE NCC as AF, AP, AR, L, and R, respectively. While IANA's list bootstraps the mapping for all ASes, the RIR delegation files correct the mapping for resources transferred between different regions after IANA's initial assignments [391]. Notably, no mapping from ASes to geographical regions is perfect; even with large amounts of active scanning, we would neither be able to reliably measure all IPs (and respectively infrastructure) that belong to an AS [55] nor would we be able to perfectly geolocate them [97]. Yet, we argue that our mapping—which relies on an AS' organizational service region rather than its infrastructure footprint—is still representative enough to provide hints on regional biases, if they really exist.

Using this mapping, we separate AS links into different link classes: If one of the involved ASes is reserved, we discard the link. If both ASes belong to the same region, we mark the link class as <*region*>° (e.g., *AF*° for links between two ASes in AFRINIC). If the ASes belong to different regions, we mark the link class as <$region_1$>-<$region_2$> where <$region_1$> is always the lexicographically smaller region, i.e., we treat AS links as undirected links.

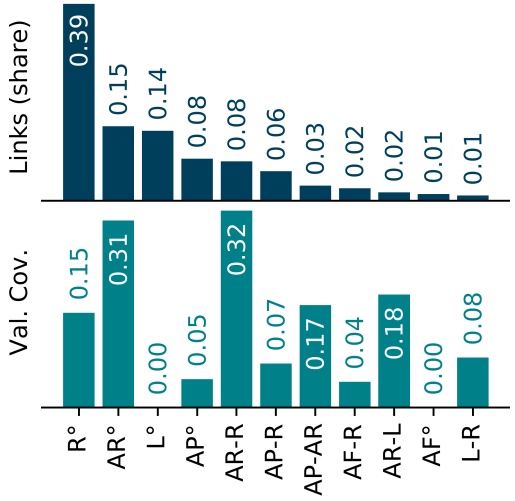


Figure 3.29: Regional imbalance.

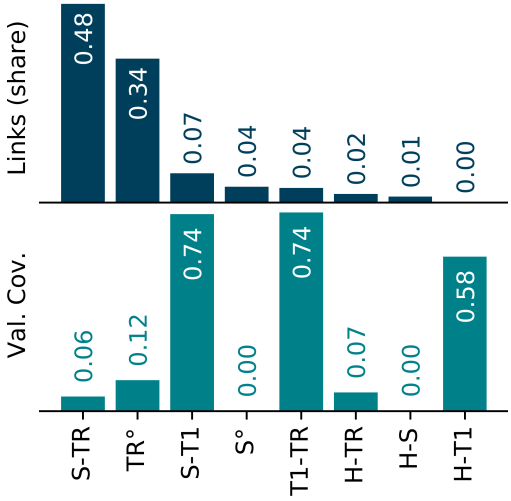


Figure 3.30: Topological imbalance.

Figure 3.29 shows the distribution of inferred relationships onto link classes as fractions (at the top) as well as the validation coverage (at the bottom), i.e., the fraction of links in a class for which we have validation labels. We observe that most (~79 %) of the relationships that we infer are between ASes of the same region. Yet, we observe drastic differences for the validation coverage among region-internal relationships: Even though we infer roughly the same number of $AR°$ and $L°$ relationships, we validate more than ~31 % of $AR°$ links but less than 1 % of $L°$ links.

Topological Imbalance. Next, we focus on whether the positioning of an AS in the Internet's hierarchical structure yields a mismatch in bias. First, we classify each AS into either "Stub" or "Transit" based on whether the AS has at least one other AS in its customer cone (see CAIDA's customer cone data set—available at [87]). Afterwards, we refine this basic mapping using two additional data sources: We re-classify ASes as (1) "Tier-1" providers based on a list from Wikipedia [507][31] and (2) "Hypergiants" (i.e., the largest content providers) based on the list generated by Böttger et al. [73].
Figure 3.30 shows the topological balance based on those classes in a similar style as Figure 3.29. We observe that we only have substantial validation data for classes that involve Tier-1 ASes. While this insight in itself is not very new (compare [292] and [248]), we find its impact to be more drastic than previously reported: For our two majority classes, S-TR and TR°, that, in summary, contain 82 % of all inferred links, we can only validate 6 % and 12 % of relationships, respectively.

While most of the inferred links are in the S-TR class, this class is rather uninteresting as it largely consists of P2C relationships (67.8% according to validation data) for which all three classifiers are well-known to perform near-perfect. Thus, we drill deeper into our second largest class, links between Transit providers. In particular, we want to understand whether the distribution of AS "size" matches between inferred and validated TR° links.

Figure 3.31 shows a heatmap over all TR° links in the inferred data (top) and the validated data (bottom) where the x-axis shows the transit degree for the larger incident AS while

[31] which largely overlaps with the set of clique ASes inferred by ASRank.

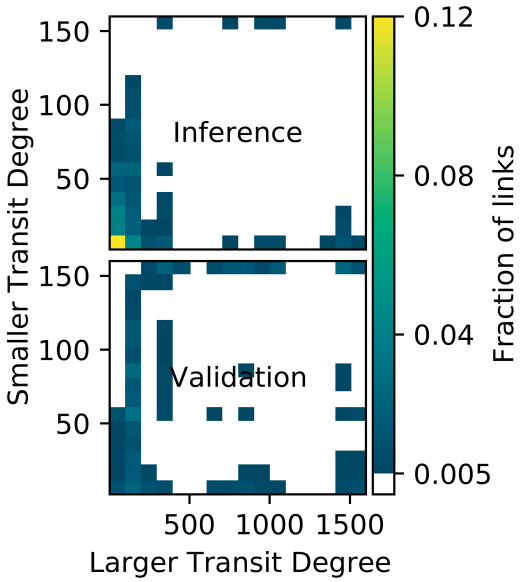


Figure 3.31: Transit degree imbalance for transit links.

the y-axis shows the transit degree for the smaller incident AS.[32] We observe that the vast majority of TR° links that we infer are between relatively small transit ASes (i.e., in the left-bottom corner). This mismatches with the more uniform distribution of our validation data. We further repeated this experiment with two alternative metrics: the provider-peer-observed customer cone—which relies on the correctness of the inferred business relationships and might hence be biased—and the node degree. The related figures (which can be found in Appendix A.3.2) suggest an even stronger mismatch.

3.3.5 Is our Validation Biased?

Now that we have a basic understanding of regional and topological bias mismatches in our validation data, we analyze how such mismatches translate to differences in classification correctness. For each of the tested classifiers, we calculate two confusion matrices (i.e., the number of True Positives, False Positives, True Negatives, and False Negatives) that result from treating either P2C links or P2P links as the "positive class."

Tables 3.5, 3.7, and 3.6 show the following classification correctness metrics for links of different classes[33]: (1) precision (PPV_X) and (2) recall (TPR_x) when choosing P2P links ($X \rightarrow P$) or P2C ($X \rightarrow C$) links as positive class[34], the number of P2P ($X \rightarrow P$) and P2C ($X \rightarrow C$) links per class as LC_X, and Matthew's Correlation Coefficient (MCC) as symmetric evaluation metric[35].

Simply put, the MCC takes all values of the confusion matrix into account (i.e., it does not matter which class is treated as positive), is relatively robust against class imbalance (i.e.,

[32]The row above 150 and the column to the right of 1500 catch all transit degree equal of larger than 150 and 1500, respectively. This prevents the few ASes with a substantially larger transit degree from distorting the plot.

[33]we only show those classes that contained at least 500 relationships in summary

[34]As they only provide additional mixtures of precision and recall, we decided to not show (balanced) accuracy and f1-score.

[35]The Fowlkes–Mallows index—as the second prominent symmetric evaluation metric—showed slightly less numerical change, yet similar results.

Class	PPV_P	TPR_P	LC_P	PPV_C	TPR_C	LC_C	MMC
Total°	0.982	0.990	14216	0.996	0.992	30105	0.980
AP-AR	0.979	0.979	546	0.988	0.988	928	0.967
AP-R	0.985	0.987	892	0.968	0.965	338	0.952
AP°	0.992	0.992	502	0.994	0.994	648	0.986
AR-L	0.930	0.976	43	0.999	0.997	872	0.950
AR-R	0.956	0.978	1752	0.994	0.987	5707	0.957
AR°	0.926	0.954	617	0.998	0.996	12871	0.937
R°	0.990	0.996	9587	0.995	0.989	8318	0.985
S-T1	0.000	0.000	26	0.999	0.999	15533	-0.001
S-TR	0.994	0.988	2538	0.995	0.997	5334	0.987
T1-TR	0.839	0.955	641	0.996	0.985	7260	0.886
TR°	0.991	0.996	10219	0.980	0.952	1822	0.959

Table 3.5: Per group validation table for ASRank

Class	PPV_P	TPR_P	LC_P	PPV_C	TPR_C	LC_C	MMC
Total°	0.966	0.976	14216	0.988	0.983	30105	0.957
AP-AR	0.973	0.939	546	0.960	0.983	928	0.927
AP-R	0.973	0.995	892	0.986	0.927	338	0.940
AP°	0.976	0.989	502	0.991	0.981	648	0.969
AR-L	0.619	0.975	43	0.998	0.962	872	0.761
AR-R	0.953	0.951	1752	0.984	0.984	5707	0.936
AR°	0.951	0.859	617	0.993	0.998	12871	0.899
R°	0.971	0.988	9587	0.985	0.964	8318	0.954
S-T1	0.295	0.650	26	0.999	0.998	15533	0.437
S-TR	0.980	0.987	2538	0.994	0.991	5334	0.976
T1-TR	0.718	0.670	641	0.971	0.976	7260	0.667
TR°	0.982	0.996	10219	0.978	0.903	1822	0.930

Table 3.6: Per group validation table for ProbLink

the fraction of validated P2P/P2C links in a class), and ranges between -1 and 1; values close to 1/-1 indicate positive/negative correlation between inference and validation while values close to 0 indicate correctness similar to an unbiased coin-toss [104].

Each table further colors differences between the classification correctness on the entire data set (Total°) as follows: If the per-class value is at least 1 % larger than the value for the entire data set, it is colored in green; if it is at least 1 %, 5 %, and 10 % lower, it is colored in yellow, orange, and red, respectively.

The tables first confirm common wisdom: All three algorithms perform near-perfect for P2C links. Yet, our evaluation further shows that all algorithms struggle with the same P2P link classes, namely AR-L, S-T1, and T1-TR. The low correctness for S-T1 links was already reported by [248], yet we disagree with their conclusion that "peering relationships between high-tier ASes and low-tier ASes are becoming more prevalent." We observe that most of those 26 links are formed with research ASes, anycast-based DNS providers, content delivery networks, and cloud providers, i.e., we observe that the problem lies in the broad aggregation of many diverse businesses models into a single "Stub" class, rather than a drastic change in policies. The overall correctness gap for P2P-based T1-TR relationships of up to 25 % shows that future classification efforts can still make substantial improvements for certain link classes. Yet, the increase of the correctness

Class	PPV_P	TPR_P	LC_P	PPV_C	TPR_C	LC_C	MMC
Total°	0.976	0.988	14216	0.995	0.989	30105	0.974
AP-AR	0.980	0.985	546	0.991	0.988	928	0.972
AP-R	0.983	0.994	892	0.985	0.959	338	0.961
AP°	0.986	0.992	502	0.994	0.989	648	0.980
AR-L	0.833	0.976	43	0.999	0.991	872	0.897
AR-R	0.947	0.975	1752	0.993	0.984	5707	0.950
AR°	0.930	0.943	617	0.997	0.997	12871	0.934
R°	0.984	0.993	9587	0.993	0.983	8318	0.976
S-T1	0.042	0.043	26	0.999	0.999	15533	0.041
S-TR	0.989	0.989	2538	0.995	0.995	5334	0.984
T1-TR	0.798	0.947	641	0.995	0.980	7260	0.858
TR°	0.989	0.996	10219	0.981	0.942	1822	0.954

Table 3.7: Per group validation table for Toposcope

gap from ASRank to the two follow-up algorithms shows that following a strategy of simply improving the overall classification error can lead to substantial correctness degradation for classes that contain fewer links. Finally, the reduced correctness for AR-L relationships might hint towards unique routing policies in the LACNIC region that are not yet captured by algorithms that were constructed and validated almost exclusively on the policies present in the RIPE and ARIN regions.

3.3.5.1 Case Study: AS714 Cogent Communications

To better understand the low performance for the T1-TR class, we do a case study for AS714 (Cogent Communications). We chose AS714 as it is involved in around half (54 out of 111) of all the links that were wrongly inferred as P2P (i.e., those links that decreased PPV_P) by ASRank (which has the best precision and recall for this class). For the remainder of this section, we call those links "target links."

When analyzing the paths that include our 54 target links, we were unable to find any triplet "$C \mid AS714 \mid X$" for which $AS714 \mid X$ is a target link and C is another clique AS. This observation is critical as such triplets are necessary for ASRank to arrive at a P2C inference for $AS714 \mid X$. While this provides us inside into what algorithmically caused the wrong inference, it does not explain why or how the routing phenomena that underpin those algorithms have changed.

To analyze target links beyond the public routing data, we focus on the 17 links that are also inferred to be P2P links in the most recent (Sept. 2021) snapshot. This allows us to directly trigger Cogent's looking glass to further investigate. We find that all the ASes involved in the 17 links consistently tag the routes they redistribute to $AS714$ with the BGP Community `174:991`[36]. This community prevents Cogent from redistributing the received routes to other peers—including all of the other clique members.

We discussed the issue with few of the involved operators and also looked up the related RPSL routing policy objects via RADB. We found that there are two reasons why ASes tagged this community: Cogent only offers them partial transit (i.e., routes towards customers but not towards peers) and inaccurate validation data[37] (only 1 case).

[36] Notably, this community is stripped before redistribution to customers; hence, it is rarely visible from the public routing infrastructure.

[37] i.e., contrary to the community-based validation data, the link is a P2P link rather than a P2C link.

3.3.6 Discussion & Outlook

Bias Mismatches. Throughout this section, we demonstrated bias mismatches between inferred and validated relationships. While the features that we analyzed showed substantial mismatches, other features could introduce similar (or even greater) ones. Even though a more complex analysis of additional groups of "hard links" lies beyond the scope of this section, we provide a list of twelve potential features for future analysis in the Appendix (§A.3.3).

Balance Through Sampling. While over-sampling of small classes or under-sampling of large classes are commonly used techniques to counteract biases, neither of them works (by default) well on AS relationships. Under-sampling prominent classes would result in a reduction of the already too small number of validated relationships. In contrast, simple over-sampling would bias the importance of specific error types (and often lead to over-fitting for ML-based classifiers). While there are more complex over-sampling methods (e.g., SMOTE [102], ADASYN [203], or MDO [1]) that synthetically (based on interpolation) produce new yet similar data points, theses techniques may introduce "incorrect" validation information when working with high dimensional data [168]. Yet, we might be able to leverage the heterogeneity and intrinsic, continuous change of the routing ecosystem to our advantage. If we understand for how long a certain set of relationships remains unchanged (e.g., via frequent exchange with network operators), we may be able to find a time frame after which the same AS can be re-sampled while still providing a unique-enough, new data point.

Future Validation Data. Most of our current validation data is passively obtained by scraping (poorly maintained) operator databases. We argue that compiling more extensive validation data requires active collaboration with network operators. In particular, we must clearly communicate incentives (e.g., services that they can benefit from) for why operators should accurately report (some of) their relationships through the channels they most commonly use (e.g., during operator meetings). A successful story using such a *do-ut-des* approach is the route collector project "Isolario." In only four years, the project acquired more peer ASes than RIPE RIS or Routeviews by partnering with HE.net. Whenever an AS connected to Isolario, HE.net would use the provided data to improve its statistics. The increase in reported size rendered the AS more attractive as a peering partner—a benefit that convinced many networks to continuously provide data.

Arguably, some operators may consider business relationships more sensitive than the routing information observed by a single (carefully selected) router. Yet, accurate information about a network's business relationships may be used to compile more valuable assets than simple statistics. One example would be router configurations generated by the Peerlock system. Peerlock utilizes relationship information to generate snippets of router configurations that prevent the redistribution of (accidental) route leaks [319].The mechanism's effectiveness may depend on the number of considered business relationships. Hence, operators might be willing to provide (and continuously update) their relationships in exchange for more secure and up-to-date Peerlock configurations. Similarly, relationship information may also be used to engineer recommendation systems for peering opportunities, i.e., rankings of beneficial IXPs (to peer at) and ASes (to peer with) for a given network.

Notably, the targeted interaction with operators could also counteract the current problem of missing validation data for an entire region that was reported in §3.3.4.

Future Research Efforts. Our analysis in §3.3.5 showed that (negligible) improvements in global classification correctness can severely impact the correctness for classes with

potentially fewer links. In line with this finding, we argue that the current goal of negligibly improving the overall correctness actually hinders progress in this research space. Hence, we advocate that future efforts should be evaluated against more diverse goals. Further, given our findings from §3.3.3.2, we advocate for more careful and explicit handling of spurious labels, sibling relationships, and complex relationships during future validation efforts.

3.4 AS Business Relationships and Routing Dynamics

The Internet consists of tens of thousands of networks—the autonomous systems—that redistribute routing information according to their business relationships among each other. While AS business relationships can be complex and location dependant [180, 185], the academic literature usually abstracts them into three main classes: transit relationships between providers and customers (C2P or P2C), settlement-free peering relationships between peers (P2P), and sibling relationships between ASes of the same organisation (S2S). Such relationship information is then used to, e.g., detect routing events including route leaks and prefix hijacks [319, 456, 460], infer performance and robustness bottlenecks [144, 517], emulate and counteract routing attack vectors [33, 463, 484], or map IP paths to AS paths [220, 313].

While relationship information is not available directly, it can be inferred from observed routes, e.g., via route collector projects such as RIPE RIS and RouteViews[38]. These projects collect the routing updates as well as periodic[39] snapshots of each peer's Routing Information Base (RIB). However, the AS relationship inference process is non-trivial and, over the last two decades, many academics works have been published. They can be loosely grouped into three classes: (1) algorithms that solve optimization problems using different objective functions [145, 150, 175, 483, 516], (2) algorithms that use a top-down approach, i.e., which first infer the apex of the routing hierarchy and then the lower levels [186, 292, 366, 521], and (3) algorithms that utilize probabilistic problem formulations and/or machine learning [167, 248, 251].

While their inference strategies differ, most recent efforts evaluate their algorithms using a similar approach: (1) They assemble input data based on (a subset of, see [248, 251]) the format introduced by Luckie et al. [292] in 2013, see [167, 248, 292][40]; (2) based on this input data, they produce inferences for their and previous algorithms; and (3) they compare the resulting inferences against some "best-effort" validation data. This type of evaluation is often repeated for varying subsets of vantage points [167, 251, 292] or few selected dates across multiple years [248, 251, 292]. While this evaluation approach already seems rather extensive, we argue that it entirely neglects that the routing ecosystem is highly dynamic. A recent work by Ariemma et al. [36] emphasized this by showing that update bursts affecting thousands of prefixes are the norm rather than an exception.

Thus, this section takes a step back and focuses on the impact that short-term routing dynamics have on the outcome of the business relationship inferences process. In particular, we focus on the inferrences produced by the ASRank algorithm [292] as it (1) is well-know in the research community[41], (2) is still considered the baseline that new

[38] Route collectors peer with many, selected (border) routers of various ASes.
[39] Every eight hours for RIPE RIS and every two hours for RouteViews.
[40] Please note that ProbLink paper is cited twice as their sections 3.1 and 7.1 claim different input formats.
[41] and has consequently been used by hundreds of academic publications throughout the last decade

algorithms have to outperform, and (3) requires only routing information as input[42]. To test the algorithm's inferences, we systematically generate tens of thousands of input data sets that slightly differ by the data they use (RIBS and/or updates), the time window over which they are aggregated, and the exact time at which their time window starts. When analysing the differences between the inference outcomes for these data sets, we find the following:

Clique inference: We first uncover that ASRank's inferred clique frequently includes "hypergiants"[43] (e.g., Akamai or Amazon) as the transit degree metric relies on an imperfect assumption. We further show that the clique inference algorithm is highly sensitive to one of its input parameters, see Section 3.4.3.

Link inference: We show that ASRank infers ~94 % of all links consistently (i.e., with the same label each time) throughout our three month period. When extending this observation to the validation phase, we distinguish between two classes of errors: *persistent* and *transient*. While the former occur in all input sets and hint at deeper algorithmic problems, the latter change their label across different input sets, likely due to short-term routing changes. Even though only ~6 % of links are inconsistently inferred, 55 % and 85 % of all inference errors for the median and worst snapshot are transient, respectively. While recent works achieved a $1.6\times$ error-rate reduction over ASRank [248] for certain snapshots, we show that ASRank's error-rate can be reduced by $5.4\times$ just by picking a different time for the evaluation, see Section 3.4.4. We conclude our work with insights into the minimum requirements needed to accurately detect the impact of short-term routing dynamics in future evaluation efforts.

3.4.1 Data Sources and Aggregation

Validation Data. We requested and received a snapshot of AS business relationship validation data from the authors of [389]; this snapshot is from April 1st, 2018 and was initially produced by the algorithm proposed in [292]. This best-effort validation data—which was already used to evaluate ASRank [292], ProbLink [248], and Toposcope [251]—is compiled from direct operator reports, routing policies within the autnum records of the WHOIS databases, and the BGP community encodings that can be found in the Internet Routing Registry (IRR) databases and public websites [292]. Please note that this validation data contains different caveats and biases; we handle caveats according to section 4.2 in [389] and discuss the potential impact of bias on our results whenever necessary.

ASRank Setup. The code to execute the ASrank algorithm is publicly available at [87]. As the authors hard-coded the reserved ASN ranges used throughout the input sanitation, we updated the code to properly reflect IANA's April 1st, 2018 allocation status (based on a snapshot of [229] that we automatically took at that time). Besides AS paths, ASRank requires a list of IXP route server ASNs as input. To generate this list we utilized a snapshot of PeeringDB [381] from April 1st, 2018 hosted by CAIDA at [90]. We obtained a list containing 131 ASNs by extracting the ASN from all `net` records for which the `info_type` field specified `Route Server`.

Hourly IPv4 AS paths. We pick the three months preceding our validation data snapshot (i.e., January 1st, 2018–April 1st, 2018) as observation period for our study. We use the IPv4 routing information in MRT format that was collected, published, and archived

[42] all other inputs remain static throughout our observation period

[43] Hypergiants are well-connected content providers that source substantial amounts of Internet traffic but usually do not provide transit [73, 393].

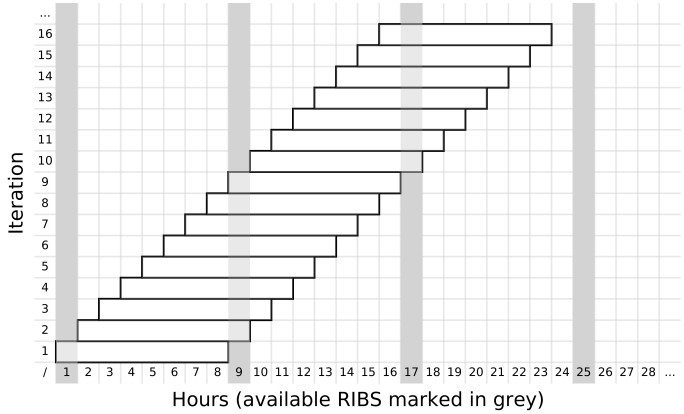


Figure 3.32: Inference snapshot generation for $ws = 8h$.

by the RIPE RIS [427] and Routeviews [364] projects. To fetch the data, we relied on the BGPStream utility [372] (more specifically, the pybgpstream Python library [91]). For each hour within our time window, we aggregated all visible IPv4 AS paths. While most hours only contain update files (i.e., incremental messages), every eighth hour includes a RIP snapshots (i.e., a full dump of all available routes) for each route collector. We explicitly utilize update files as we find that 5-20 % of routes change within less than 24 hours (based on the last-update timestamps across all rib entries in our observation period.)

Inference Snapshots. To generate the input AS path data for ASRank, we start at the first hour of our observation period, aggregate all hourly paths that fall into a given time window of size ws, and then repeat this process after shifting the window by one hour. We illustrated this process in Figure 3.32 for $ws = 8h$. Using this method, we generate between 1440 and 2176 slightly different input sets for each ws of eight hours ($8H$) as well as one ($1D$), two ($2D$), five ($5D$), ten ($10D$), and thirty days ($30D$). By shifting the window by 8 hours each time and only using the paths from rib snapshots, we additionally produced 258 snapshots of the "one rib snapshot per day for five consecutive days" ($5D_{ro}$) input format used by ASRank, UNARI, and ProbLink[44].

3.4.2 The ASRank Algorithm

ASRank is a top-down business relationships inference algorithm that works on AS path triplets—continuous (sub-)paths of length 3 (e.g., the AS path $A - B - C - D$ contains the two triplets $A - B - C$ and $B - C - D$). The entire algorithm contains eleven separate steps that—when ignoring various input sanitizing steps—can be grouped into three phases: **(1) infer a clique of provider-free ASes** at the apex of the hierarchy (and, consequently, infer peering relationships between those ASes), **(2) iteratively infer transit relationships** based on a set of heuristics that relies on size comparisons and AS path triplets for which one of the two links was labeled in the previous iteration, and **(3) infer all still unclassified links as peering relationships**. While step (1) contains its own sub-algorithm (see below), all steps heavily rely on the transit degree metric rather than the node degree metric as a measure of a network's size and/or importance to "...

[44]For ProbLink it remains unclear whether a single day or five days of data were used as input to the algorithm as Sections 3.1 and 7.1 state contradicting information.

avoid mistaking high-degree nodes (but not high-transit degree, e.g., content providers) for transit providers." [292], which we show to be incorrect in Section 3.4.3.

Transit degree. The transit degree metric can be calculated from AS path triplets ($A - B - C$) by adding the outer two ASes (A and C) to the transit set of the center ASN (B) and finally determining the length of the the transit set. Consider the following example: There are only two AS paths, $A - B - C$ and $C - D$. As only B appears in the center of a triplet, it is the only AS with a non-empty transit set (B's transit set is $\{A, C\}$). Hence, the transit degree of B is 2 while all other transit degrees are 0; in comparison, the node degree (i.e., the number of neighbors) is 1 for A and D and 2 for B and C. Notably, ASes that never appear in the center of a triplet will always have transit degree 0 but can have arbitrarily large node degrees.

Clique inference (sub-)algorithm. ASRank's clique inference starts by sorting all ASes by transit degree in descending order. It then finds the maximal clique C_1 among the N largest ASNs using the Bron/Kerbosch algorithm [78]—which we call the C_1 seed clique throughout the section. Notably, the authors argue that setting $N = 10$ "... reveals most clique ASes and is small enough to prevent the incorrect inference of a clique below the top of the hierarchy." [292], which we show to be incorrect in Section 3.4.3. Starting from the AS with the largest transit degree, ASRank then extends C_1 whenever an AS has links to all current C_1 members—which we will refer to as the extension phase. If an AS has links to all but one AS in C_1, it is added to C_2. After all ASes were tested, ASRank determines the clique of provider-free ASes by finding the maximal clique that can be build from all ASes in $C_1 \cup C_2$ (again using the Bron/Kerbosch algorithm). When determining each maximal clique, the summed transit degree of a clique is used as a tie-breaker between cliques with the same number of ASNs.

Version-dependant implementation bug. While analyzing the generated data, we noticed that the `asrank.pl` script, available at [88], does not implement the C_2 clique behavior described in the paper. The authors confirmed this finding after we contacted them, and also informed us that an updated (and correct) version of the script is available in the artifacts of Müller et al. [334], which were published on GitHub in 2019 [335]. As we received this information after our massive data generation effort, our insights—similar to those of other follow-up work (e.g., ProbLink [520])—are based on the old script that did not implement the C_2 behavior.

3.4.3 Clique Inference

After briefly introducing its main concepts, we first examine the influence of short-term routing dynamics on ASRank's clique inference. Figure 3.33 shows a normalized histogram (y-axis of each vertically stacked subplot) of the inferred cliques sizes (x-axis) over all snapshots for a given window size.

Based on this plot, we make two main observations: (1) incorporating update messages (all subplots except $5D_{ro}$) drastically changes the outcome of the clique inference algorithm and (2) the larger the window size gets, the smaller the set of unique inferred cliques gets, resulting in only two possible outcomes for a window size of thirty days.

Cliques from the $5D_{ro}$ input. We observe that the cliques inferred using the $5D_{ro}$ input format always contain the following 15 Tier 1[45] ASNs: 209, 286, 701, 1239, 1299, 2828, 3320, 3356, 3491, 5511, 6461, 6762, 6830, 7018, and 12956. Besides these ASNs, there are two disjunct sets of ASNs appearing in the clique: (1) ~30 % of times the IPv4 Tier 1 ASNs

[45] Based on Wikipedia's well-maintained list [507]

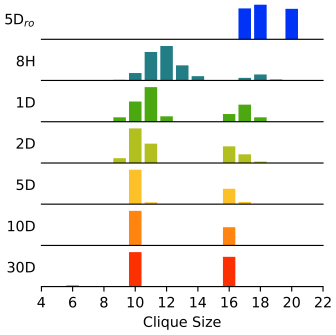


Figure 3.33: Clique size.

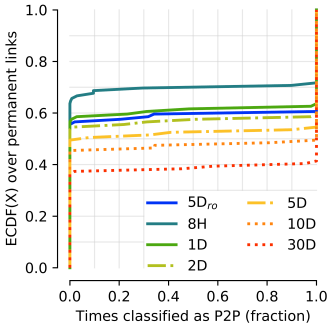


Figure 3.34: Inference consistency.

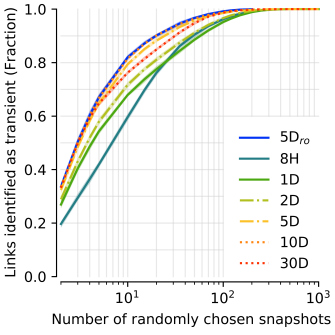


Figure 3.35: Inconsistency discovery

174, 2914, 3257, and 6453 as well as AS4134 (which purchases transit from AS3356 and AS174) help to form the 20 ASNs large clique. (2) ~70 % of times AS7922 and AS6939—two major transit networks that only rely on providers to reach small portions of the IPv4 address space—appear in the clique (with clique size 17). In the latter case, there is also a ~50 % chance that either AS20940 or AS16509 is added to the clique, increasing the clique size to 18.

Hypergiants as clique members. Most of the ASes that appear in the cliques are either Tier 1 networks or major transit providers. Yet, AS16509 (Amazon) and AS20940 (Akamai) are major content providers (also called hypergiants) that the authors explicitly wanted to avoid by changing from node-degree as underlying metric to transit degree. They still appear in approximately one-third of all inferred cliques due to the combination of the following two problems: *(1) The transit degree metric is build on an imprecise assumption.* While it is generally true that hypergiants do not actively seek out transit customers, most of them provide transit for their own sibling networks, e.g., Amazon's AS16509 provides transit for its siblings AS8987, AS14618, and AS62785. Consequently, AS16509 appears in many $X - 16509 - S$ triplets, where X is a large transit provider or a direct route collector peer and S is one of AS16509's sibling ASes. When checking for the hypergiants identified by Bottger et al. [73], only the transit degree of AS46489 is 0 while the transit degree of all other hypergiants is closer to their node degree than to 0, effectively nullifying the benefits of choosing the transit degree metric over the node degree metric.

(2) Some hypergiants peer with (almost) all Tier 1 networks. As a result, they would extend most C_1 cliques in the extension phase. If that is the case, why do they not appear in 100 % of all inferred cliques? Whenever AS16509 is not in the clique, it misses a link with AS2828.[46] While, in theory, AS16509 should get admitted to C_2 (unless AS4134—with which AS16509 also has no direct connection—is added to the clique before it), the `asrank.pl` script does not implement this behavior (as previously discussed in Section 3.4.2). Hence, AS16509 is not added to the clique.

In summary, whether a hypergiant enters the clique strictly depends on the visibility of few AS links. If these links are present, the clique extension phase will eventually—in fact, very early for hypergiants with non-0 transit degree—test a hypergiant and add it to C_1. We spot-checked a few snapshots in 2013 and found that both of our observations

[46] Please note that while paths with this link exist in the input data set (e.g., $27446 - 27446 - 2828 - 16509$), the list of inferred relationships does not include the link, i.e., the sanitation process removes this path even though we could not explicitly identify the corresponding sanitation rule.

already existed when the ASRank algorithm was originally published. At that time, hypergiants did not continuously appear in the clique as fewer Tier-1 ASes peered with route collectors, leading to "just-enough" unobserved (yet likely existing [49]) links between Tier-1 ASes and hypergiants.

Cliques from other input formats. When relying on any input format besides $5D_{ro}$, there is a set of eight Tier 1 ASes that consistently appears in the clique. These ASes are AS209, AS701, AS1239, AS1299, AS3356, AS6762, AS7018, and AS12956. Yet, most of these ASes do not appear as part of the C_1 seed clique, e.g., for the $ws = 30D$ snapshot staring at Jan. 31st, 2018 00:00 UTC+0 (which we picked at random) AS209, AS701, AS1239, AS6762, and AS12956 are added to the clique during the extension phase. With larger window size it becomes clearer that there are only two main AS sets that complete the above eight ASes to form cliques of size 10 and size 16. The clique with 16 ASes further contains ASN 172, 2914, 3257, 3320, 3491, 4436, 6453, and 6461, i.e., most of the ASes from the $5D_{ro}$ size 20 clique. The clique of 10 ASes further includes AS6939 as well as one of the following hypergiants: AS714 (Apple), AS8075 (Microsoft), or 16509 (Amazon).

Pinpointing the clique decision. To gain deeper insight into why the algorithm flips between two cliques, we ran a manual in-depth analysis on few example snapshots. We find that the outcomes depends on the first step in the clique inference: find the largest clique among the top N largest ASes by transit degree. While the difference between the transit degree of the first and second ranked AS is ~2000, the difference for the 6th and 20th ranked AS is only ~500, i.e., there are many ASes with comparable transit degree around the $N = 10$ limit that the authors deemed reasonable. Which ASes actually make the top 10 (and become eligible for the C1 seed) varies drastically based on few temporarily (in-)visible AS links. In summary, our analysis (for which details are available in Appendix A.3.6) suggests that the clique inference algorithm is very sensitive to how the N parameter is chosen.

Summary of takeaways: (1) For certain Hypergiants (e.g., Amazon or Akamai) the transit degree metric is often closer to the node degree than to 0 as they provide transit to their sibling ASes. (2) The clique inference algorithm always includes (even more so if the C_2 set mechanic would be properly implemented) hypergiants into the clique unless a very limited number (often only one) of links is temporarily invisible. (3) When including BGP updates in the input format, the clique inference algorithm mainly chooses between two drastically different cliques. The actual result strictly depends on the N parameter (chosen by the authors as 10) that limits the number of ASes among which the C_1 seed clique is build.

3.4.4 Link Inference

After taking a closer look at the clique inference, we now dive deeper into the link inference. While a correctness analysis of each inference rules is beyond the scope of this section, we again analyse the consistency of the inference results.

Consistency of Link Inferences. Figure 3.34 shows an ECDF over the fraction of times each link was assigned a peering/P2P relationship across all snapshots of a given window size. Please note that this figure only incorporates permanent links (i.e., links visible in every snapshots), and that the plot looks the almost identical for temporary links (not shown). We first observe that the fraction of P2P links increases when using longer observation periods—a well-known property of the routing ecosystem [204, 367].[47] More

[47]The $5D_{ro}$ format is the only outlier to this observation; however, it is, in general, hard to compare the

importantly, we observe that, regardless of the input format, the vast majority of links are either never or always classified as P2P, i.e., they are continuously assigned the same relationship. In fact, only 3.61 %, 5.87 %, 5.51 %, 5.38 %, 4.97 %, 4.72 %, and 4.10 % of the inferences observed for the $5D_{ro}$, $8H$, $1D$, $2D$, $5D$, $10D$, and $30D$ are inconsistent, respectively.

Classifying errors by stability. Next, we study the importance of these inconsistencies on the evaluation of inferences. We evaluate each snapshot by calculating error rates for three types of link sets: (1) for all links (ANY), (2) only for p2p links[48] (P2P), and (3) only p2c links (P2C). Whenever we evaluate an inference for a snapshot as erroneous, we classify it into *persistent* or *transient*. We call an error *transient* if its links is one of the < 6 % of inconsistently inferred links; otherwise, we call the error *persistent*. While transient errors are affected by short-term routing events, persistent errors hint at deeper (potentially algorithmic) problems.

Figure 3.36 shows an ECDF of the fraction of transient errors across all snapshots of a given window size. First, we point out the drastic variance across snapshots for the $5D_{ro}$ format. While ~55 % of all errors are transient for the median snapshot, the snapshots with the smallest and largest fraction contain ~37 % and ~87 % transient errors. A second important observation is the difference between P2P and P2C links. Only between ~9 % and ~32 % of errors for P2C links are transient, yet for certain snapshots up to ~93 % of errors for P2P links are transient. Our results further suggest that for ANY, P2P, and P2C links error-rate decreases of up to $2.22\times$, $3.57\times$, $1.19\times$ on median and $7.38\times$, $14.64\times$, and $1.49\times$ are theoretically[49] possible as a result of shifting the observation window in time. When we actually compare the $5D_{ro}$ snapshots with the least and most errors, ASRank's error rate has decreased by $5.4\times$. Finally, Figure 3.36 shows substantially lower variance within and difference between the fraction of transient errors for ANY, P2P, and P2C links if the input format contains update information.

Uncovering transient errors. As this result suggest that short-term routing dynamics influence the evaluation of business relationships, we now focus on how to uncover their impact in the future. While we analyzed thousands of slightly shifted snapshots in this work, it is an Utopian idea to require future works to go through similar efforts. Hence, we ask the question: "How many of the total N inconsistently inferred links can we uncover when using only s randomly chosen snapshots?" To better understand the involved variance, we repeat the analysis 100 times for each size s and calculate the minimum, median, and maximum fraction of uncovered links.[50] Figure 3.35 shows the fraction of links that were detected as transient on the y-axis and the number of randomly chosen snapshots on the x-axis. It shows the median across all 100 repetitions as lines and shades the area between the min. and max. fractions in the same color. We first observe that the variance in our experiment is rather small; in fact, the min. to max. areas are barely larger than the linewidth of the curves. We observe that for the $5D$ and $10D$ input formats, ~80 %, ~90 %, and ~95 % of inconsistent links can already be observed with ~13, ~22, and ~60 snapshots; hence, we belief that future studies can arrive at meaningful claims about the impact of short-lived routing dynamics when using approximately 20 (random) snapshots across three months.

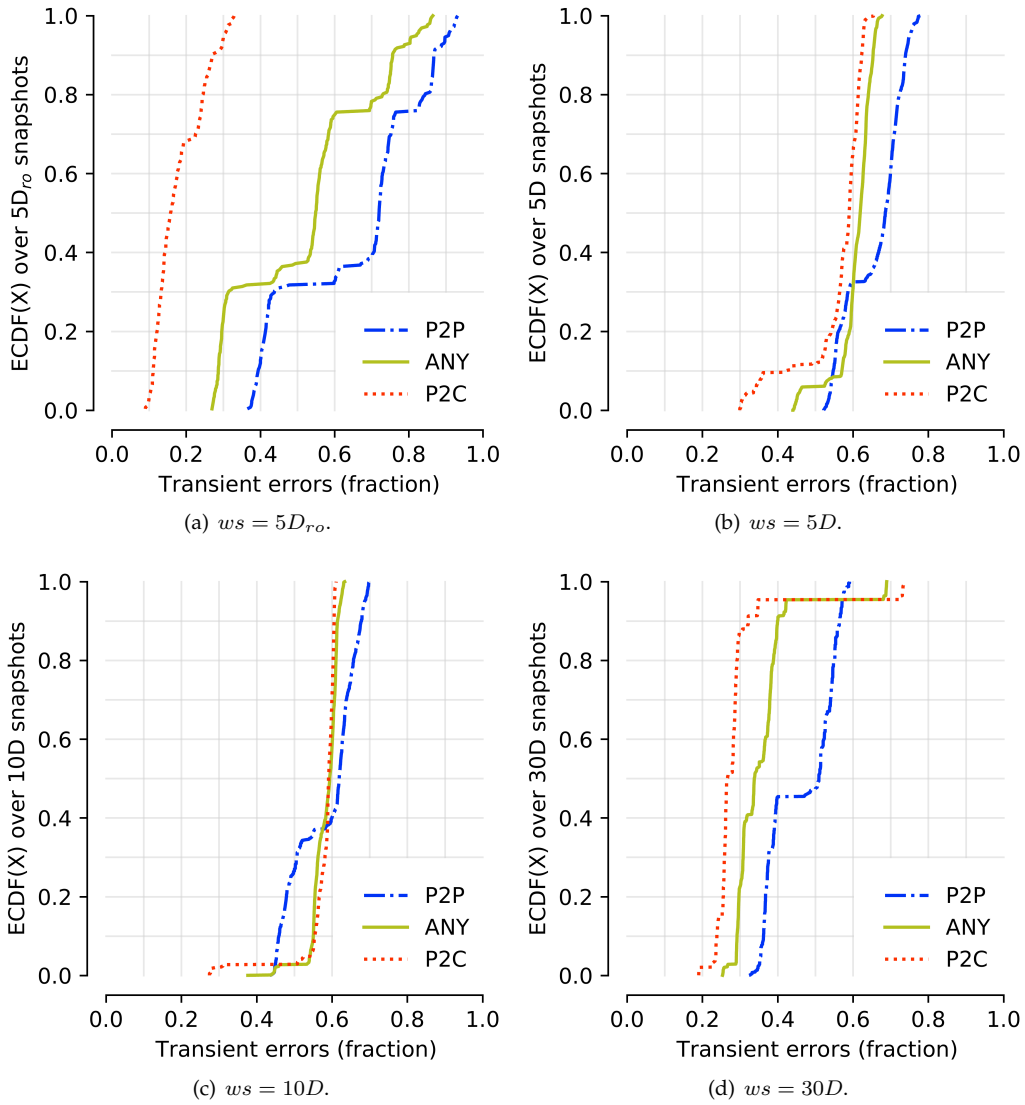
Summary of takeaways: (1) Even though only less than 6 % of links are inconsistently inferred, these links account for the majority of errors in evaluations. (2) Choosing ~20

perspective gained from five independent, single-moment samples to that of a continuous observation period due to, e.g., short-lived routing information that (dis-)appears between two snapshots.

[48] as identified by the validation data

[49] iff all transient links are inferred in-/correctly for the worst/best snapshot

[50] we only test up to $s = 200$ for $5D_{ro}$ due its limited amount of snapshots



(a) $ws = 5D_{ro}$. (b) $ws = 5D$.

(c) $ws = 10D$. (d) $ws = 30D$.

Figure 3.36: ECDF over contribution of transient links to miss-classified links across all ws snapshots.

snapshots across three months uncovers ~90 % of transient inferences. (3) Shifting the observation window may result in a $5.4\times$ decrease in ASRank's error-rate.

3.4.5 Discussion

Input format. While the $5D_{ro}$ input format has been frequently used to evaluate business relationships, we showed that it introduces various problems. In Section 3.4.3, we demonstrated that various algorithmic problems in the clique inference remained hidden by using this input format. We further demonstrated in Section 3.4.4 that this input format leads to a very unstable fractions of transient inference errors per snapshot—

which makes it hard to properly interpret previous evaluations. We belief that future business inference efforts should not only be more elaborate on why they decide for a certain input format, but also analyze the sensitivity of their method towards it.

Clique inference. The authors of ASRank decided to infer transit-free ASes rather than Tier-1 ASes "Since Tier-1 status is a financial circumstance, reflecting lack of settlement payments, ...". The same decision was later made by the authors of ProbLink [520], TopoScope [251][51], and UNARI [167]. We showed throughout this section that this decision may lead to drastic short-term variance in the inference results. Hence, we argue that using a fixed clique as input to the algorithm (i.e., the same way that CAIDA produces their publicly available inference results since August 2019) may provide inference results that are more robust to short-term routing dynamics. The Tier 1 network page from Wikipedia [507] may be useful to determine the input clique as its content is actively maintained, moderated[52], and discusses deeper nuances such as "regional Tier-1s."

Past & future evaluations. While it is common that authors test their new algorithms on varying subsets of vantage points [88, 167, 251] or for long-term routing dynamics (e.g., via few snapshots across multiple years) [248, 251, 292], it is rare that authors also analyse the impact of short-term routing dynamics. This is problematic as we showed that substantial error-rate improvements can be achieved by slightly shifting the observation period. While, based on our analysis in Section 3.4.4, we would recommend to use around 20 snapshots across 3 months to properly capture short-term routing dynamics, the authors of ProbLink already made a step in the right direction: They aggregated daily data sets and ran ten individual evaluations for ten consecutive days. We hope that such types of evaluations become more prominent in the future.

3.4.6 Conclusion

In this section, we have analyzed the impact short-lived routing dynamics have on business relationship inferences and their evaluation. We have compared thousands of inferences generated from subtly different inputs and showed that these dynamics can substantially vary the outcomes. Throughout our analysis, we have uncovered various issues (e.g., not only is the clique inference algorithm highly sensitive to a argument, it also frequently mistakes hypergiants for large transit providers) that remain hidden when only analyzing long-term routing dynamics. We further have shown that for certain snapshots almost all inference errors are transient (i.e., are classified inconsistently across snapshots) even though only $< 6\,\%$ of all links are transient. These transient inferences may lead to a $5.4\times$ decrease in error-rate when slightly shifting observation window for which ASRank is invoked. To enable future works to incorporate short-lived routing dynamics in their evaluation, we analyzed the discovery rate for transient links and found 20 random snapshots within a three month period may uncover up to 90 % of all transient links.

3.5 Chapter Summary

We analyzed the routes available via multi-lateral, bi-lateral, and private peering at a large European IXP. For multi-lateral peering, we analyzed Route Server snapshots from

[51] Toposcope relies on the clique inferred by ASRank

[52] Misleading or wrong edits are promptly discussed among various editors and reverted if necessary.

eight of the world's largest peering LANs and showed that most of their routes lead to out-of-continent locations via three or more AS hops. While remote peering might be a major contributor to the geographic distance of Route Server destinations, we observe that close and distant IXP members alike provide lengthy, unattractive routes to the Route Server. When comparing those findings to peering LAN traffic, obtained through a collaboration with one large IXP, we saw that mostly one-hop routes saw substantial traffic. In fact, we observed that 25 % and 77 % of IPv4 and IPv6 Route Server prefixes with at least four hop long paths see no traffic at all. This indicates that even though Route Servers provide many routes, most IXP members only make use of local routes. Afterwards, we used two heuristic-based methodologies to infer bi-lateral and private peering routes from the IXP's peering LAN traffic. During our inferences, we carefully isolated transit connections that were established over the peering LAN—a phenomenon previously reported by Ager et al. [15]. Based on our inference, we observe that at least 19.8, 57.1, and 57.4 % (37.3, 37.4, 37.8 %) of all routed IPv4 (IPv6) address space can be reached at our IXP via multi-lateral, bi-lateral, and private peering, respectively. Those results provide practical contrast to the 70+ % reachability theoretically calculated by Böttger et al. [71]. Finally, we show that almost all of the top 10k egress prefixes of a large European eyeball network can be reached via bi-lateral peerings. In contrast, we also find that up to 15 % of top 10k domain-serving prefixes can not be reached via any type of peering at our IXP. Notably, we observe that most of these prefixes belong to large transit and Tier 1 providers.

After confirming the blindness of the public route collector infrastructure, we took a closer look at its inherent biases. Before our work, significant biases in IMP vantage point placements have been documented by experience papers from well-established scientists (e.g., [82, 436]) or via a few dedicated analyses [52, 53, 212]. Besides reproducing their original findings, the framework we introduced drastically facilitates finding new biases among diverse dimensions and tracking the evolution of these biases over time. We demonstrate, e.g., that while IXP peering oriented networks are over-represented in RIPE RIS, their peering policies are representative of the Internet's peering ecosystem[53]. Our framework further provides the tools needed to counteract the impact of bias when extending IMP infrastructures or choosing an unbiased set of vantage points for measurements.

After uncovering new biases in the placement of our Internet Measurement Platforms, we took a closer look at our state of the art AS business relationship inference algorithms. We demonstrated substantial geographical and topological bias mismatches between those links for which we inferred relationships and those that we were able to evaluate, e.g., while we infer roughly the same amount of business relationships for the ARIN and LACNIX regions, our validation data covers 31 % of inferences within the ARIN region yet only less than 1 % of inferences within the LACNIC region. We further find that he near-perfect classification precision of 96-98 % for the entire validation data set drops by 14-25 % (depending on the algorithm) for peering relationships between Tier-1 and transit providers. This finding further emphasizes the importance of the biases within our validation data. Our analysis further showed that (negligible) improvements in global classification correctness can severely impact the correctness for classes with potentially fewer links—an insight that calls for the introduction of more diverse evaluation goals in the future.

Finally, we analyzed the impact short-lived routing dynamics have on business relationship inferences and their evaluation. We compared thousands of inferences generated

[53] as captured by PeeringDB.

from subtly different inputs and showed that these dynamics can substantially vary the outcomes. Throughout our analysis, we have uncovered various issues (e.g., not only is the clique inference algorithm highly sensitive to a argument, it also frequently mistakes hypergiants for large transit providers) that remain hidden when only analyzing long-term routing dynamics. We further have shown that for certain snapshots almost all inference errors are transient (i.e., are classified inconsistently across snapshots) even though less than 6 % of all links are transient. These transient inferences may lead to a 5.4× decrease in error-rate when slightly shifting observation window for which ASRank is invoked. To enable future works to incorporate short-lived routing dynamics in their evaluation, we analyzed the discovery rate for transient links and found 20 random snapshots within a three month period may uncover up to 90 % of all transient links.

Discussion. Our understanding of the Internet's routing topology—even at the AS-level—is not only very limited but further seems to diminish over time (compare ~30 % AS link visibility from route collectors in Ager's 2012 study to the ~22 % discussed in Section 3.1). At the same time, our tools to infer business relationships are heavily influenced by short-term routing dynamics. This lack of knowledge makes it difficult to accurately estimate the baseline from which measurement configurations are drawn, ultimately clouding our understanding of the biases present in the placement of vantage points and hence Internet routing measurements in general. Further, our results suggest that it remains unclear whether recent business relationship inference algorithms really made progress over their predecessors. Even though they may depend on time, location, protocol, and diverse handshake agreements, we still infer AS business relationships primarily based on time-wise and location-wise aggregated AS paths and evaluate them against binary (rarely ternary) identifiers obtained by scraping voluntarily maintained operator databases.

While it is not trivial to break this paradigm, we belief that long-term progress can be achieved through various independent contributions. While the lack of broad, up-to-date validation data set is currently a major road block, previous efforts (e.g., the Isolario project) showed that the operator community is willing to share their information for mutual benefits. Hence, a platform that safely ingests data from operators and produces, e.g., secure router configuration snippets may be able to obtain, actively maintain, and characterize business relationship validation data. Another contribution could be an annotation service for routing information based on, e.g., decoded BGP communities, data plane measurements, and data base look-ups, providing the inference process with a richer set of characteristic for each route. Further, a field study is needed to better understand the spectrum of current relationship types and the data sources that would be needed to identify them (e.g., AS paths alone are unable to distinguish economical variants such as paid-peering and settlement-free peering).

Chapter 4
Managing IPv4 Address Exhaustion

After our assessment of the limitations of our observation infrastructure and inference methods, we are now prepared to focus on one of the most pressing routing ecosystem changes throughout the last decade: IPv4 exhaustion. IP addresses are essential for networks as they allow devices to connect to the Internet and communicate with each other. While there are more than enough available IPv6 addresses, many parts of the Internet—including, e.g., end-hosts, servers, and routing infrastructure—have not fully adopted IPv6 due to, e.g., a lack of awareness, the potential upgrading costs, or the reluctance to change a well-running system. As a result, ASes may not rely on IPv6 addresses for global reachability. At the same time, the pools of available IPv4 addresses that can be assigned by the RIRs have depleted rapidly throughout the last two decades—a process known as IPv4 exhaustion. This puts operators in a position in which a simple switch to IPv6 may be economically or operationally infeasible while obtaining IPv4 addresses through the standard allocation process is impractical.

In this chapter, we asses two potential ways to deal with this situation. First, we take a look at the emerged buying and leasing markets that allow operators to temporarily or consistently obtain IPv4 address space at low and high cost, respectively. Besides investing substantial resources into obtaining address space, we further analyze the viability of announcing the already obtained address space in smaller, hyper-specific, prefixes. The contributions of this chapter are:

- We provide an overview over the IPv4 exhaustion status, address allocation policy, and waiting list status for each RIR. While three RIRs gave fully depleted their IPv4 address pools and subsequently started waiting lists, two RIRs still maintain small reserves of IPv4 addresses.
- We use the RIR transfer statistics as well as pricing data from four large brokers to investigate the IPv4 address transfer market. We find that prices have doubled between 2016 and 2020, yet they are significantly lower than predicted by previous work. In 2020, the average price for an IP address, regardless of the RIR, lies around $22.50 with little variance.
- We revisit the concept of address space delegation using publicly available BGP

data, RPKI data, and data from RIPE's RDAP database. We observe that, for the RIPE region, the amount of delegated IPv4 address space visible in BGP heavily underestimates the actual market size. We further find that current leasing prices vary significantly—$0.3 to $2.4 per address per month—based on the leasing provider.

- We analyze the CIDR sizes, BGP communities, and services involved in the publically announced hyper-specific prefixes across multiple years and infer potential use-cases. We find that IPv4 HSPs mostly represent (internal) routes towards peering subnets and blackholing, whereas IPv6 HSPs are mainly used for address block relocations and, in substantially fewer cases, blackholing. We further find that HSPs are unlikely to contain many end hosts and that they are rarely used for traffic engineering.
- We compare the HSPs visible in BGP with those that were explicitly entered into routing databases—in particular, the IRR and RPKI databases—to investigate intended or accidental use of HSPs. We find that while thousands of ASes explicitly specify their intent to use HSPs, many HSPs likely represent accidentally leaked routes.
- We discuss future direction in which the research and operator communities may use or handle hyper-specific prefixes.

4.1 IPv4 Buying & Leasing Markets

As of today, the world has almost run out of unallocated IPv4 addresses to satisfy the ever-growing demand for new addresses [341]. Three RIRs, namely ARIN, LACNIC, and RIPE NCC, have depleted their pool of unallocated IPv4 addresses, and the last two RIRs, APNIC and AFRINIC, currently allocate from their last /10 and /11 address block, respectively. Thus, all have changed their IPv4 allocation policies to shortage management [27, 41, 340, 354]. As part of those policies, all RIRs have reduced the size of the IPv4 blocks a new member can receive (e.g., the RIPE NCC only allocates /24 IPv4 prefixes [418]), and most of them instantiated a waiting list for already approved but not fulfilled requests. Since their pools contain no more unallocated addresses, depleted RIRs have to rely on organizations to return some of their allocated resources; thus, the amount of time a request stays on the waiting list is unpredictable. As a result, networks may need to use alternative ways to acquire IPv4 address space: leasing and buying.

The Internet standardization and governance community already foresaw the exhaustion of IPv4 address space some 20 years ago and introduced IPv6[210]. In the past, networks got a decently sized address block upon becoming an RIR member and could request additional resources as needed. Today, this only holds for IPv6. Despite increasing IPv6 adoption (especially by end-users [191]), many popular services are still not reachable via IPv6 [140]. Thus, many networks—even in 2020—prefer (or may need) new or additional IPv4 addresses over IPv6 addresses [303]. While techniques such as Carrier-Grade NAT (CGN) [413] reduce the need for public IP addresses, they do not eliminate it.

Motivated by the fact that the RIPE NCC allocated its last IPv4 address block on 25th Nov 2019, we study how networks satisfy their demand for IPv4 addresses in 2020. We highlight the challenges and the costs that IPv4 dependent networks face by analyzing the IPv4 address markets that emerged as a result of the address shortage. We do not only

study allocations and address transfers within the RIRs but also the emerged leasing and buying markets for IPv4 address space. Accordingly, the contributions of this sections are:

- For each RIR, we summarize the IPv4 exhaustion status, address allocation policy, and waiting list status in §4.1.1.
- To understand the IPv4 address transfer market, we use the RIR transfer statistics as well as pricing data from four large brokers, see §4.1.2. We find that prices have doubled since 2016, yet they are significantly lower than expected given predictions from a previous study [285]. Today, the average market price per IPv4 address, regardless of the RIR, lies around $\$22.50$ with little variance.
- To get a glimpse of the IPv4 leasing market, we revisit the concept of address space delegation using publicly available BGP data as well as data from RIPE's RDAP database. We observe that, for the RIPE region, the amount of delegated IPv4 address space visible in BGP heavily underestimates the actual market size. We further find that current leasing prices vary significantly—$\$0.3$ to $\$2.4$ per address per month—based on the leasing provider, see §4.1.3.

We outline how our work extends existing literature in §4.1.4 and discuss the interplay of our findings combined with insights from discussions with 13 IP brokers in §4.1.5. We, finally, point out that due to the huge range of leasing prices the amortization time for buying IPv4 address space can range from less than a year to 36 years.

4.1.1 Getting IP Resources

To actively participate in the Internet's BGP routing ecosystem, organizations need IP resources. Initially, Jon Postel manually assigned IP resources. This process was formalized later in the 1990s, see RFC 7020 [216]. The Internet Assigned Numbers Authority (IANA) manages hierarchical allocations of IP addresses and AS numbers to the five Regional Internet Registries (RIRs). Those RIRs, namely AFRINIC (African region), APNIC (Asia Pacific region), ARIN (American region), LACNIC (Latin American region), and the RIPE NCC (European and Middle Eastern region), are responsible for address assignment, bookkeeping, and community support. Some of the IPs that Jon Postel assigned are still not managed by the RIR framework and, hence, are called "legacy" addresses. There are three options to obtain new IPv4 resources: *(i)* joining an RIR as a member and requesting new address space; *ii)* joining an RIR and buying address space; and *iii)* leasing address space.

RIR Membership. RIRs are membership-based organizations: If an organization receives address space from an RIR, it typically becomes a member of that RIR—a Local Internet Registry (LIR). Membership status is not necessarily bound to address space allocation but rather implies participating in the Internet governance effort. RIRs develop their operational policies through engaging discussions among their members. LIRs can (e.g., by voting for board members or (dis)approving financial decisions) influence the policies of their RIR. To become and stay an LIR, an organization has to pay an annual membership fee plus fees depending on the number of requested resources. Yet all five RIRs differ in their exact pricing model. [11, 30, 38, 275, 423]. In October 2020 the RIPE NCC handled a first time incident where the "Right to Registration of IPv4 Addresses" was auctioned to recover money in a legal case [342]. case.

RIR	Down to last /8	Start of Recovery
AFRINIC	03/31/2017 [9]	— (last /11, 01/13/2020 [13])
APNIC	04/15/2011 [26]	07/27/2014 [28] (still /10 available)
ARIN	04/23/2014 [358]	09/24/2015 [127]
LACNIC	02/15/2017 [276]	08/19/2020 [273]
RIPE NCC	09/14/2012[430]	11/25/2019 [418]

Table 4.1: IPv4 exhaustion timeline for the five RIRs.

Towards IPv4 exhaustion. All RIRs maintain IPv4 and IPv6 address pools. They receive addresses from IANA, other RIRs, or organizations that no longer need their allocated IPs. The IPv4 address pools decreased drastically over time. Since IANA allocated its last remaining IPv4 address blocks to APNIC on 31st January 2011, the RIR pools can no longer receive reserved IPv4 addresses. As a result, the RIRs soon reached their last /8 (see Table 4.1). To distribute the remaining resources fairly, all RIRs soon established more restrictive assignment policies—this phase is also known as "soft-landing".

IPv4 exhaustion. ARIN, LACNIC, and the RIPE NCC completely depleted their address pools in 2015, 2020, and 2019, respectively. Now, those RIRs have to recover unused IPv4 address space before they can fulfill any requests. APNIC and AFRINIC combined still have less then the equivalent of a /9 of IPv4 addresses left. Currently all RIRs recover IP address space if an organization closes down or the original criteria for the initial assignment are no longer satisfied [370]. APNIC actively contacts members who have received delegations that are not at least partially visible in the global routing system [354]. Upon recovering IP address space and removing the associated objects from the database most RIRs put the blocks into a six month quarantine period before redistributing it again [339]. As a result, assignment policies became more restrictive, and most RIRs introduced waiting lists.

Today, AFRINIC [13], ARIN [39], and LACNIC [272] limit the assignable address space per organization to a /22. For APNIC [30] it is a /23 and for RIPE [418] a /24. That does not mean that the RIRs force organizations to return address space if they previously received a larger address block. To highlight the impact of IPv4 address exhaustion, we point out that the waiting lists of ARIN, LACNIC, and RIPE held up to 202, 275, and 110 approved requests, respectively [45, 277]. For Arin, this corresponds to waiting times of up to 130 days.

Since November 2019 RIPE used recovered address space to fulfill all approved waiting list requests [341, 428]. Currently, its address pool still contains around 340k IPv4 addresses (equivalent to more than a /13) [420]. In contrast, APNIC abolished its waiting list on 2nd July 2019 [29] since it only included requests from members whose already allocated address space exceeded a /22. Since then, APNIC only hands out IPv4 addresses to new members.

Buying IPv4 addresses. Since neither the limited size of approvable IPv4 address space nor the required waiting time satisfies their needs, many LIRs started to buy additional IPv4 address space. Formally, LIRs do not buy IP addresses per se—there is no clear notion of legal IP ownership [459]—but rather acquire their usage rights. In October 2020 the RIPE NCC had a first time incident where the "Right to Registration of IPv4 Addresses During a transaction, a buying LIR pays the selling LIR such that the latter invokes a resource transfer of the address space to the buyer. This resource transfer may involve additional payments to the RIR. After the transfer, the buying LIR is responsible for the RIR's annual resource maintenance costs. Certificated IPv4 brokers help to

facilitate this process [42, 43, 422]. They connect buying and selling LIRs, help them in price negotiation, and often handle the formalities of the address transfers. Based on discussions with 13 brokers, their commissions range from ~5 % to ~10 % and may be charged to either LIR or partially by both of them.

Leasing IPv4 addresses. While Internet Service Providers lease address space to their customers for a long time, there is a recent increase in the number of organizations that lease their address to any organization independently of routing or connectivity agreements. Leasing providers are LIRs that temporarily delegate the usage rights for some of their address space to a customer. While leasing does not involve any resource transfers at the RIR, it may encompass altering objects in the WHOIS database—a database that contains information about Internet resources, organizations, and contact persons. A leasing contract can restrict address usage and may include hosting or network connectivity agreements, or both. For hosters, the leased address space is usually still located in their own AS. In this section we are considering two types of IP leasing models. In the first model an IP broker only leases IP address space to a customer, while in the second the IP leasing is bundled with antoher service contract, e.g., infrastructure hosting.

Not all IP addresses are equal. Over the years, IP based blacklists have become very popular to mitigate malicious activities, e.g., E-Mail spam or flooding attacks [472]. These blacklists contain IP address blocks that are associated with said activities, and network operators rely on them to filter ingress traffic. Once an IP address block appears on a blacklist, it can be hard to remove it again—the IP address is tainted. IP address blocks that never appeared on a blacklist and have no association with any malicious activity are known as ”clean IPs.” To keep their address blocks clean, leasing providers often demand information on how a potential customer intends to use the leased resources. Besides, leasing providers often install registry data—e.g., Shared WHOIS Project records (also known as SWIP records [44])—to secure their remaining address space from getting blacklisted when spamming is detected in a delegated block [473]. Similarly, most LIRs check the “reputation” of address blocks before buying them to ensure the addresses are globally reachable.

4.1.2 The IPv4 Address Reseller Market

When the RIPE NCC entered the "Recovery Only"-phase, many organizations expected an increase in brokered IPv4 transfers as well as address prices. Therefore, we first analyze the number of transfers. Next, we augment our findings with public and private pricing information from four IPv4 address brokers. Finally, we discuss the ability to obtain address space from foreign RIRs.

Each RIR publishes daily transfer statistics. Those not only include transfers between LIRs but also include transfers that are results of Mergers and Acquistions (M&A) of companies that consolidate their IPv4 address space. While AFRINIC [10], ARIN [40], and RIPE NCC [421] label such transfers APNIC [31] and LACNIC [274] do not. Thus, we can remove M&A transfers for the former RIRs. For the latter RIRs, we could potentially use the heuristics proposed by Giotsas et al. [184]. However, since the authors do neither present an evaluation nor an analysis of the output’s sensibility to the input parameters, we decided against this option.

Figure 4.2 shows the number of transfers aggregated over three months for each region

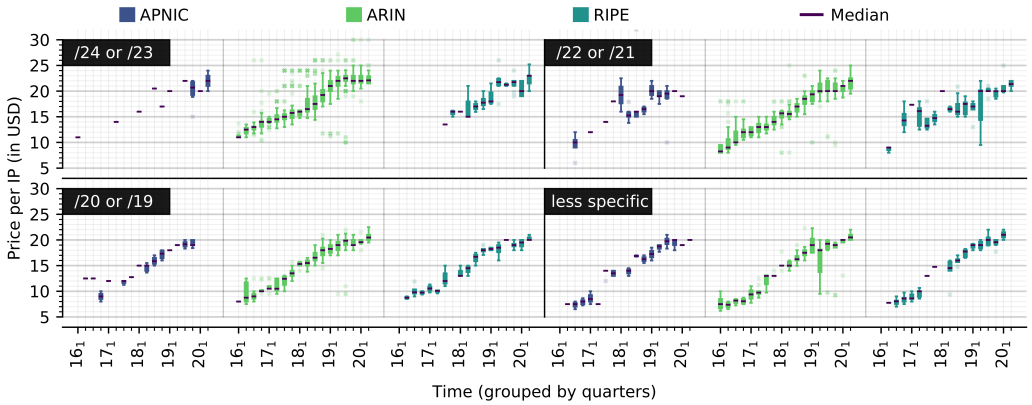


Figure 4.1: Evolution of Price per IP based on prefix size and region.

from October 2009 to June 2020. Observe that the regional transfer markets[54] start, i.e., for APNIC, ARIN, and the RIPE NCC, once the RIR was down to its last /8 (compare to Table 4.1). ACNIC has recently depleted, the number of transfers in those regions is negligible. This can be attributed to two factors: (i) cloud providers consume a huge amount of IP space [491], yet most of their data centers are located in the ARIN, APNIC, and RIPE NCC regions [308]; (ii) mobile rather than fixed-line broadband access is the norm in the AFRINIC and LACNIC regions—mobile operators often rely on IPv6 [470] and deploy carrier-grade NAT for IPv4 more aggressively [413]. We further observe that the number of monthly transfers fluctuates significantly. While for the RIPE NCC the pattern aligns with the end of each year, we cannot identify any specific patterns for ARIN. This indicates that the IPv4 transfer markets are in flux.

To understand the cost of buying IPv4 addresses, we use publicly available pricing information from IPv4.Global [239] as well as private pricing information obtained from Brander Group [196], IPTrading.com [237], and IPv4 Market Group [197]. All data is anonymized: Rather than containing the IPv4 prefix and the participating organizations, we track the number of IP addresses transferred per region. Since prefixes less-specific than a /16 are rarely transferred, they are identifiable; therefore, our data set only contains transactions for /16 or more-specific prefixes. In total, we obtained pricing information for 2.9k transactions between 1st January 2016 and 25th June 2020. Since our data set only contains 31 transfers for those regions, and they do not yet have vibrant transfer markets, we exclude AFRINIC and LACNIC from our analysis.

Figure 4.1 shows the pricing information as box plots grouped by prefix size, region, and three months interval. Across all prefix sizes, our data contains 8-23, 83-196, and 12-19 transactions APNIC, ARIN, and RIPE per interval, respectively. We first observe that there is no statistical difference in pricing across the regions, i.e., whether a prefix is allocated to the APNIC, ARIN, or RIPE NCC region has no significant impact on its price. In contrast, buying IP addresses in /24 or /23 blocks is more expensive than buying larger address blocks; when a broker decides to sell a large address block in separate small parts, the associated secondary costs increase. The broker not only needs to find more buying parties but also needs to initiate more separate transfers. As our analysis

[54]Here, "region" refers to the region associated with the RIR that allocated (and maintains) the IPv4 address block. ICANN established this relationship when initially handing our addresses to an RIR. If an RIR transfers address space to another RIR, the region is updated accordingly. Therefore, every region-based analysis refers to the RIR that an IP address block belongs to.

does not cover prefixes less specific than /16, we can only report on them based on information from IPv4 brokers: Since large continuous blocks are rare, the price per IP rises again. Overall, we observe that prices, regardless of the actual prefix size or region, have doubled since 2016, which correlates with the diminishing availability of unallocated address blocks from the RIRs.

Starting from Spring 2019, the IPv4 market seems to have entered a *consolidation phase*, i.e., a state in which the market price barely changes and where the number of transfers no longer corresponds to the actual demand [232]. We observe the first, see Figure 4.1, and learned about the latter via discussions with the brokers. During a consolidation phase, sellers hold back many of their assets as they wait for the largest sellers to dictate new pricing trends. When discussing our findings with thirteen additional brokers, they argued that the disclosure of market prices by IPv4.global provided a reference point for the average cost of an IP address that is also available to potential buyers. Thus, increasing prices beyond this reference point lead to a decrease in potential customers. As a result, most brokers told us that they strictly align their prices with those advertised by IPv4.Global.

An organization can also get IP address space from a foreign RIR and then request a transfer of the addresses to the RIR associated with its actual region. While such action must adhere to the policies of all involved RIRs, some policy regulations are easy to satisfy. For example, ARIN's current policy practice for out-of-region requestors is [126]: A requestor is eligible for receiving an allocation if it announces the least-specific prefix in ARIN's service region. Thus, organizations with a single PoP in ARIN's service region (even if it only consists of a single router) are eligible to receive addresses. However, inter-RIR-transfers can only take place between APNIC, ARIN, and the RIPE NCC since these RIRs agreed on common transfer policies [416]. Figure 4.3 shows the number of inter-RIR transfers by origin and destination for each RIR from 2012 to 2020. While the number of inter-RIR transfers continuously increases, the blocks transferred get smaller. Most transfers move address space away from ARIN and either to APNIC or RIPE. The latter may, in part, be explained by ARIN's assignment policy and different feature sets of the RIR management interfaces.

4.1.3 The IPv4 Address Leasing Market

Buying IP addresses does not only require a significant upfront investment but also introduces delay (i.e., for the address transfer). Leased address space is often available in less than a day [236] and only requires monthly payments. Thus, IP leasing is an attractive option for businesses with immediate needs, small budget, or limited long-term perspective. To understand to which extend the leasing option is currently used, we analyze two sources of data: *(i)* we *infer* leasing agreements from routing information by revisiting the concept of BGP delegations; *(ii)* we use a snapshot of RIPE's WHOIS database and queries to its RDAP database to track address block delegations.

Inferring BGP delegations. Leasing IP address space is only economically useful if the organization also announces the prefix within the BGP eco-system. Therefore, most leased address space should be visible as delegated address space whereby the leasing provider may still announce a less-specific prefix. We say that a *delegator* AS S owns a prefix P and delegates a more-specific sub-prefix P' to a *delegatee* AS T. We infer a delegation P'_{ST} if we observe that S and T originate P and P', respectively. To infer such delegations, we build upon the work of Krenc and Feldmann [268] (our extensions are marked with $+$): *(i)* We obtain the set of all prefix-origin pairs. *(ii)*$^+$ We remove all pairs

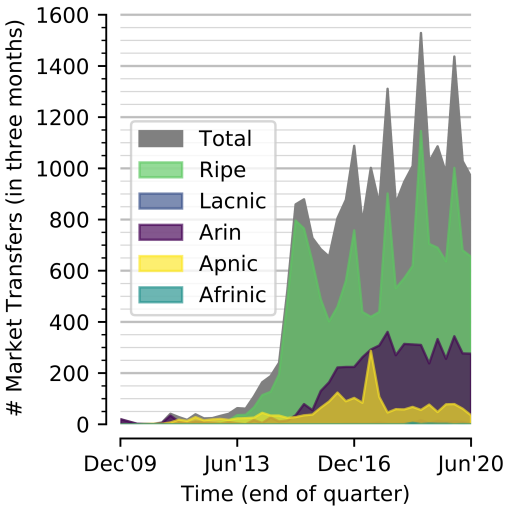


Figure 4.2: # of market transfers.

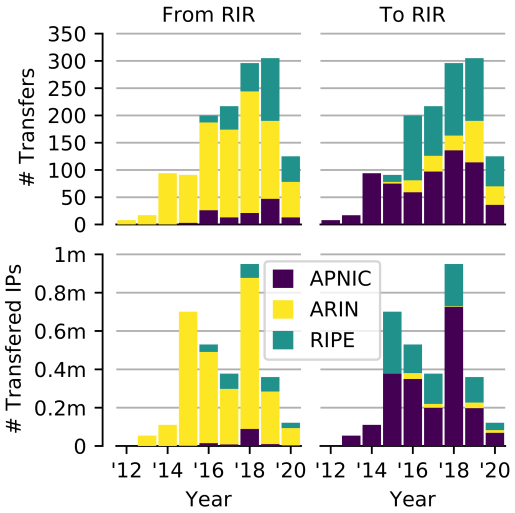


Figure 4.3: Inter-RIR transactions.

seen by less than half of all BGP monitors to ensure global visibility[55]—this limits the impact of, e.g., local misconfigurations or locally-spread BGP hijacks. *(iii)* We remove pairs for which the respective prefix is originated by an AS_SET or multiple ASes. *(iv)*$^+$ Relying on CAIDAs AS-to-Organization mapping [89], we remove delegations between ASes of the same organization within the next available snapshot[56]. *(v)*$^+$ We compensate for temporarily not announced delegations—many delegations show on-off-patterns—based on insights from analyzing delegation consistency in RPKI: If we observe the same delegation ten days apart while not observing a conflicting delegation (i.e., we observe P being delegated to another delegatee AS T') in the meantime, we presume that the delegation also exists for all days in between. We chose this rule as its fail-rate (i.e., the fraction of possibilities with an invalid conclusion based on all possibilities with a valid premise.) is below 5 % for delegations inferred from RPKI snapshots between 01/01/2018 and 06/01/2020, see Appendix B.1.

Limitations. Despite taking precautions against hijacks, our algorithm may still infer a delegation between a victim AS and a hijacking AS if the hijack is performed using a more-specific prefix. Our algorithm may wrongly infer delegations in combination with BGP-based scrubbing services (i.e., services that announce their customer's prefixes, analyze and drop incoming malicious traffic, and tunnel the remaining "clean" traffic back to their customer).

BGP-delegations. We apply our inference algorithm to the routing information collected by RIPE RIS [427], Route Views [364], and Isolario [244] between 1st January 2018 and 1st June 2020. We aggregated the data daily; i.e., we use the RIP snapshot at 0:00 UTC+0 and all update files for that day. If an update file is missing, we additionally download the first available rib snapshot afterward. To sanitize our data, we remove all routes for private and reserved address space [486], routes that contain ASes currently reserved by IANA [229], and routes that contain a loop in their AS-PATH. We find that our extensions significantly reduce the number of inferred delegations but eliminate the large

[55] As long as the monitor threshold is chosen between 10% and 90% the difference in inferred delegations is negligible.

[56] Internet resources such as address space or AS numbers are assigned to organizations; thus, ASes that belong to the same organization can utilize each other's address space without an actual leasing agreement.

variance produces by the previous approach, see Figure B.2. Overall, we see an increase in delegations by ~7 % with a negligible change in delegated IPs caused by decreasing delegation sizes.

RDAP-delegations. Some RIR's maintain publicly accessible Registration Data Access Protocol (RDAP) [353] interfaces designed to eventually replace the WHOIS protocol. Like WHOIS, this database contains registration information but is more extensive. If an LIR assigns address space to an "end-host"[57], the `parentHandle` field contains an RIR-unique identifier for the parent network—this can be used to infer delegations. Since RDAP interfaces do not allow wild-card or range requests, we rely on `inetnum` objects from a current WHOIS snapshot [424] as input space to our RDAP queries. While the RIPE NCC and ARIN provide the `parentHandle` field in their RDAP responses, only RIPE NCC also offers publicly available WHOIS snapshots. Thus, we restrict this analysis to the RIPE region[58]. First, we select all `inetnum` objects from RIPE's WHOIS database with delegation-related types: The `SUB-ALLOCATED PA` type refers to address space sub-allocated to another organization, and the `ASSIGNED PA` type refers to address space assigned from an LIR to an end-host. We find ~4.5k entries and ~3.96M entries for June 2020, respectively. Notably, most, 91.4%, of the `ASSIGNED PA` entries are for address blocks smaller than /24. To minimize the load on RIPE's RDAP interface, we ignore all blocks smaller than /24. We further remove intra-organization delegations, i.e., where the child block has the same registrant or administrator as the parent block. Aftward, we have 181k remaining RDAP-based delegations.

BGP-delegations vs. RDAP-delegations. When comparing the delegations identified via BGP on June 2020 with those from RDAP delegations we observe that: BGP-delegations cover only ~1.85 % of the RDAP-delegated IPs while the RDAP-delegations cover ~65.7 % of the BGP-delegated IPs in the RIPE region (using [421]). This limited coverage of BGP-delegations implies that the leasing market is significantly larger than previous work has predicted [268].

The limited coverage of BGP-delegations may be due to: *(i)* the assumption that the delegated prefix *and* the covering prefix are announced within the BGP eco-system; *(ii)* even if the delegatee announces it, the more-specific prefix may be aggregated and is no longer globally visible; and *(iii)* large LIRs often delegate medium-sized address blocks to ISPs. These ISPs use some of the address space but reserve significant chunks for future customers. The latter is invisible in BGP. While organizations have incentives to enter their leasing agreements into the WHOIS and RDAP databases (e.g., to reduce the blacklisting risk due to malicious activity in a leased prefix [473]), not all leasing provider require entries. Hence, RDAP-delegations will also miss some leasing agreements.

RDAP-delegations are complementary to BGP-delegations—the former captures the administrative processes, the latter the actual usage. Neither can catch all leasing agreements. Thus, combining both data types is essential to estimate the size of the IPv4 leasing market.

Leasing prices. To understand the leasing price evolution, we fetched the advertised leasing prices from 12 websites [142, 178, 205, 214, 233, 236, 238, 240, 289, 290, 352, 392] between 26th October 2019 and 1st June 2020. On the 1st June 2020, we added 9 additional websites [24, 95, 141, 208, 283, 388, 402, 406, 471].

Even though some websites offer up to 10% discounts when either leasing larger prefix sizes or committing to multi-month contracts, we consider the prices for leasing a /24

[57] Here, end-hosts are networks that cannot further assign the addresses to other LIRs or end-hosts.
[58] Snapshots from other RIRs may depend upon signing access agreements.

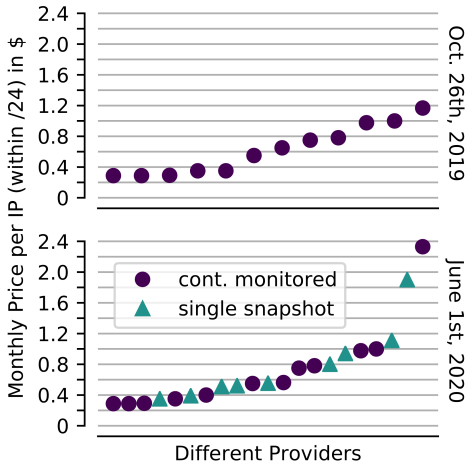


Figure 4.4: Advertised leasing prices.

for a single month. The advertised leasing prices are shown in Figure 4.4. In general, we observe that prices vary substantially: from $\$0.30$ to $\$2.33$ per IP per month. We also find no structural price difference between pure leasing providers compared to IP leasing bundled with infrastructure hosting. This indicates that the market has not converged yet. Still, only three providers changed their advertised leasing prices: Heficed reduced its monthly per IP price from $\$0.65$ to $\$0.40$; IPv4Mall [240] increased it from $\$0.35$ to $\$0.56$; and IP-AS [233] from $\$1.17$ to $\$2.33$. IP-AS also seems to have tested the market in January by increasing the price to $\$3.90$—more than $> 10\times$ the lowest available price.

4.1.4 Related Work

The IP address trading market is not yet widely discussed in the research world, as practical relevance became only apparent with the global IPv4 depletion. Livadariu et al. [285], in 2017, characterized the evolution of RIR allocations and transfers. Using the few publicly disclosed transactions, they proposed a model to predict the value of the IPv4 address market. Their estimated price of $\$30$ per IP for the end of 2015 exceeds the actual price at that time by about 200 %. In contrast to their claims, we also find no statistically significant difference in the prices of different regions. In 2015, Richter et al. [410] provided a detailed, historical perspective on changes and policies that preceded the current IPv4 exhaustion and pointed out possible solutions, e.g., Internet-wide adoption of IPv6, carrier-grade NAT, and efficient use of address space (reassign unrouted address space). In 2015, Edelman and Schwarz [157] proposed a market regulation rule to avoid excessive trading and provided an estimation model for IP leasing and buying prices. However, their model lacks validation based on real-world data and shows opposing trends to the market price evolution we observe. While Van Audenhove et al. [495] showed that permitting active transfers of IPv4 resources between entities of the same RIR leads to a "thriving" transfer market in 2013. Livadariu et al. [286] reported that most transfers until 2013 occur between a small set of countries. In 2011, Osterweil et al. [373] discussed the implications of possible regulations for IPv4 transfers. They argued for using additional reverse DNS records to validate IP ownership. Giotsas et al. [184] reported inconsistencies between RIR transfer data and information inferred from BGP data sets. They also point out that frequently transferred blocks are more likely to be involved in malicious activity.

4.1.5 Discussion

In the first half of 2020, the IPv4 address market showed a stable price range. When, in contrast, considering current leasing, buying, and maintenance prices, we observe that possible amortization times for buying IPv4 addresses are somewhere between 10 months and multiple tens of years. Through our discussions with brokers, we learned that the average amortization time for their customers is between two to three years. Given the current state of the buying market, IPv4 address holders need to be careful *when* to sell their assets: An early sell may not reach the maximum economical gain (if prices increase) while waiting for too long may result in an economical loss (if prices fall). On the other end, the demand for IPv4 addresses currently outweighs its supply. As a result, potential customers are currently willing to pay higher prices.

How an organization engages with the leasing and transfer markets is often strongly correlated to its business model: Internet Service Providers (ISPs) often buy blocks larger than /20 with the intent of leasing parts of them to (potential) customers; on the other end, long-term customers buy address space smaller than /20 to fulfill their addressing needs and terminate their address leasing contracts with, e.g., an ISP. Young businesses often start by leasing small address blocks and then increase their leased address space until they have secured enough funding to buy address space. In contrast, many VPN providers continuously lease address space but frequently "rotate" the actual IPs such that it is harder to block their service. Finally, spammers often use short-lived leasing agreements of varying sizes while ensuring that their own address space remains clean when they engage in malicious activities. Another strategy we have encountered in discussions with brokers is *buy and lease back*: In this model organizations owning more IPv4 address space than they currently utilize (e.g., ISP) sell this address space to a broker and, in return, only lease the amount they need with previously agreed terms should they ever need additional space. Such a contract can provide the selling organization with immediate cash flow while ensuring a continuous supply of addresses.

There are still allocated but unused IPv4 addresses even though the RIRs have moved to shortage management. The active policies require the return of unused addresses [340, 354], but the current market situation provides little incentive to release acquired resources. As such, the number of available IPv4 addresses will soon hit rock-bottom, a point at which the world-wide deployment of IPv6 becomes inevitable for future services. With advancements in IPv6 deployments, the focus may shift away from IPv4 addresses. Thus, one might argue that the question is no longer if IPv4 address prices will drop, but *when*. On the other hand, brokers also told us that they expect a huge price increase because many players prefer to engage in the ”known” costs of acquiring IPv4 address space rather than moving their networks forward to IPv6.

4.1.6 Conclusion

In this work, we present the current state of the IPv4 address market and provide a base for further scientific analysis as well as a starting point for organizations to assess their options for offering and obtaining IPv4 addresses. The current state of IPv4 address exhaustion is: APNIC and AFRINIC still have a small amount of address space available, RIPE NCC was recently able to fulfill the approved requests in its waiting list using recovered addresses, ARIN’s waiting list has waiting times up to 130+ days, and LACNIC’s waiting list currently holds 275 approved but not fulfilled requests. As a result, networks that are still relying on IPv4 lead to vibrant leasing and buying markets.

Although prices for IPv4 addresses doubled since 2016, previous work [286] significantly over-estimated the price development—especially for small address blocks. While the prices per IP decrease with address block sizes (except for large blocks), we find no statistically significant difference between regions. We observe that the buying market has been volatile since 2016 but went into a consolidation phase at the beginning of 2019. Even though individual price changes by the "big players" may dictate future pricing trends, directly buying IPv4 address space—at an average cost of $\$22.50$ per IPv4 address for a /24—is currently an economically viable option. When focusing on the IPv4 leasing market, we find that, through the lens of BGP, the amount of leased IPv4 addresses increased by 7% over the last two years. Current leasing prices range from $\$0.30$ to $\$2.33$ per IP per month (for a /24). This huge range (even though the service levels may differ) indicates that the market has not converged on a price tag.

We showed that state-of-the-art delegation inferences are noisy and only reveal a small fraction of the actual delegations. We highlight that delegations in the RPKI system provide a rather different perspective on the consistency of delegations. We argue that future research efforts should combine routing information, RPKI data, as well as the RDAP databases to obtain a better picture of the leasing ecosystem and its characteristics.

4.2 Hyper-specific Announcements

Autonomous Systems use the Border Gateway Protocol to announce prefixes to their peers [405]. Each BGP-speaking router of an AS can decide to accept or reject incoming announcements based on the prefix itself, the AS path, or other attributes that are attached to a route (e.g., BGP community values). Due to this concept, every single AS (and, in fact, also all its routers) may have a unique viewpoint into the Internet's routing ecosystem [436].

Many popular BGP guidelines recommend the rigorous filtering of prefixes that encompass only a few addresses [153, 156, 306, 356, 359, 464, 465] and, hence, those prefixes have been shown to propagate neither far nor reliably [481]. While the possible reasons for announcing these types of prefixes are broad and range from traffic engineering over multi-homing configurations to prefix-hijack prevention [120, 227], the boundary for announcements which are deemed "widely acceptable" are usually considered to be a /24 prefix in IPv4 and a /48 prefix in IPv6.

In this section, we perform an in-depth analysis of prefixes that are more specific than those boundaries (i.e., /25 to /32 IPv4 prefixes and /49 to /128 IPv6 prefixes). We refer to those prefixes as **hyper-specific prefixes** (HSPs) and analyze their prominence in the global routing ecosystem, the functions that they serve, and whether they represent intentional or accidental announcements. More specifically, we make the following main contributions:

Observability. We perform a decade long analysis of HSPs as seen by 67 route collectors (see §4.2.1). We find that the number of HSPs has increased substantially since 2010 and peaked in 2018 at around 115K IPv4 and 18K IPv6 prefixes. While we observe that especially HSPs which are announced consistently for an entire year are visible by hundreds of collector peers, the average HSP can only be seen by a handful of them.

Use Cases & Functions. We analyze potential use cases of HSPs by combining insights from different analyses of CIDR sizes, BGP communities, and service hit rates across multiple years. (see §4.2.2). We find that IPv4 HSPs mostly represent (internal) routes towards peering subnets and blackholing, whereas IPv6 HSPs are mainly used for

address block relocations and, in substantially fewer cases, blackholing. We further find that HSPs are unlikely to contain many end hosts and that they are rarely used for traffic engineering.

Intended or Accidental Use. We compare the HSPs visible in BGP with those that were explicitly entered into routing databases—in particular, the Internet Routing Registries (IRRs) and Resource Public Key Infrastructure (RPKI)—to investigate intended or accidental use of HSPs (see §4.2.3). We find that while thousands of ASes explicitly specify their intent to use HSPs, many HSPs likely represent accidentally leaked routes.

The Future of HSPs. We discuss how the research and operator communities could make use of HSPs in the future. Finally, we plan to maintain a dashboard providing up-to-date HSP statistics to help AS operators in detecting leaked internal routes.

4.2.1 Observability

We begin our exploration of hyper-specific prefixes by analyzing their current and past presence in the Internet's routing ecosystem.

In particular, we examine the routing information from hundreds of globally distributed ASes—called "feeder ASes" or "route collector peers"—collected by the Isolario [244], RIPE RIS [349], and Routeviews [364] projects. Starting from January 2010, we generate snapshots consisting of a week of RIB and update files every three months until October 2021. We provide further details about the choice of this window size in Appendix B.2.2. We employ various filtering steps to sanitize the data from, e.g., announcements of unallocated Internet resources, certain noisy origin ASes[59], or temporarily misconfigured feeder ASes. We also reached out to operators of noisy origin ASes. Two of these operators were not aware of this problem, but addressed it quickly upon our notification. A comprehensive lists with justifications for the individual steps can be found in Appendix B.2.5.

First, we investigate the evolution of HSPs from January 2010 to October 2021. Figure 4.5 shows the number of hyper-specific prefixes (lines) and ASes that originate them (bars) over time. Looking at the left sub-plot, we observe that the number of observed HSPs (despite being noisy) consistently increases throughout the eleven years. We see more than 10k IPv6 and 100k IPv4 HSPs by the end of 2021, i.e., approximately one-tenth of all visible prefixes are hyper-specific (see Appendix B.2.3 for further details). Relative to the increase in HSPs, we also observe an increase of ASes that originate them, with 584 and 2.5K ASes announcing hyper-specific prefixes via IPv6 and IPv4 by the end of 2021, respectively.

Given that the route collector projects acquired feeder ASes within our observation period, the increasing trend could simply be a sampling error. To test this hypothesis, we replicate the analysis using only data from the 105 IPv4 and 45 IPv6 feeder ASes that were consistently peering with route collectors throughout all snapshots. While our observations remain similar for IPv6, there are two changes for IPv4: (1) the number of hyper-specific prefixes that can be seen by a consistent set of ASes appears more stable (if any trend exists, it remains hidden behind the massive fluctuations); and (2) despite an initial increase, the number of ASes originating HSPs stagnates after 2016. Therefore, the number of IPv4 HSPs does not show a constant increase over time, but rather we

[59]These ASes announced either (1) an extraordinarily high number of HSPs (i.e., 100 or more times higher than in other snapshots) or (2) HSPs in an extraordinarily high number of anchor prefixes for a limited amount of time.

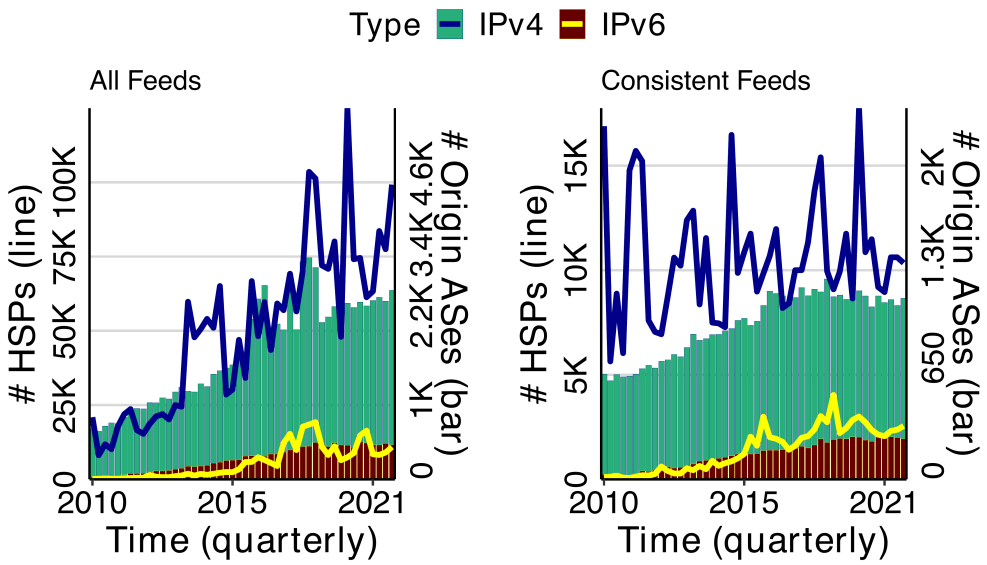


Figure 4.5: Growth of HSPs and HSP origin ASes as visible in all feeder ASes and consistent set of feeder ASes.

observe more IPv4 HSPs due to an increase in feeder ASes at route collector projects.

This hypothesis check leads to another observation: When shrinking the set of feeder ASes, the number of HSPs and their respective origin ASes drops substantially. To improve our understanding of this insight, we analyze the visibility of HSPs, i.e., by how many peers each HSPs is seen. At the same time, we want to understand what causes the substantial fluctuations in the number of HSPs; hence, we also analyze their consistency, i.e., the fraction of time for which the prefix was seen by at least one feeder AS. Given that a one-week observation period would not provide much insight into consistency patterns, we conduct this analysis using data from the entirety of 2020. We first read the RIB snapshots from January 1, 2020 and then apply all updates for the whole year sequentially. By tracking the state of each routing table on a per-update basis, we can extract consistency in seconds granularity.

Figure 4.6 reports the visibility of an HSP on the y-axis against its consistency on the x-axis. For both heatmaps—IPv4 (left) and IPv6 (right)—each cells represents groups of ten feeder ASes on the y-axis and two weeks of time on the x-axis. We first observe that there is no particular consistency trend: While some HSPs can only be observed for less than two weeks, others can be observed throughout the entire year. Our second observation is that the vast majority of hyper-specific prefixes can only be observed by a small number of collector peers, although we do also observe HSPs being visible during the entire year by hundreds of peers. This observation aligns with the restricted propagation characteristics of HSPs reported by previous blog posts [4, 5, 481] and observed by our own active experiments (an in-depth description of the experiments, their analysis, and subsequent results can be found in Appendix B.2.4). We hypothesize that the substantial fluctuations in the number of totally observed HSPs is a result of these two observations; the restricted propagation of HSPs might inflate the importance of the individual placement of feeder ASes and HSP origin ASes, and the tens of thousand of short-lived HSPs might cluster around certain real-world events, such as DDoS attacks or data center outages.

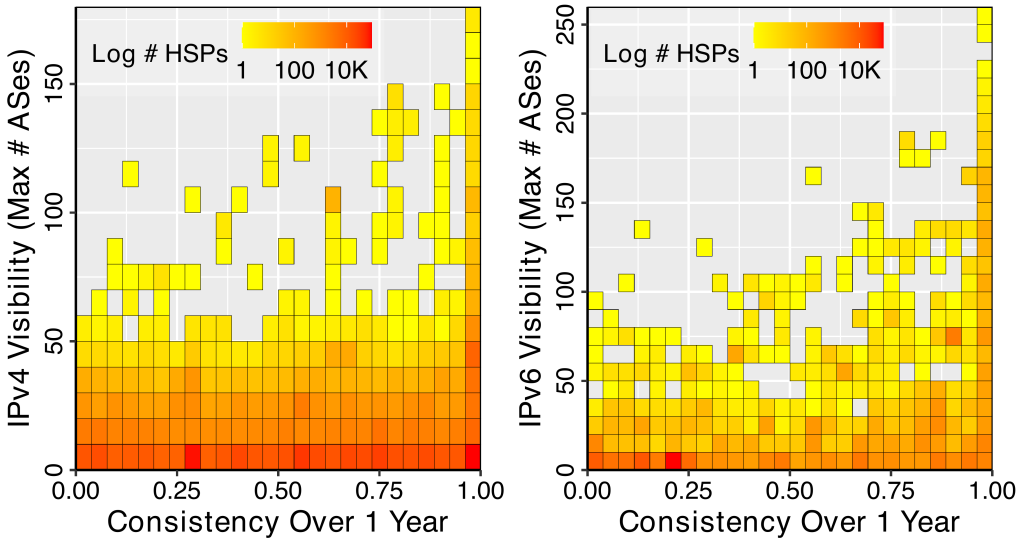


Figure 4.6: Heatmap showing HSP visibility and consistency for IPv4 (left) and IPv6 (right).

In summary, we observe that the presence of hyper-specific prefixes in the Internet's routing ecosystem has increased through the last decade and HSPs make up about one-tenth of all the prefixes that are observed by route collectors. In IPv4 the increase in HSPs is driven by an increment in feeder ASes, whereas in IPv6 we see an increase also for a constant set of feeder ASes. While most HSPs only propagate locally, some of them are globally visible and can be consistently observed throughout an entire year.

4.2.2 Use Cases & Functions

Given their past and current presence in the global routing table, we want to get deeper understanding of the functions that hyper-specific prefixes potentially serve. As a first step in this direction, we use the fact that specific CIDR sizes often hint towards certain use cases. Consider the following example: If an AS wants to defend one of its servers against an ongoing DDoS attack, it may use blackholing announcements. Up to 98 % of these announcements are /32 (/128) IPv4 (IPv6) prefixes, i.e., they only cover the specific addresses of the attacked servers [148, 149, 187]. Larger CIDR sizes are rarely used for blackholing, as they would impair the services running on non-attacked servers as well, i.e., they would introduce unnecessary collateral damage [338]. Using similar lines of reasoning, we rely on the following associations between CIDR sizes and intended use cases: We associate (1) /25 and /26 IPv4 prefixes with traffic engineering (e.g., selective announcements[100, 396]), (2) /29 and /30 IPv4 prefixes with (Point-to-Point) peering subnets (i.e., the subnets needed to form inter-AS connections) [407], (3) /31 and /32 IPv4 prefixes with blackholing [148, 149, 187], (4) /49 to /64 IPv6 prefixes with address block reassignments [376], and (5) /113 to /128 IPv6 prefixes again with blackholing[60].

Figure 4.7 shows the number of IPv4 (left) and IPv6 (right) HSPs over time colored by

[60] In private conversations a large European IXP confirmed that around 90 % of all blackholed IPv6 prefixes fall into the /113 to /128 range.

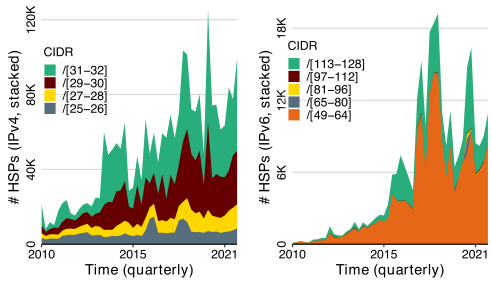


Figure 4.7: HSPs per CIDR size over time.

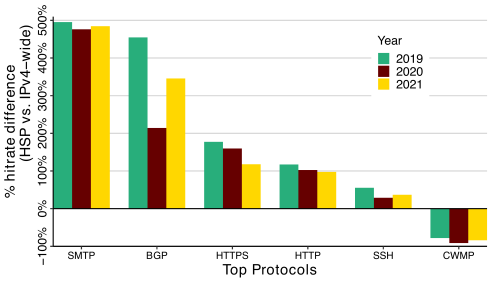


Figure 4.8: Hit rate comparison of HSPs vs. IPv4-wide.

their respective CIDR size groups. We first observe that the overall trends are stable over time. In IPv4, we observe that the most common CIDR size is /31–/32, i.e., the most prominent use case seems to be blackholing. Yet, we also observe that that /29-/30 HSPs are comparably common; hence, many HSPs may actually represent peering subnets. Given that only about 10 % of HSPs have a CIDR size of /25 or /26, we believe that traffic engineering is a rare use case. For IPv6, we mainly observe the /49–/64 CIDR size range that we associate with address block relocations. In some ASes we also observe instances of /64s being used by hypergiants for off-nets [179]. We further observe a small fraction of /113–/128 CIDR sizes that we associate with blackholing. The share of blackholing HSPs is smaller in IPv6 compared to IPv4, which is in line with reports that blackholing in IPv6 makes up less than 2 % compared to IPv4 [187, 338]. Those observations also explain some of the fluctuations that we observed in the previous section—blackholing events, and their subsequently announced prefixes, are often short-lived [338] and subsequently can cause substantial changes in the number of unique HSPs seen throughout a week. As our CIDR-based analysis only provides us with hints on the actual usage, we now also analyze the services hosted in hyper-specific prefixes. For this analysis, we leverage archived scanning data from Rapid7's Open Data platform [401] for 2019, 2020, and 2021. Rapid7 frequently scans the entire routed IPv4 address space[61] for more than 100 well-known TCP and UDP ports. To compare regular with hyper-specific prefixes, we rely on the difference in protocol hit rate, i.e., we compare the fraction of responding hosts and total tested hosts[62] on a per-protocol basis. We observe that four out of the top five protocols with the highest hit rate for regular and HSP prefixes overlap; BGP is only present in the HSP top five while CWMP is only present in the IPv4-wide top five. For those six protocols, Figure 4.8 shows a the relative change of hitrates between regular and hyper-specific prefixes, where a positive value indicates an increase of hit rate in hyper-specific prefixes. While HTTP and HTTPS overall only see an increase of +100 %, we observe strong differences when drilling down on a per-CIDR level: When considering only /32 prefixes, HTTP's hit rate increases by more than +500 % compared to its hit rate for IPv4-wide scans—which substantiates the association of the /32 CIDR size for blackholing. Even more pronounced than HTTP(S), SMTP and BGP see increases of up to +500%. When digging deeper we further observe that BGP is mainly prevalent in /30 and /29 prefixes which substantiates that these sizes might be dedicated to routing infrastructure. In contrast, we observe the only hit rate decrease (of more than 90%)

[61] Except for prefixes on their blocklist which were explicitly requested by network operators.

[62] Given that Rapid7 does not publish the state of their blocklist, we assume that all (at the time of the scan) routed IP addresses were tested. Additionally, we focus on analyzing what services are prominent in HSPs. We can not ensure that Rapid7 (or its upstream) does in fact receive the HSP announcements, as information about their probing vantage points and routing is not available.

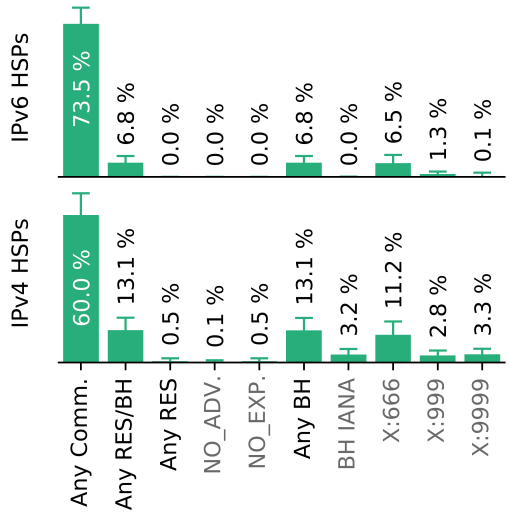


Figure 4.9: BGP communities distribution for HSPs.

for CWMP—a protocol used to remotely manage Customer-Premises Equipment (CPE) devices such as home routers [489].

Finally, we investigate BGP communities attached to HSP announcements. BGP communities are used for many different reasons, such as information tagging, blackholing, route redistribution. The most common BGP communities attached to hyper-specific prefixes are route steering or prepending instructions. In our analysis we look for BGP communities which are specifically used for blackholing (BH) [264] or restrict route propagation (RES)[63]. Figure 4.9 shows the use of BGP communities among HSPs from snapshots between 2019 and 2021. The bars indicate the median share of HSPs with the respective community, the whiskers denote the standard deviation over time. The "Any" keyword is used to specify groups of community targets, e.g., "Any RES" describes all prefixes that have any restriction community attached (i.e., it refers to the union of prefixes with "NO_ADV" community and prefixes with "NO_EXP" communities); similarly, the "Any Comm." bar refers to the highest aggregation, i.e., number of prefixes for which we saw any community attached. As we can see, 60% of all IPv4 HSPs and almost three quarters of IPv6 HSPs come with some form of BGP communities. The vast majority of these communities is, however, not related to blackholing or restricting propagation. Only about 13% and 7% of prefixes can be associated with blackholing for IPv4 and IPv6, respectively. The by far most popular blackholing community is `X:666`. Moreover, we see no propagation restriction communities (“no advertise” or “no export”) in IPv6 and only about 0.5% in IPv4. Furthermore, we see that RES communities are a subset of BH communities, hinting that operators do not want their blackholing prefixes to propagate. Blackholing is therefore one contributor of HSPs, but blackholing communities are not present on the majority of HSP announcements. We note that the blackholing communities that we see at route collector peers is a lower bound: Blackholing communities—similar to other communities—could be cleaned along the path but the prefix itself could continue to propagate [267].

In summary, we observe that for IPv4 many and for IPv6 some HSPs are likely related to

[63]We also test for communities such as NOPEER or NO_EXPORT_SUBCONFED, but these are not prevalent among HSPs.

blackholing activities. While we also observe many HSPs dedicated to routing infrastructure (e.g., peering subnets or address relocations), we observe that hyper-specific prefixes rarely contain any CPE devices.

4.2.3 Intended or Accidental Use?

Now that we have a basic understanding of the use cases of HSPs, we want to analyze whether HSPs are used intentionally or accidentally by ASes and their operators. If operators take the time and effort to explicitly enter hyper-specific prefixes into voluntarily-maintained databases, then it is likely that they plan to use them. Hence, we look at the Resource Public Key Infrastructure (RPKI) and Internet Routing Registry (IRR) operator databases.

We use private, three-monthly IRR snapshots [242] between January 1, 2017, and October 7, 2021, which contain information about routing policies. The RPKI database contains legally binding mappings between Internet resources and ASes. We use daily snapshots of the RPKI database [432] from April 1, 2015, until October 7, 2021, generated by Chung et al. [108] to verify the validity of HSP announcements by ASes.

While we extract HSPs directly from the `route(6)` objects contained in the IRR databases, the Route Origin Authorization (ROA) objects in the RPKI snapshots describe CIDR size ranges [222]. Hence, a ROA can explicitly describe an HSP when both the minimum and maximum prefix length are hyper-specific, or implicitly when only the maximum prefix length is hyper-specific. When extracting HSPs and their origins from the RPKI database, we rely solely on explicit definitions as these clearly represent the desire to use HSPs (as all covered prefixes are hyper-specific). As implicit definitions might describe the future—but not necessarily current—use of HSPs (e.g., an AS might currently announce a /24 but has already entered a currently unused max-length of /25), we decided to ignore them. We compare the HSPs on those two databases against the HSPs visible via BGP route collectors.

Figure 4.10 shows the number of unique origin ASes for both IPv4 and IPv6 within each dataset over time. We classify those origin ASes available in more than one dataset into the "Multiple" category. Our first observation is that for both IPv4 and IPv6, the IRR dataset contains the largest fraction of HSP origin ASes. While this might imply that network operators tend to actually use HSPs, it is well-known that route objects can become stale given that the database is only maintained on a voluntary basis [462]. Yet, Some entities, e.g. certain IXP Route Servers [133], require route objects in the IRR database to redistribute prefixes (i.e., HSPs). Yet even for the RPKI database we observe hundreds of explicitly defined HSPs[64]. Notably, for the last snapshot in October 2021, implicit HSPs would have increased the number of RPKI origin ASes from 294 to 990 for IPv4 and from 172 to 794 for IPv6, respectively. Beyond these intentional HSPs, we also observe that many of the HSPs from Route Collectors have no entries in operator databases, hence, they could potentially represent accidental announcements or misconfigured route collector sessions that leak internal routes.

While it is hard to link malicious intent to a more-specific announcement (since it could be, e.g., an address leasing agreement [391] or traffic engineering of sibling ASes [175]), we want to understand if the visible HSPs in the BGP are legitimate prefix advertisements by valid origin ASes or associated with possible prefix hijacks. Therefore, we perform Route Origin Validation of HSPs and its origin AS by checking them against the ROA

[64] Most of these HSPs are also in the BGP data set and hence end up up in the multiple class.

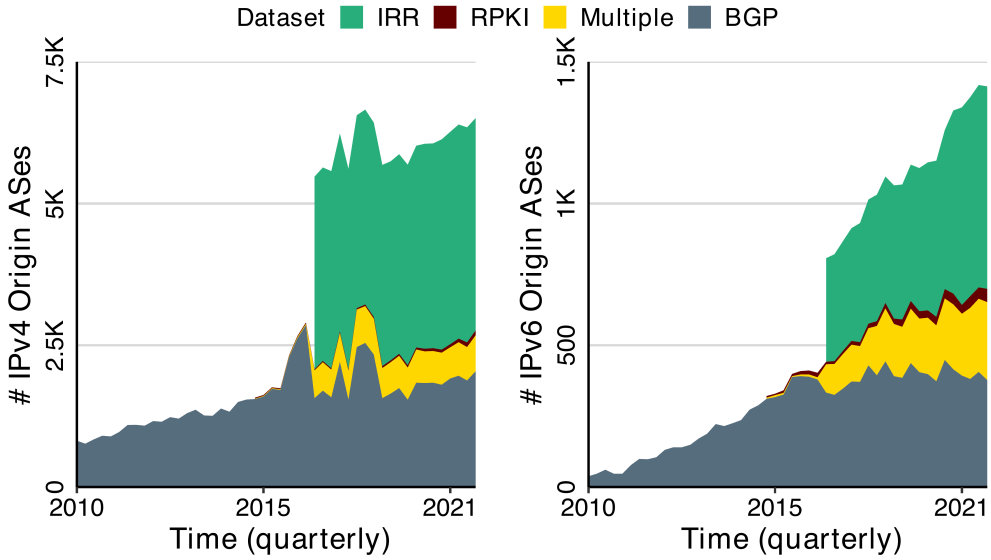


Figure 4.10: Visibility of origin ASes across data sets.

records from the RPKI dataset. If a ROA covers the address space described by the prefix, then this prefix can violate the ROA in two ways: it can be too specific—which we mark as "Invalid (Length)"—and it can be announced by a different origin—which we mark as "Invalid (Origin)." If both of these conditions are met at the same time, we mark a prefix as "Invalid (Both)." If none of these conditions are met, we consider the prefix as "Valid." Notably, we observe that 22 % of IPv4 and 19 % of IPv6 HSPs have a covering ROA entry (median percentages across snapshots in 2020 and 2021).

Figure 4.11 shows that legitimate ASes, i.e., the valid and invalid length categories together, advertise around 75 % of all HSPs. With an average of 25 % peaking to around 50 % in 2016, 2017, and 2019 IPv6 has a higher percentage of valid HSPs than IPv4. The HSPs with invalid length form the largest group in IPv4, and mostly the second largest group in IPv6. The third largest group of HSPs has the "Invalid (Both)" ROV state, while the invalid origin category forms a minor fraction of HSPs' ROV state. Legitimate ASes advertise around 75 % of HSPs, which indicates that HSPs are not majorly associated with BGP prefix hijacks. Beyond malicious ASes, the "Invalid (Origin)" and "Invalid (Both)" status could also be caused by not properly entered sibling ASes [175].

In summary, we observe that for both IPv4 and IPv6, hundreds of ASes intentionally entered hyper-specific prefixes into operator databases. Yet we also saw that many of the HSPs that are visible from route collectors have no respective entries and are likely related to the accidental announcement or disclosure of internal routes. This is further substantiated by the observation that most HSPs are actually ROV invalid since they are more specific then intended by their covering ROA entry.

4.2.4 Discussion

Research Community. While many HSPs seem to be intentional, we also observe a large number that potentially represent leaked internal routes. While the task of reconfiguring a leaking router ultimately belongs to the feeder AS' operators, we believe

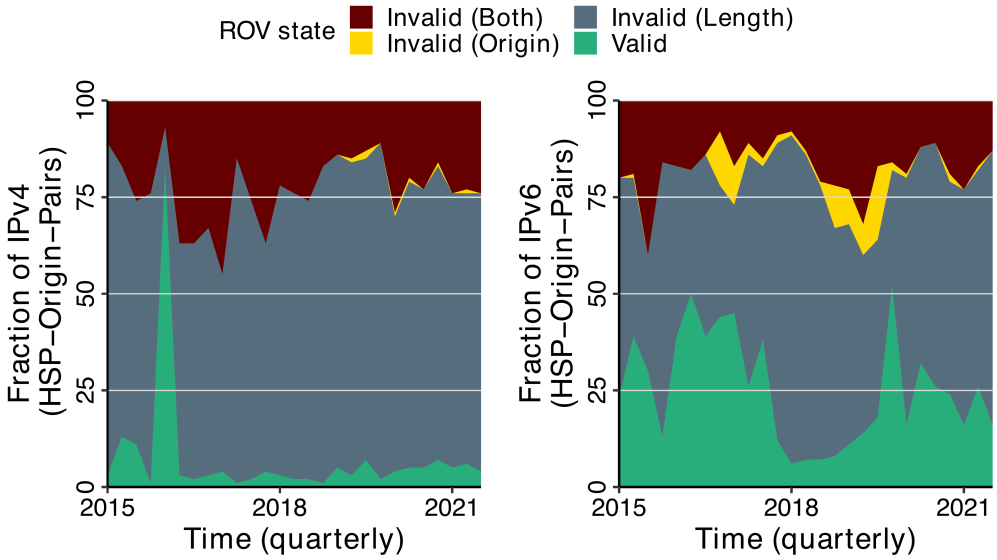


Figure 4.11: ROV status for HSPs

that the maintainers of route collector projects play a vital role when it comes to raising awareness for the existing problems. To support and guide this process, we plan to maintain a dashboard that provides up-to-date HSP statistics as well as a rankings of the top HSP contributors. Beyond fixing potential leakage errors, we believe that studying the potential correlations between hyper-specific prefixes and their less-specific counter parts may lead to new insights into the routing optimizations used by ASes.

Operator Community. Even though various guides [153, 156, 306, 356, 359] recommend strict filtering of HSPs, we observed that many hyper-specific prefixes propagate to 100 or more collector peers. After discussing our results with thirteen operators from different types of networks, we believe that the limited filtering is often a result of popular customer requests. The operator of a major transit network told us that his network recently (throughout Summer 2020) changed from the filtering of all IPv4 HSPs to only filtering prefixes more specific than /28; this shift enabled (especially new and small) customer networks to perform basic traffic engineering despite a limited address allocation[65].

This opens up the question whether operators should filter HSPs in the first place. We believe that for IPv6 the answer is a resounding "yes". Given that there is no shortage of IPv6 addresses and obtaining new blocks is virtually free (compared to the high costs of obtaining IPv4 addresses), we do not see any reason to loosen the current filtering guidelines. For IPv4, we think that the answer should be more nuanced. While loosening the filter guidelines allows even small ASes to perform traffic engineering, it would also further increase the routing table size. Hence, we believe that shifting the acceptable boundaries by a few CIDR sizes (e.g., /26 or /28) might be an agreeable compromise.

[65]This is a direct result of the current IPv4 Address exhaustion and the subsequently inflated prices [391].

4.2.5 Related Work

In this section, we report on related work in the areas of hyper-specific prefix analysis and prefix deaggregation.

HSP Analysis: Previous research in this area consists mostly of blog posts. In 2014, Aben and Petrie report on an experiment where they announced /24, /25, and /28 IPv4 prefixes and ran RIPE Atlas measurements to them [4]. Their findings show that HSPs are visible for at most 20 % of RIPE RIS peers [349] with route objects slightly improving the visibility. The RIPE Atlas experiments lead to similar results with fewer than 15 % of probes reaching their targets. One year later, Aben and Petrie revisit the propagation of hyper-specific prefixes and find a marginal increase of a few percent [5]. In 2017, Strowes and Petrie conclude that not much has changed regarding hyper-specific prefix propagation and at most one fourth of all BGP peers receive those announcements [481].

Prefix Deaggregation: In 2002, Bu et al. first characterize prefix deaggregations and the reasons for them, e.g., traffic engineering, multi-homing, and address fragmentation [80]. Meng et al. report in 2005 that even newly assigned address space is deaggregated and that the deaggregation rate of prefixes increases over time [324]. In 2010, Cittadini et al. [120] report that more than 10 % of ASes deaggregate their prefixes while around 1 % of ASes announce more than 10 prefixes for each address block they got assigned. Lutu et al. present a simulation model that estimates that origin ASes can reduce their transit cost by 5 % by using more-specific announcements [294–296]. Notably, the authors neither focused on IPv6 nor on hyper-specific prefixes. In 2016, Krenc and Feldmann analyze the address delegations realized via prefix deaggregations and report on delegations from customers to providers or between unrelated ASes (often involving CDNs) [268]. In 2017, Huston analyzes the prevalence and different types of more-specific prefix announcements in the Internet as an effect of prefix deaggregation [227]. His taxonomy attributes MSPs to three different root causes, hole punching (different origin AS), traffic engineering (same origin AS, but different AS path), and overlay (same AS path). He concludes that the former two play a useful role for network operators, while the usefulness of overlay more-specific prefixes could be argued about. Huston did not specifically investigate the effect of hyper-specific prefixes.

To the best of our knowledge, this section presents the first scientific analysis of hyper-specific prefixes by providing an in-depth look into the prevalence and possible root causes for HSPs in the wild.

4.2.6 Conclusion

In this section, we analyzed the presence of hyper-specific prefixes in the Internet's ecosystem throughout the last decade. While we found an overall increase in the number of HSPs, most of them can only be observed by a few route collector peers. Yet, there are still plenty of HSPs that propagate to hundreds of route collector peers and can be consistently observed throughout an entire year. Inspired by those findings, we took a closer look at the function that these prefixes serve. For IPv4, we observed that HSPs are mainly associated with blackholing and infrastructure announcements (e.g., routes to peering subnets). While we only found limited evidence for any connection to traffic engineering, we observed that hyper-specific prefixes are less likely to contain end-user devices. For IPv6, we observe that almost all hyper-specific prefixes are related to address block reassignments, with only a small fraction representing blackholing. Even though we have seen that hundreds of networks use HSPs intentionally, we attributed even more

cases to the accidental "leakage" of internal routes. Finally, we discussed the current state of HSPs from an academic as well as an operator-related review.

4.3 Chapter Summary

In this section, we described the IPv4 exhaustion state in 2020, took a look at the emerged IPv4 buying and leasing markets, and analyzed the viability of hyper-specific announcements.

While AFRINIC and APNIC still have a small number of addresses left in their allocation pools, the three other RIRs introduced waiting lists with current waiting times of at least 300 days. As a result, networks that are still relient on IPv4 enter the vibrant leasing and buying markets. Although prices for IPv4 addresses doubled between 2016 and 2020 as well as between 2020 and 2022, the actual price is still lower than previous work predicted. While the prices per IP decrease with address block sizes (except for large blocks), we find no statistically significant difference between regions despite their varying allocation policies. We observe that the buying market has been volatile since 2016 but went into a consolidation phase at the beginning of 2019. When inferring IPv4 leasing agreements from BGP announcements, we found that the amount of leased IPv4 addresses increased by 7% between 2018 and 2020. Yet, at the same time we showed that state-of-the-art delegation inferences are noisy and only reveal a small fraction of the actual delegations. While a completer picture could be drawn by incorporating RPKI and RDAP data, not all RIRs provide an RDAP interface. We further find that the leasing market for IPv4 addresses had not converged in 2020 as prices per IP per month (within a /24) range between $0.30 to $2.33.

We compared the visibility of hyper-specific prefixes within BGP route collectors, the RPKI ecosystem, operator databases, and active measurements. While we find thousands of ASes announcing hyper-specific prefixes, we observe that only few of these prefixes propagate globally. For IPv4, we observed that HSPs are mainly associated with blackholing and infrastructure announcements (e.g., routes to peering subnets). We only found limited evidence for any connection to traffic engineering, yet found that hyper-specifics are less likely to host end-user devices. For IPv6, we observe that almost all announced hyper-specific prefixes are related to address block reassignments, with only a small fraction indicating blackholing activities.

Discussion. While the leasing market, at face value, appears to be an option for only temporary resource use, it takes multiple decades to amortize a similarly sized IPv4 prefix when currently buying it. Hence, leasing IPv4 addresses to bridge the gap until full IPv6 adoption may be economically cheaper than actually buying IPv4 resources. At the same time, the increased demand and subsequently rising prices for IPv4 addresses may be a future driver for accelerated IPv6 adoption. In the meantime, ASes may further announce their existing IPv4 address space as hyper-specific routes. While these routes are not frequently redistributed, they may be used in combination with less-specific covering prefixes to ensure global reachability. As many (especially larger) ASes accept them locally (despite not redistributing them), hyper-specific prefixes may be effective for traffic engineering across immediate neighbors. The ongoing flattening of the Internet's hierarchy and the subsequent increasing of peering partners makes this characteristic even more attractive over time.

Chapter 5
Securing Routing Operations

The previous chapters focused on how to track and navigate parts of the Internet routing ecosystem's evolution. In this chapter, we will take a closer look at how an evolving routing ecosystem may affect security aspects.

We start this chapter with an explicit demonstration of how parts of the routing ecosystem changed throughout the last decade, and how these changes affect the safe use of a popular, well-known, and easy-to-use inter-domain traffic engineering technique called AS Path Prepending. By combining more than a decade of passively obtained routing information with active, real-world experiments, we measure the prevalence of prepending in the wild, identify typical usage patterns, assess the security risks associated with these patterns, and quantify the number of risk-exposed ASes.

After this initial demonstration, we want to stress the potential impact that the ecosystem's evolution has on routing security. In particular, our focus lies on two specific changes: the rising deployment and viability of IPv6 and the increasing availability of low-cost peering opportunities. We analyze the synergies between these changes and prefix de-aggregation attacks and find that they allow adversaries to overcome previously deployed prevention mechanisms.

The contributions of this chapter can be summarizes as follows:

- We perform a longitudinal characterization of ASPP usage and identify a steady increase. On May 2020, 30 % of ASes prepend at least one of their prefixes, resulting in 25 % of the IPv4 prefixes being originated with prepending. We further find that ASes mainly originate their prefixes with two distinct prepending sizes (e.g., without prepend and with two extra prepends) to indicate their preference for inbound traffic. Surprisingly, we also find that roughly 6k ASes originate a total of more than 28k prefixes with a single prepending size (different than zero), thus resulting in no traffic steering effect.
- We discover that in scenarios with only two upstreams, ASPP effectiveness is strongly dependent on the vantage point. Yet, when using many diverse upstreams, ASPP shifts traffic from most incoming sources.
- Using active experiments, we identify that prefixes with three or more prepends

are highly vulnerable to prefix hijacking. Today, there are more than 15k prefixes with at least three prepends, increasing the risks of widespread route leaks or prefix hijacking without apparent traffic steering benefit.

- We revisit prefix de-aggregation attacks and analyze their theoretical and practical viability using a mix of a theoretical Integer Linear Programming formulation, the deployment of a real-world testbed, various BGP data analyses, real-world route propagation measurements, and router testbed experiments.
- We find that the routing ecosystem's evolution weakened previous prevention mechanisms to the point where prefix-deaggregation become viable on the Internet today, by anyone, and with a limited budget. Hence, we propose and extensively discuss possible defense mechanisms and perform a two stage vulnerability notification campaign involving 8 major IXPs, 20 Tier-1 ASes, and 7 major content providers.

5.1 AS Path Prepending

Many Internet Autonomous Systems (ASes) receive significantly more traffic than they send. They often use Inbound Traffic Engineering (ITE) to influence the link through which they receive traffic based on economic considerations (e.g., transit cost) or operational demands (e.g., latency, packet loss, capacity). ITE has become even more important, as there are more options for inter-AS connectivity due to, e.g., IXPs (Internet eXchange Points), PNIs (Private Network Interconnects), and an overall increase of peering [71, 447, 512, 518, 519]. Border Gateway Protocol (BGP)-enabled ITE techniques include AS path-prepending (ASPP) [100, 146, 523], selective or more-specific prefix announcements [174], BGP communities [151, 478], or Multi Exit Discriminator (MED) values [164, 321].

In this section, we focus on understanding ASPP deployment and the potential issues associated with it. ASPP is a straightforward, easy-to-use technique that is often mentioned among the first ITE techniques by router vendors [114, 138, 173, 253, 328]. It is a technique where an AS artificially inflates the BGP AS path by inserting (subsequent) duplicate entries of its ASN. Since the length of an AS path is the second most important tie-breaker in BGP best path selection, ASPP may steer traffic from one route to another. However, its effect depends on route propagation and the routing decisions made by other ASes. Despite (or because of) its simplicity and its inherent limitations, the appreciation of ASPP among operators and researchers is mixed. On the one hand, ASPP—unlike other ITE techniques—does not need any support from other ASes, nor deaggregatable prefixes. On the other hand, its need, effectiveness, and predictability have been questioned [298, 395, 485]. In addition, there have been concerns about the extent to which ASPP can amplify existing routing insecurities [299, 309, 479], and reports of improper ASPP configurations triggering bugs in router software [529, 530].

Motivated by the mixed views about the ASPP method, we investigate the current use of ASPP and find that more than 30% of ASes use it. Thus, to contribute to an informed discussion, we address three fundamental questions:

(1) **How do ASes use prepending?** To put effectiveness and risk into context, we first identify and characterize the *policies* ASes apply (i.e., the number of prepends used for each prefix) when using ASPP. Even when using data from all route collectors over the last decade, limited route visibility [103, 194, 365] poses a significant challenge. We deal

with it by conducting interviews with more than 20 operators and by cross-checking our results with private data sources from large Internet players.

(2) **How effective is prepending?** Among both operators and academics, the opinion on whether ASPP is effective as an ITE technique diverges and often depends on the position of an AS in the routing ecosystem. For example, Quoitin et al. [395] showed that ASPP is unpredictable using their vantage point. We claim that the effectiveness of ASPP is indeed diverse—it depends on the vantage point within the routing system and the number of available upstreams. We highlight this behavior by actively testing a large number of vantage points and varying the number of upstreams.

(3) **Does prepending amplify existing routing security risks?** Often, a "malicious" route needs to be the shortest path in order to be adopted. ASPP facilitates the spreading of malicious routes by making the legitimate paths longer. While one may observe malicious routes in public BGP data, the lack of suitable what-if scenarios (i.e., how would the scenario change with a larger prepend size) poses a significant challenge. We shed light on this topic by systematically emulating numerous prefix hijacks from many vantage points.

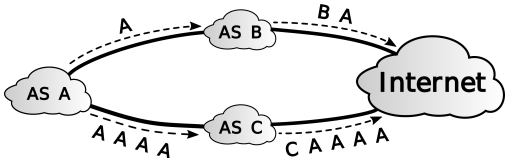
We approach these questions using both active and passive measurements. We use passively collected routing information from Isolario [244], RIPE RIS [349], and Route-Views [364] to perform a longitudinal study. We then use the PEERING testbed [444, 448] to systematically explore ASPP from a large number of vantage points and emulate many scenarios through targeted BGP route announcements and probing traffic.

We summarize our main contributions as follows:

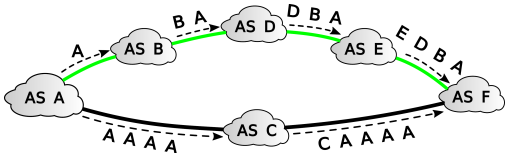
- We perform a longitudinal characterization of ASPP utilization and identify that, despite the community mixed opinions, its utilization has been steadily increasing. We find that, on May 2020, 30 % of the ASes prepend at least one of their prefixes, resulting in 25 % of the IPv4 prefixes being originated with ASPP (see § 5.1.3).
- We also identify that ASes mainly originate their prefixes with two distinct prepending sizes (e.g., without prepend and with two extra prepends) to indicate their preference for inbound traffic. Surprisingly, we also find that roughly 6k ASes originate a total of more than 28k prefixes with a single prepending size (different than zero), thus resulting in no ITE effect (see § 5.1.4).
- We discover that in scenarios with only two upstreams, ASPP effectiveness is strongly dependent on the vantage point. Yet, when using many upstreams, ASPP shifts traffic from most incoming sources (see § 5.1.5).
- Using active experiments, we identify that prefixes with three prepends are highly suitable for prefix hijacking. Today, ASes originate more than 15k prefixes with at least three prepends, increasing the risks of widespread route leaks or prefix hijacking with no apparent ITE benefit (see § 5.1.6).

We discuss ethical considerations in Appendix § 5.1.9, and to foster reproducibility and research on ASPP, we make all of our analysis code available to the research community.[66]

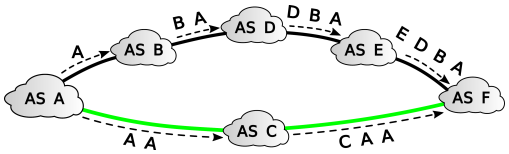
[66] `https://gitlab.mpi-klsb.mpg.de/lprehn/imc20_aspp`



(a) AS A announces the prefix P with ASPP to one of its neighbors.



(b) ASPP in effect.



(c) Route adoption regardless of ASPP.

Figure 5.1: AS-Path Prepending behavior.

5.1.1 Primer on Path Prepending

ASPP is an ITE technique in which an AS adds its own AS number n extra times ($n \geq 1$) before originating/propagating a BGP route, thus artificially increasing the resulting AS-Path length by n. We refer to n as the *prepend size*. Whenever an AS receives a route announcement, it chooses the best path according to a list of tie-breaking rules. The first rule relies on local preference. To affect the route selection of remote ASes [405], an AS uses ASPP to inflate the AS Path to influence the second tie-breaking rule: to prefer the shortest AS path. (If the tie persists, route origin and MED values are among the remaining tie-breakers.)

In Figure 5.1(a), we illustrate the use of ASPP by an AS with two neighbors. AS A announces a prefix P to both neighbors with different prepend sizes. By making one path longer, AS A attempts to influence remote ASes to send traffic through AS B. The success of this attempt will depend on how remote ASes will receive the announcements. In Figure 5.1(b), we depict a case where ASPP can influence the decision of AS F. Even though the path traversing AS C has fewer ASes than the one going through AS B, AS F prefers the second path as it is the shortest. In Figure 5.1(c), we show a case where the ASPP by AS A cannot influence the decision of AS F as it has fewer prepends—AS F prefers the path traversing AS C as it has the smallest AS path length. These cases underline that ASPP cannot guarantee remote route changes and the resulting ingress traffic distribution.

We distinguish two forms of prepending. If the AS prepending is the originator, we refer to it as *origin-prepending*; otherwise, we refer to it as *intermediate-prepending*. When an AS prepends on behalf of another AS, we refer to this particular form of intermediate-prepending as *remote-prepending*. In such cases, ASes can use BGP communities or web interfaces to ask the other ASes to prepend. ASes use remote-prepending to affect path choices that are beyond the reach of origin-prepending.

ASes can use ASPP for *load balancing* among upstreams, to *minimize transit cost* (by moving traffic away from an expensive upstream), or to establish *backup* links. Among the reasons mentioned by operators for ASPP popularity are its ease of use on commercial routers, its efficiency in steering incoming traffic, and the requirements and shortcomings of alternate mechanisms.

5.1.2 Data sets and Data Sanitation

To analyze ASPP utilization, we rely on (BGP) MRT data publicly available from Isolario[67] [244], RIPE RIS[349], and Route Views [364]. We use the following datasets in our analyses.

$BGP_{Continuous}$: This dataset contains RIB snapshots from all available BGP collectors on March 1st, 2020 at 0:00 UTC+0. In addition, it contains all subsequent update files until April 1st, 2020, at 0:00 UTC+0. If an update file is missing in a collector's repository, we add the next available RIB snapshot to capture potentially missed changes.

BGP_{Weekly} : This dataset contains data for each Monday between January 1st, 2018, and May 4th, 2020. For each day, we use the RIB snapshots from all available BGP collectors at 0:00 UTC+0 and all consecutive updates for that day. We, again, compensate for missing files.

$BGP_{Monthly}$: This dataset contains data for the 15th day of every month between January 15th, 2010, and April 15th, 2020. We generate the data of a single day in the same way as for the previous dataset.

ROAS: Rather than using tools (such as Routinator [355]) to preprocess RPKI data, we take advantage of the preprocessed data provided by Chung et al. [108]. We use data for the same days as in the BGP_{Weekly} dataset.

RIR: This dataset contains the (extended—if available) delegation files from AFRINIC [8], APNIC [25], ARIN [37], LACNIC [271], and RIPENCC [415] for all days in the $BGP_{Monthly}$ dataset.

Data sanitation. Before analyzing our BGP data, we remove well-known artifacts. First, we remove bogon routes, i.e., routes that lead only to reserved address space [486] or routes that contain ASes currently reserved by IANA [229]. Similarly, we remove all routes to prefixes less specific than /8. This step ensures that we only analyze default-free routing information.[68] We further remove all routes for which the path contains a loop. The sanitation, up to this point, removed ~3.36M (0.7 %) routes and reduced the number of prefixes from ~1.29M to ~932k (-28 %) using the last snapshot of the BGP_{Weekly} dataset as reference (we find similar values for other snapshots). To avoid making false inferences due to lack of visibility, we only analyze prefixes visible by at least one-third of the BGP monitors set on the corresponding date. When analyzing how many monitors see each prefix, we find a clear separation between locally and globally visible prefixes regardless of the exact year (in Appendix § C.1.2 we report more details). Notably, the last step reduced the number of unique prefixes to ~803k.

5.1.3 Trends in the Use of ASPP

Previously reported metrics about ASPP differ across studies, with the most recent results being from 2016 [77, 164, 174, 503]. To understand ASPP utilization better, we analyze its trends over the last decade using the $BGP_{Monthly}$ dataset. We note our numbers represent lower bounds of the actual ASPP utilization, as (1) the visibility of route collectors is

[67] Isolario was hit by lightning on July 30th, 2019, leading to some missing files until August 16th, 2019 —we find that the impact to our analysis is minimal.

[68] As opposed to cases in which an AS uses the default route (i.e., 0.0.0.0/0) to send traffic to some/all destinations.

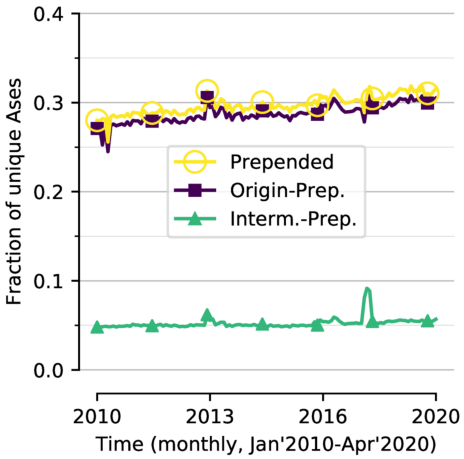


Figure 5.2: Fraction of ASes deploying ASPP.

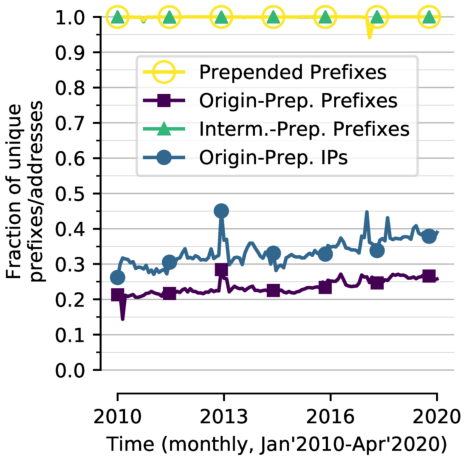


Figure 5.3: Fraction of Prefixes/IPs with ASPP.

limited [103, 194, 365]; (2) prepended paths tend to be less attractive than non-prepended ones; (3) we sanitize our data (see § 5.1.2).

One-third of all ASes use origin-prepending. Figure 5.2 shows the fraction of ASes using ASPP (for IPv4) separated by prepending type (recall § 5.1.1): *origin-prepending* or *intermediate-prepending*. First, we see that the fraction of ASes using ASPP has increased slightly, from ~28% (9.4k) on January 15th, 2010 to ~31.4% (21.6k) on April 15th, 2020, with most ASes using origin-prepending. Similarly, we observe a small increase in intermediate prepending—from 4.7% (1.6k) on January 15th, 2010, to 5.5% (3.8k) on April 15th, 2020. [69] We also see a very small fraction (<1%) using only intermediate prepending, some of which might be due to ASes offering remote-prepending, e.g., via BGP communities.

The fraction of prepended prefix-origin pairs and addresses has increased slightly. Next, we focus on prefixes. We consider a prefix/IP address as prepended if at least one AS has added its ASN more than once (consecutively) to the path. In Figure 5.3 we observe that the increase of prefixes with origin-prepending is similar to the one observed respective to ASes—from ~21.3% (65.2k) on January 15th, 2010 to ~25.9% (207.7k) on April 15th, 2020.[70] Regarding the intermediately prepended prefixes, we observe that for the entire $BGP_{Monthly}$ dataset, (almost) all prefixes contain prepend in all snapshots. Such a condition happens because there are transit ASes (especially Tier-1s) that prepend most prefixes before redistributing them to at least one of their neighbors.

For IP addresses we see a larger increase of origin-prepending—from ~26.2% (570 million) in January 15th, 2010 to ~38.9% (1.1 billion) on April 15th, 2020. This more pronounced increase is likely correlated to the exhaustion of the IPv4 address space and the fact that prefixes more specific than /24 tend to propagate less [384].

[69]The spike on the fraction of ASes applying intermediate prepending corresponds to the period in which a set of experiments [478] involving the use of BGP communities to manipulate ASPP was taking place.

[70]We also analyzed ASPP growth considering only monitors available on January 15th, 2010 and find similar behavior.

To not overestimate the numbers, we check how many of these addresses are covered by more-specific non-prepended prefixes.[71] We find that on April 15th, 2020, only 9.3% of the origin-prepended address space was reachable through more-specific non-prepended announcements. We also observe that for 79% of the cases, the less specific announcements were visible in more monitors than the more-specific ones, indicating that 9.3% may be an over-estimate.

Discussion. Despite various public call-outs of the drawbacks of ASPP [298, 299, 309, 479], we observe that its use has not decreased. Most operators were surprised by the results. According to them, the long-term use of ASPP is a sign of either bad capacity planning or inexperienced network engineers. Nevertheless, some argued that many factors might render ASPP more attractive than other ITE techniques for network operators, including not being able to obtain an address space larger than a /24 in certain regions (e.g., RIPE[417, 419]); the capital required to expand the infrastructure; the simplicity of ASPP; and its prominence in router vendor handbooks.

Focus on origin-prepending and IPv4. As the large number of intermediately prepended prefixes is the result of the routing policies of a small number of large ASes (e.g., Tier-1s), for the remainder of this section, we focus on the (far more common) origin-based prepending. Also, we choose to focus on IPv4 prepending, as IPv6 accounts for only 6% of the total cases of prepending on April 15th, 2020.

5.1.4 Prepending Policies in the Wild

To understand how operators use ASPP with the prefixes they originate, we identify different *policies* and look at their prevalence in-the-wild, both in terms of prefix-origin pairs (§ 5.1.4.1) and ASes (§ 5.1.4.2). In our analyses, we find a surprising incidence of a seemingly innocuous form of ASPP, called *uniform* prepending, which we thus investigate more closely (§ 5.1.4.3). Last, we examine the evolution of prepending sizes in different geographic service regions (§ 5.1.4.4). As in the previous section, we consider only prefix-origin pairs visible by at least one-third of all BGP monitors.

5.1.4.1 ASPP Policies: Prefix-Origin Pairs

We identify four different prepending policies that can be used in a prefix-origin pair. They are (1) *no-prepend*: no visible prepended route; (2) *uniform*: the only visible prepend size is N, where $N > 0$; (3) *binary*: visible routes either have prepend size M or N, where $M, N \geq 0$ and $M \neq N$; (4) *diverse*: the number of different prepend sizes in the visible routes exceeds two.

ASes tend to stick with a (per-prefix) policy over time. Our first focus is on policy *consistency*—how often does an AS change a prefix prepending policy? For this analysis, we use the $\text{BGP}_{\text{Continuous}}$ dataset, in which we identify roughly 2.3 million unique prefix-origin pairs. For each pair, we define as its *primary policy* the one we observe more often throughout the full month (among *no-prepend*, *uniform*, *binary* and *diverse*). We examine the stability of the primary policy for a prefix-origin pair with respect to its visibility period. Figure 5.4 shows a heatmap, where colors indicate the number of pairs in each cell. We observe a concentration in the top right section of the plot, which corresponds to 54% of prefix-origin pairs, indicating that they are visible all the time and never change

[71] Recall BGP longest-prefix-matching prefers routes for (non-prepended) more-specific prefixes over (prepended) less-specific ones.

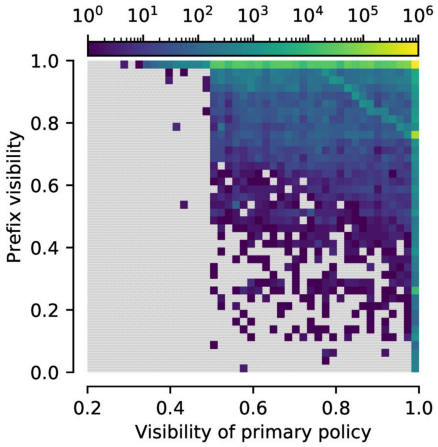


Figure 5.4: Prefix-origin primary policy consistency across a month.

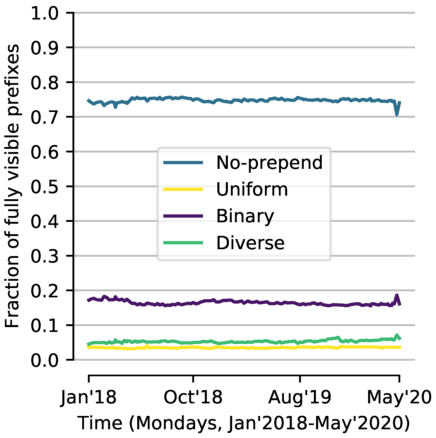


Figure 5.5: Prefix-origin: Fractions through time of visible prefixes per ASPP policy.

their primary policy. We repeated the analysis for another month (Sep. 2019) and found similar primary policy stability, which allows us to adopt weekly (BGP_{Weekly}) or monthly ($BGP_{Monthly}$) snapshots without any loss in the subsequent analyses.

The use of policies has been stable, with *binary* policy being more common. Using the BGP_{Weekly} dataset, we examine the use of prepending policies for prefix-origin pairs between January 1st, 2018, and May 4th, 2020. In Figure 5.5, we see that the most common prepending policy is *binary*, followed by *diverse* and *uniform* policies. Their popularity remains largely stable, if considering the proportion to the full set of pairs: *diverse* increased from 4.5% (30k) to 6.1% (50k), *binary* decreased from 17.2% (114k) to 15.9% (131k), while *uniform* remained at 3.6% (24.4k to 29.4k)[72]. We note that the trend regarding the use of more fine-grained policies might be related to the increasing connectivity level of ASes (e.g., connecting to more IXPs). For the sake of comparison, we looked at the use of *uniform* prepending back in January 2010, and it was 2.7% (8.2k). The consistent presence of *uniform* policy through time is surprising since, in theory, it should not influence any remote BGP decisions. We take a closer look at this phenomenon in § 5.1.4.3.

More prepending during COVID-19 lockdown. We also note that between February and April of 2020, the number of prefix-origin pairs with ASPP reached approximately 30% (4% increase). Such a peak is likely related to the lockdown measures due to COVID-19, which resulted in people staying more time at home [189, 513, 514]. In this period there have been reports of traffic increases [21, 132, 333], which also resulted in content providers such as Netflix and Youtube stopping streaming in 4k to save bandwidth [199].

We believe that the higher use of ASPP during this period was necessary for network operators to handle the increasing demands of traffic while upgrading their links (as ASPP use has decreased in May). When we discussed this with network operators, some of them mentioned that they were observing more use of ASPP, especially during large live events streamed on Youtube [327], and that some of their transit customers were requesting capacity upgrades.

[72]In May 4th, 2020, all types of prepending combined represented 25.6% of all prefix-origin pairs, corroborat-

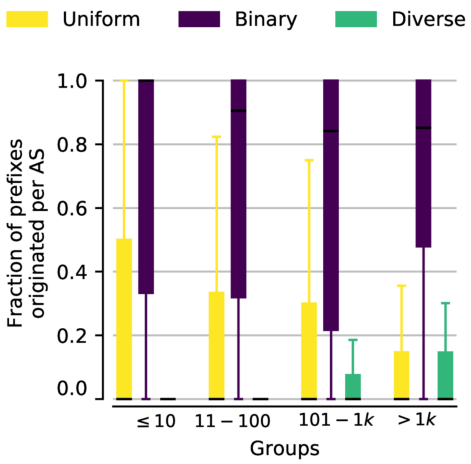


Figure 5.6: *Mixed* policy ASes grouped by # of prefixes.

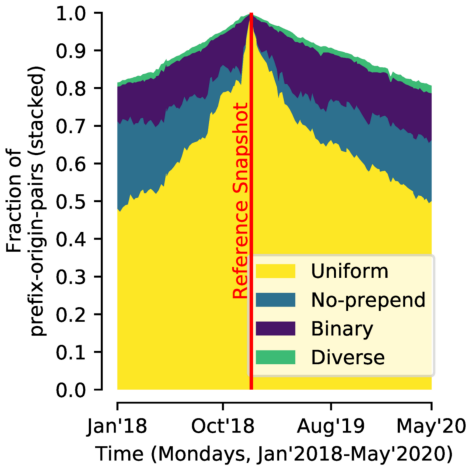


Figure 5.7: Fractions of prepending policies through time for a fixed set of uniform-prepend prefixes.

5.1.4.2 ASPP Policies: ASes

We now change the perspective of our policy analysis to ASes. We differentiate per-AS ASPP policies as follows. When an AS employs a single policy for all prefixes it originates, we say it adopts one of the four policies already defined: *no-prepend*, *uniform*, *binary* or *diverse*. Otherwise, we say an AS employs a *mixed* set of policies.

Most ASes that prepend use multiple policies. Using the $\text{BGP}_{\text{Weekly}}$ dataset, we analyze the use of AS prepending policies between January 1st, 2018, and May 4th, 2020. We observe that more than 30.8% (20.8k) of the ASes prepend at least one prefix they originate (consistently with § 5.1.3), and most ASes use *mixed* prepending policies on May 4th, 2020. Among those using a single policy, the most common case is the *binary*policy, followed by *uniform* and *diverse*, respectively. Over time the fractions of different policies are substantially stable, with only a slight increase in all but *binary* policies. (In May 4th, 2020 we observe the following percentages: *uniform* 2.5%, *diverse* 1.4%, *mixed* 16.4%, and *binary* 10.4%). Once again, we note an increase in fine-grained prepending policies, which may be associated with a general increase in AS connectivity. We consider conceptually more straightforward for an AS to employ a single policy. Nevertheless, ITE may require the AS to use *mixed* policies, such as *binary* for some prefixes and *no-prepend* for others.

ASes with *mixed* policies mainly use *binary* policies. Next, we focus on ASes using a *mixed* set of policies and analyze the fraction of prefixes using each of the prepending policies. We group these few ASes according to the number of originated prefixes, in four "bins": $1-10, 11-100, 101-1000, 1000+$ prefixes. For each AS in a bin, we calculate the fraction of prefixes for each policy and present it as a boxplot in Figure 5.6 for May 4th, 2020 (we observe similar behavior for other snapshots). The plot shows only ASes that employ a *mixed* set of policies[73], and we observe that for these ASes, the most common is the *binary* policy (in all bins). We also find, confirming our intuition, that

ing findings in § 5.1.3.

[73]The same plot for different dates, namely all snapshots of the $\text{BGP}_{\text{Weekly}}$ dataset showed similar results.

the fraction of the *diverse* policy increases with the number of prefixes an AS originates (more pronounced for the two larger bins). Conversely, we see the fraction of uniformly prepended prefixes decreasing (with AS size).

5.1.4.3 Uniform Prepending

Uniform prepending is widespread. There is no apparent reason for an AS to use the same prepending size for all its neighbors when originating a prefix, as it implies no differentiation among them. Nevertheless, on May 4th, 2020, we observe more than 29k (3.6%) uniformly prepended prefix-origin pairs originated by 5.8k (8%) ASes, out of which 1.7k (2.5%) ASes prepend all their prefixes uniformly.

Some prefixes use the uniform policy consistently. The use of *uniform* policy might be the result of temporary events. To determine whether this is a common case, we pick all the (25.8k) uniformly-prepended prefix-origin pairs on a specific date (December 31st, 2018), and use the $\text{BGP}_{\text{Weekly}}$ dataset to show the fractions of policy type change for these prefixes in the preceding/following months. Figure 5.7 shows that the total number of prefixes decreases both sides, as up to 20% prefixes were not visible earlier or stop being visible afterward. We observe that both before and after December 31st, 2018 the fractions of *no-prepend*, *binary*, and *diverse* (for this fixed set of prefix-origin pairs) increase while *uniform* decreases. In other words, for some of these prefixes, *uniform* prepending was temporary. On the other hand, for the entire period, we see at least 50% of prefix-origin pairs using the *uniform* policy. Since there is no guarantee that these are the same prefixes, we look into it further.

Between January 1st, 2018, and May 4th, 2020, we observe 1.16M prefix-origin pairs in our $\text{BGP}_{\text{Weekly}}$ dataset. Out of these, 108k prefixes are uniformly prepended in at least one snapshot, and 3.4k (originated by 1.1k ASes) use this policy the entire time—henceforth referred to as consistently uniform. We also note that another 13.1k (originated by 4.3k ASes) are uniformly prepended for at least one year, *continuously*. Thus, counter-intuitively, we find that a substantial number of ASes, roughly 6% on the Internet, are making consistent use of *uniform* prepending.

Uniform prefix prepending is dominated by small ASes. How large are those interesting cases of ASes uniformly prepending all their prefixes? To answer this question, we determine the total number of prefixes each of these ASes originate. Taking May 4th, 2020, as an example, there were 848 (out of 1717) ASes with only a single prefix. Another 767 ASes originated between 2 and 10 prefixes, and 89 ASes, between 11 and 50. The remaining 13 ASes originated more than 50 prefixes, *all* of them uniformly prepended, with the largest one originating 379 prefixes. Among the larger ASes (with 50+ prefixes), we identified a large online social network, two universities, and several ISPs. These ASes are from North America, South America, and Asia.

We then check for how long these ASes used the *uniform* policy. We find that 6k ASes (out of 74k that we observe when combining all snapshots) uniformly prepend all prefixes in at least one snapshot, and 263 ASes used this policy between January 1st, 2018, and May 4th, 2020. We also see that other 716 ASes uniformly prepended all their prefixes for at least one year. From the group of 13 ASes that on May 4th, 2020, were uniformly prepending all of their 50+ prefixes, we find the following: one AS used it at least since January 1st, 2018, five for at least the past two years, one for the past 22 months, three for at least one year. The others consecutively prepended between 2 and 5 months.

We account for potential artifacts when measuring *uniform* prepending. Even though our sanitation ensures global visibility of all prefixes, missing interconnections may cause

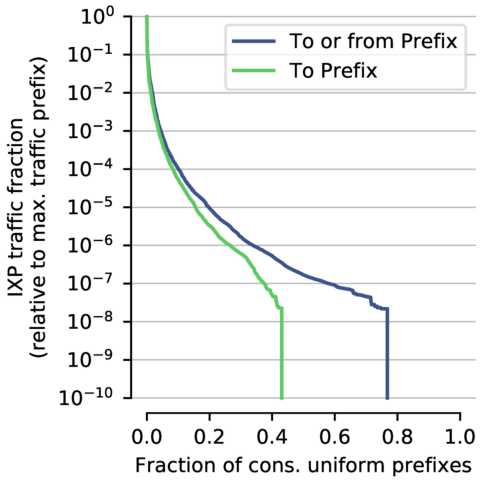


Figure 5.8: Uniform prefix-origin IXP traffic on April 28, 2020.

prefixes to incorrectly appear as *uniformly* prepended. There might be additional private network interconnects and peering links that are not visible to the BGP monitoring infrastructure [103, 365, 519]. We use two different approaches for cross-checking the results. First, we use bdrmapIT [313], a state-of-the-art tool, to infer interconnections based on public traceroutes from CAIDA's Archipelago (Ark) [84] between March 25th, 2020 and April 4th, 2020. We picked Ark traceroutes as it contains measurements to each /24 sub prefix from multiple vantage points. We then compare the list of interconnections from bdrmapIT with the ones we observe in our snapshot from March 30th, 2020 (the mid-point of our traceroutes). On our reference date (March 30th, 2020), 5.8k ASes were originating at least one prefix uniformly prepended. With bdrmapIT, we identify additional interconnection links for 1.7k (29%) of these ASes. Nevertheless, for the other 71% ASes originating uniformly prepended prefixes, bdrmapIT did not add any additional links. For the 263 ASes that uniformly prepended all their prefixes in all snapshots of the $\text{BGP}_{\text{Weekly}}$ dataset, we identify new links for only 18 of them. We note that even though we identify new links, we cannot draw any inference regarding the BGP announcements made through those links.

The second cross-check is to increase our visibility into the BGP routing system with data from two large global CDNs (each connected to more than 200 peering infrastructures) and one regional CDN present in more than 25 peering infrastructures. We choose CDNs since they have many private peering interconnections and need excellent visibility within the routing system for their operations. When checking their private data for all prefixes uniformly prepended in all snapshots of the $\text{BGP}_{\text{Weekly}}$ dataset, we observe more diverse policies for only 51 of those prefixes. Thus, we can conclude that our inferences are valid for the vast majority of the uniform cases.

Some of these prefixes carry large volumes of traffic. Some operators mentioned that consistently uniformly prepended prefixes might only carry little traffic, reducing the need to care about them. To check this hypothesis, we use a large European IXP as our vantage point on April 28th, 2020. We check the traffic volumes to and from each of the consistently uniformly prepended prefixes and observe that some of them carry as much traffic as prefixes of large social networks.[74]

[74]We are not allowed to disclose the actual byte counts of each prefix.

To provide a picture of the traffic associated with all consistently uniformly prepended prefixes in our vantage point, Figure 5.8 shows the fraction of bytes flowing towards each prefix (as well as in both directions) relative to the prefix with the most significant amount of traffic. For 57% of the prefixes, we do not observe any traffic towards them, and for 35%, we observe traffic from them, but not towards them. We note that only a few prefixes (<2%) carry representative volumes of traffic, either considering one or both directions. The vast majority of the prefixes we observe carry small volumes of traffic. While we cannot guarantee that other vantage points would observe similar numbers, we can conclude that, contrary to network operators' intuition, some of the consistently uniformly prepended prefixes carry substantial traffic volumes.

Many plausible causes for *uniform* prepending. Is there any *practical* explanation for the use of *uniform* prepending? We investigate this aspect by interviewing network operators, and report here a summary of potential causes: *Loss of a neighbor:* an AS may have used ASPP to differentiate between multiple upstreams but later terminated the relationship with some. Indeed, we observe that many (77% on May 4th, 2020) of the uniformly prepended prefixes are propagated via a single neighbor. *Lack of knowledge:* A reoccurring opinion is that many network operators, especially from small ASes, have limited understanding of BGP. Indeed, our analysis showed that many of the cases of *uniform* prepending were from small ASes. *Procrastination for stability:* Some network operators know about the presence of ASPP but are reluctant to remove it, out of fear of negatively affecting their reachability and/or routing stability in general. *Good news travels fast—bad news, slowly:* Some operators indicated that *uniform* prepending may help implement ITE policies when needed quickly. Instead of waiting to insert prepends when some change is needed, an AS can prepend in advance, and when the time comes, remove from one upstream to indicate a preferred route. Since "good news" travel fast, such an approach provides faster BGP convergence. *Sibling artifacts:* One operator pointed out that there might be cases in which two or more sibling ASes originate the same prefix, but with different prepending policies. We analyze this possibility using the CAIDA AS2Org dataset [219] and the data from May 4th, 2020. We find 17 cases in which two or more sibling ASes individually announced the same prefix, one uniformly prepended and the other with a different policy, resulting in a non-uniform policy. Strikingly, in 16 out of 17 cases, one of the ASes announces using *uniform* and the other one with a different policy. In one case, both ASes originate the prefix uniformly prepended, but with different prepending sizes. *Other ASes ignoring prepends:* One operator argued that *uniform* prepending might even lead to the desired traffic shift due to route-optimizers ignoring all prepends on one upstream and not on the others.

Looking at two relevant cases of *uniform* prepending. To validate our observations and to understand some of the actual reasons why ASes uniformly prepend their prefixes, we reached out to network operators from two ASes that have been originating uniformly prepended prefixes for more than one year.[75] One is a regional ISP that uniformly prepends 25 prefixes (out of 100+), while the other is a large online social network uniformly prepending all its 80+ prefixes. The operators from the regional ISP confirmed that the *uniform* prepending was unintentional and attributed it to legacy configurations and changes to their upstreams. The large online social network also confirmed that they were using *uniform* prepending unintentionally: the prepends are a result of how their internal routing platform operates. Since then, none of these ASes have removed the *uniform* prepends.

[75]We note that not all ASes are interested in discussing aspects of their operational practices. While discussing with network operators might not be enough for generalization, their comments allow us to provide insights regarding uniformly prepended prefixes.

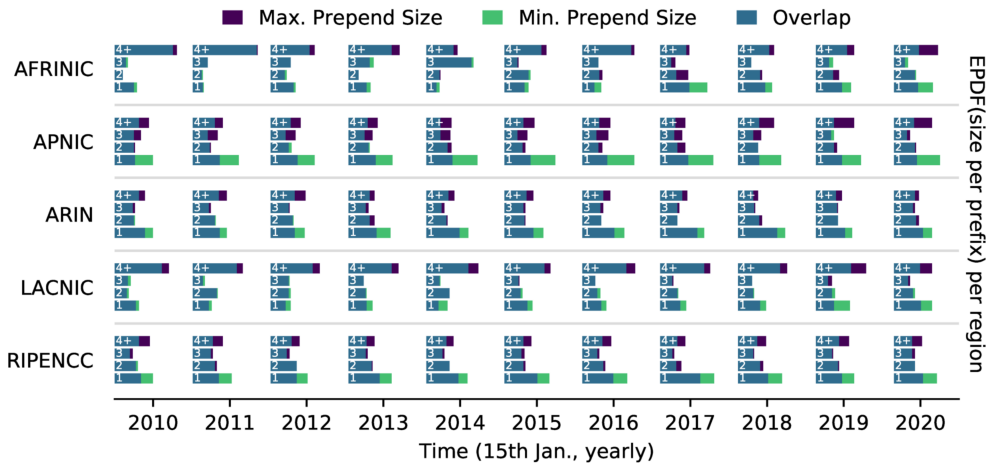


Figure 5.9: Prefix-origin: Prepend size by region across time.

5.1.4.4 Prepending Sizes

We use the $BGP_{Monthly}$ dataset to track if ASes changed the *number of prepends* they use over time. Since different service regions have distinct characteristics (e.g., availability of peering infrastructures [505]), we analyze them individually. We use the delegation files from the Routing Information Registries (RIRs) to identify the prefix region. While we acknowledge that there might be some misclassification, e.g., for global ASes, transferred prefixes, or due to IPv4 address delegations, we expect it to provide valid data for most prefixes. For each prefix, we analyze its minimum (non-zero) and maximum prepend size, i.e., if an ASes originates a prefix with 0, 2, and 3 prepends, its minimum prepended size is 2, and its maximum is 3. Figure 5.9 shows the results as a set of subplots, one for each service region and year. Each subplot shows a histogram for both the minimum (green) and maximum (purple) prepend sizes across all prepended prefixes. The blue bars represent the overlap between the green and the purple bars.

Prepending sizes are polarized and consistent among regions. We observe that the prepending size distributions for ARIN and RIPE, which hardly change during the decade, are *polarized*: most prefixes either have a prepending size of one or at least four.[76] LACNIC and AFRINIC are different: in 2010, there is no polarization, with a substantial number of prefixes with at least four prepends, while in 2020 polarization happens with a more significant incidence of prepending of size one. The change happens gradually over time, but in AFRINIC, the period 2014–2017 was an exception: prepending sizes varied "rapidly" and somewhat unpredictably. Towards 2020, the observable differences between the service regions become negligible—they are all polarized. In APNIC, the span between max and min prepend sizes increased, indicating more polarization, with an even more fine-grained set of prepending policies.

When we discussed these results with operators, they pointed out that the Internet infrastructure changed significantly throughout the decade, particularly for LACNIC and AFRINIC. Before 2015, many routes within Africa took long inter-continental de-

[76] In July 2019, we spot an AS originating four prefixes with 905 prepends, which is the maximum number of prepends we observe in our datasets.

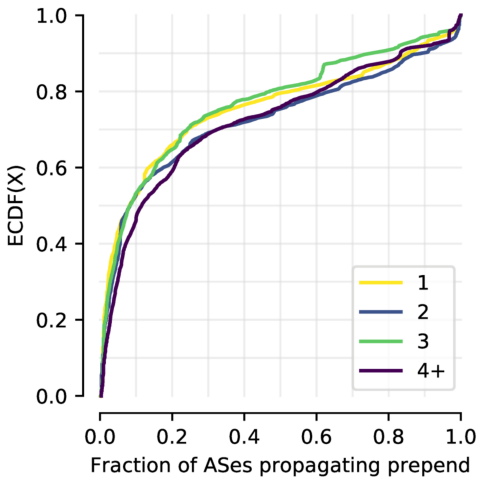


Figure 5.10: Fraction of ASes adopting longer alternative.

tours [198]. In order to use intra-continental paths whenever possible, ASes resorted to excessive prepending. With the increased availability of IXPs and peering within each region, intra-continental path diversity increased [163]. This may have reduced the need for excessive prepending, thus reducing prepend sizes.

5.1.5 Evaluating ASPP's Effectiveness

Given the widespread use of ASPP, in this section we explore the propagation of prepended routes and how effective ASPP is today.

Prepended paths propagate less than non-prepended ones. The common assumption is that the larger the prepend size of a route is, the less a network operator will expect it to propagate. Thus, prepending should mainly affect routing in the local neighborhood of an AS. To investigate how prepended prefixes propagate, we analyze all prefixes with a binary prepending policy where the prefix originator has not prepended one of the alternatives. For each prefix, we compute the fraction of ASes (out of those that we observe propagating the prefix) that propagate each alternative. Figure 5.10 shows the results for May 4th, 2020 (we observe a similar behavior for other snapshots). We observe that in 70% of the analyzed cases, independently from the prepend size, the prepended alternative traverses fewer ASes than the non-prepended one. While it may seem that the prepend size has no direct effect on route propagation, a more plausible explanation is that the ASes are tuning their prepend size to control how far the prefixes can propagate. Figure 5.10 shows that the distributions of the intended scopes of propagation are quite similar for different prepending sizes.

Nevertheless, it is unclear to which extent the adoption of a prepended path impacts the actual traffic flow, since (a) different routers in an AS may pick different preferred paths, (b) BGP monitors cover only a subset of ASes, and (c) some ASes might even remove ASPP (see § 5.1.6.1). Thus, we run active measurement experiments using the PEERING testbed as our vantage point. The PEERING testbed offers unique possibilities for our experiments. First, it operates on a geographically diverse set of locations—we refer to each location as Point of Presence (POP). Second, each PoP has a diverse set of

upstreams—the number of upstreams and the degree of connectivity of the individual upstreams differ among PoPs. Third, the PEERING testbed allows us to originate probing traffic towards a diverse set of targets using ICMP, TCP, and UDP.

On an abstract level, we create a scenario where we announce a route with prepends for some upstreams and no prepends (preferred) for others. Then, we use ICMP/TCP/UDP ping probes towards a diverse set of targets to generate response traffic towards the PEERING testbed AS. If the traffic enters via one of the preferred ASes, we refer to the result as a "hit", otherwise as a "miss". We note that the PEERING testbed allows us to correctly identify in which of the POPs the response has arrived.

Target selection. We base our target selection on Rapid 7's list of HTTP/1.1 GET responses [400]. We first select only IP addresses that responded with the HTTP status code "200 OK" when queried by an HTTP/1.1 GET request. To sample a diverse set of targets, we first map IPs to ASes by performing a longest prefix match on the closest snapshot of our $\text{BGP}_{\text{Weekly}}$ data set. Afterward, we classify ASes as follows: (1) we use a public list [507] to identify Tier-1 ASes; (2) we use CAIDA's AS type classification [85] to identify "Content" and "Enterprise" ASes; (3) we identify the remaining ASes as either "Access" or "Transit"—based on whether we observe them *only* as origin ASes in the $\text{BGP}_{\text{Weekly}}$ snapshot[77]. Since the Tier-1 class only contains 23 ASes, we use all of them as target ASes. For each of the remaining classes, we sampled 250 target ASes, resulting in 1023 targeted ASes. By running our own GET requests, we make sure to select only ASes for which 20 different IPs respond, resulting in a final target set of 20460 IP addresses.

Upstream selection. While the PEERING testbed has hundreds of upstreams, only roughly 20 provide transit. Since ASPP will have no effect if the prefix is subject to prefix aggregation [405] by a remote AS, we check how "well" our prefix propagates. We then announce it in one upstream per time and check how many monitors observe the prefix without aggregation. We filter out those upstreams that propagate our prefix to less than 200 monitors after 30 minutes of convergence. After this step, 11 transit providers—present at 10 different PoPs—remain. For the sake of simplicity, we focus on only one transit provider per PoP. We use the following PoPs: Amsterdam (A), Clemson University (C), Georgia Institute of Technology (GA), GRnet (GR), Northeastern University (N), Seattle (S), UFMG (UF), Utah (UT), University of Washington (UW), and University of Wisconsin (W).

Experiments. Each experiment employs a pair of PoPs, and we repeat it for all combinations and for different sizes of prepending (none, one, two, and three). We then run three sets of experiments. In the first set, we pick one upstream from each PoP and announce our test prefix on both—one with prepending and one without prepending. In the second set, we announce the prefix to all upstreams, prepending for all but one. In the last set of experiments, we announce the prefix to all upstreams but prepend to only one. We refer to specific choices of prepending size, upstreams, and experiment-class as an *iteration*. Our experiments took place between August 27th, 2020, and September 21st, 2020.

Iteration schedules. We deploy two similar iteration schedules that only differ in their first two rounds of announcements. For the "*Post*"-schedule, we start each iteration announcing our prefix P via all upstreams without any prepending. After waiting 15 minutes to allow BGP to converge, we announce P with X prepends via the chosen upstream (Appendix § C.1.1 shows a detailed graphical timeline). For the "*Pre*"-schedule, we do the opposite: we first announce P with X prepends via the chosen upstream; we wait for 15 minutes, and finally, announce P without prepending via all but the

[77] All those ASes are in the "Access/Transit" class in CAIDA's classification.

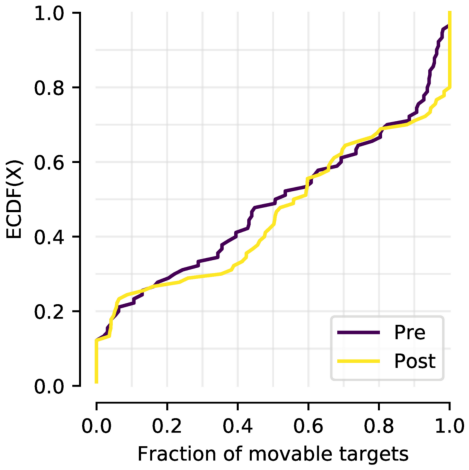


Figure 5.11: Fraction of potentially movable targets.

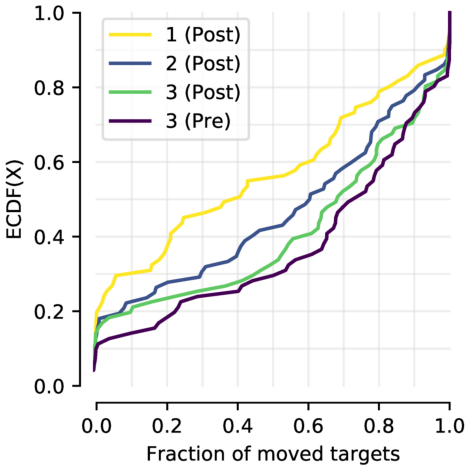


Figure 5.12: Fraction of actually moved targets.

chosen upstream. We employ both schedules to contain the impact of route age as a tie-breaking factor. We wait another 15 minutes for both schedules for BGP convergence before starting a 25 minutes long probing period. Each probe consists of ICMP, TCP, and UDP pings since the transport protocol can potentially bias the forwarding path [50, 382]. We probe once per minute all targets. To reduce probing bursts, we spread the packets evenly across the one minute time interval. To enable targets to opt-out, we embed our contact information in the payload of every probe. The cleanup phase starts 30 minutes after the start of the probing phase. Thus, we have a 5-minute break to ensure that the last responses can arrive before we withdraw the prefix. To allow for BGP to converge and minimize the risk of BGP Route Flap Damping, we wait for 30 minutes before starting a new iteration.

Data cleaning. In our results, we only consider those targets for which we see a significant number of responses: we require at least 10 of 25 probes for each protocol to be successful. However, we notice multiple probing artifacts, including many duplicates, additional ICMP packets, and RST packets. Thus, we first clean our data in the following manner: (1) we remove duplicate packets by relying on ICMP and TCP sequence numbers—since we sent SYN-packets, we receive duplicate TCP SYN-ACKs and RESET packets caused by receiver timeouts; (2) we only consider ECHO-REPLY ICMP packets—we remove, in particular, ICMP TYPE 3 (destination port unreachable) for UDP and TCP probes; (3) we hardly get any responses to the UDP probes, hence, we do not further consider them; (4) for a given iteration, we remove all targets for which we receive responses via multiple interfaces—this can, e.g., occur if an AS uses load balancing. Overall, these steps remove less than 3 % of the unique iteration-target combinations for ICMP and TCP.

Location matters when using only two upstreams. First, we look at how different prepending sizes influence routing behavior when using only two upstreams. Figure 5.11 shows the ECDF for the fraction of potentially movable targets (i.e., those targets initially routed via the later prepended upstream) per iteration and iteration type. We observe that our tested upstream-pairs cover the entire spectrum of scenarios, i.e., few,

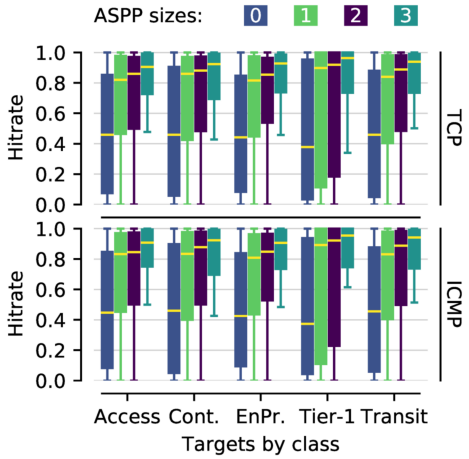


Figure 5.13: Hitrates by protocol and target class.

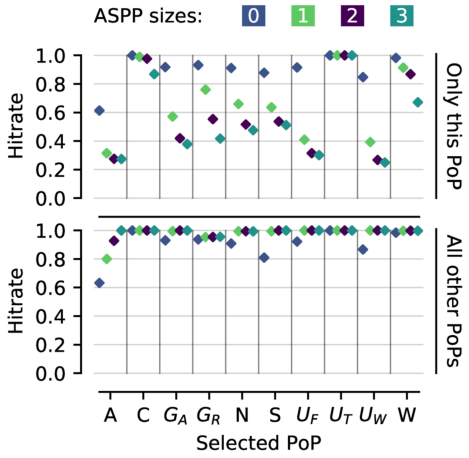


Figure 5.14: Hitrates when prepending 1 (top) | N-1 (bottom) PoPs.

medium, and many potentially movable targets. Given this insight, we investigate how many of the potentially movable targets have been moved by each prepend size. Figure 5.12 shows an ECDF for the fraction of actually moved targets (based on the number of potentially movable targets) per PoP combination. We observe that the effectiveness of prepending can strongly depend on the location (for around 20 % of cases, ASPP has moved no targets, while for another 20 %, it moved almost all targets). We further observe that the change from a prepend of size one to a prepend of size two has a much larger impact than the change from size two to three. While we observed that the *Pre*-schedule performs slightly better than the *Post*-schedule (see the effectiveness of the maximum prepend size for both schedule types in the figure), the route age did not significantly affect our results. When manually looking into our data, we observe that for some pairs, the traffic shifts can happen either way (e.g., GRnet and Northeastern University), whereas for others, prepending has little effect (e.g., for Georgia Institute of Technology and Clemson University). The lack of effectiveness of ASPP might be caused by the low connectivity degree of the ASes. However, we observe a different result for Northeastern University despite the same number of upstream providers of Clemson University and Georgia Institute of Technology. This highlights that location (not only connectivity) plays an essential role in the effectiveness of ASPP. In addition, we observe that traffic shifts, in most cases, are not gradual; instead, there is a minimum prepend size necessary to shift a majority of the targets.

Effectiveness differs based on the target class. Based on the above results, we study if the probing protocol and target class change the effectiveness of ASPP. Figure 5.13 shows a box plot of per-target hit rates (i.e., fraction of experiments where the target was a hit) per prepend size, network type, and transport protocol. Comparing the top plot with the one at the bottom, highlights that the overall hit rates are the same for both protocols. Comparing the different network classes, shows that Tier-1 targets were the hardest to influence using ASPP; however, the difference between target classes is not statistically significant.

With many upstreams, ASPP is able to shift almost all targets consistently. Finally, we

analyze prepending's effectiveness for more than two upstreams (second and third sets of experiments). Figure 5.14 shows the hit rate per PoP when only one PoP is prepended (*top*) or when all other PoPs are prepended (*bottom*). In the experiments in which all but one upstream use prepending (bottom plot), we observe that, except for few cases, even small prepending sizes steer all traffic to the non-prepended upstream. The same holds for the inverse (top plot). If only a single PoP prepends, its hit rate quickly drops with increasing prepend size; however, it never drops to zero.

Discussion. In conclusion, with only two upstreams, the effectiveness of ASPP is strongly dependent on the location within the routing ecosystem; whereas with many upstreams, ASPP is able to shift almost all targets consistently. This notion is consistent with our conversations with operators. On the one hand, a few operators told us that certain ASes (mostly CDNs) might ignore prepends during their best-route selection, leading to limited effectiveness. On the other hand, many operators claimed that prepending works well for their networks most of the time, highlighting that ASPP is indeed useful for certain ASes.

5.1.6 Security Implications

In this section, we shed light on some of the security concerns of ASPP that the community recently brought to network operators' attention [298, 299, 309, 479]. We first analyze if ASes manipulate prepended paths, i.e., remove prepends. Then, we experimentally verify and evaluate—on the Internet—the potential impact of hijacking of prepended prefixes as a basis for discussing the increased vulnerability of prepended prefixes. Finally, we estimate if ASes that prepend their prefixes also use RPKI-based Route Origin Validation to protect their prefixes against hijacks.

5.1.6.1 Is Removing Prepends a Common Case?

When propagating routes, ASes should prepend their ASN at least once and keep the remaining AS path unchanged [405]. Nevertheless, no mechanism prevents an AS from modifying the path. Indeed, there have been reports about ASes (possibly) removing prepends from paths [524]. An AS might remove (all) prepends from a path to create a shorter path and potentially attract more traffic. Besides malicious behavior (i.e., for traffic inspection), potential reasons include economics (e.g., to earn revenue by trying to increase the 95th-percentile of the exchanged traffic [361, 453]) and performance (e.g., to adapt traffic flow).

Consider the scenario of Figure 5.1(b), where AS A announces the prefix P to its two upstreams (*AS B* and *AS C*). *AS B* receives the non-prepended route, while *AS C* receives a route with three extra prepends. AS A would expect that most of the traffic towards prefix P would arrive on the link with *AS B*. Now suppose that *AS C* intends to increase its revenue. If *AS C* removes (all) prepends added by AS A, it makes its route shorter and more attractive to others.

Methodology. We check if we can observe such behavior happening systematically in the wild. We perform active measurements, since using passive BGP data to infer path manipulations is difficult (e.g., due to lack of visibility). Using the PEERING testbed, between May 3rd, 2020 and May 12th, 2020, we announce our prefix with three prepends via one of the PEERING's upstreams, and 30 minutes later, we withdraw it. After the withdrawal, we wait for another 30 minutes before starting a new iteration using a different upstream. After iterating through all available upstreams, we analyze all BGP

updates (visible at route collectors) for our prefix. If we identify an update where at least one prepend is missing, we mark the upstream for further analysis. In the end, we do an in-depth experiment for the marked upstreams (that removed at least one prepend). For each of these, we announce a prepended path and wait 15 minutes for BGP to converge. Then we manually inspect the chosen best routes via BGP looking glasses and route servers to identify which AS is likely the one that is removing prepends. Then, we withdraw the prefix. After 45 minutes, we check the next marked upstream. We announce our prefix using 231 different upstreams, resulting in more than 22k observed paths and 738 traversed ASes.

Prepending removal is rare. After manual investigation, we find that a single AS removed prepends, on a single path (in a previous run of this experiment, in September 2019, we found three ASes consistently removing prepends). We cannot attribute this to malicious behavior, as we learned from conversations with network operators that some route optimizers might remove prepends.

5.1.6.2 Can ASPP "Ease" Prefix Hijackings?

By artificially increasing the AS path length, an AS makes a route "less attractive" to other ASes. However, this behavior may create opportunities for other ASes to hijack this prefix for a larger part of the Internet ecosystem, since longer paths are more suitable for prefix hijacking [54]. Recall the scenario of Figure 5.1(b). Let us assume an AS X (un)intentionally originates a path for prefix P that contains AS A as the first hop. ASes that use a prepended path are more likely to adopt this new route (originated by AS X) since it is shorter than the one originally propagated by *AS C*. Possible variations of this scenario reflect different prefix hijacking types (e.g., using an illegitimate origin, or manipulating the path so that the malicious AS is next to the actual origin AS [107, 456]) and route leaks [211, 475]. In all these scenarios, a "bad" route may replace a legitimate prepended route.

Routes with at least three prepends are more vulnerable to prefix hijacking. Recall that there have been reports that ASPP may increase the risk of prefix hijacking [298, 299, 479]. To better understand to which extent different lengths of ASPP facilitate the adoption of hijacked routes, we performed an experiment using the PEERING testbed.

We ran our measurements between January 13th, 2020, and January 17th, 2020. In each round, we announce our prefix using two different ASNs as originators. We first announce it via one of the PEERING's PoPs to all attached upstreams using AS61574 as originator and 0, 1, 2, or 3 prepends. Then, 15 minutes later, we also announce the same prefix via a second PEERING PoP to all its upstreams without prepends using AS61575.[78] 30 minutes after the second announcement, we withdraw all routes for the prefix. 30 minutes later, we repeat the experiment using a different combination of PoPs and/or number of prepends. We select PoPs based on their location and number of upstreams: Amsterdam, 44; Seattle, 33; GaTech, 4; GRnet, 4; and Clemson, 1. To capture the prefix hijack's impact, we analyze the fraction of BGP monitors that adopted the "hijacked route" via AS61575.

Figure 5.15 shows the fraction of monitors that adopted the hijacked route per pair of PoPs and prepend size. The results confirm the intuition that the likelihood of prefix hijacking succeeding increases with the number of prepends. Overall, we find that if the initial announcement used three prepends, at least 94% of the monitors adopted the hijacked route, even when the hijacking location only has single upstream (e.g.,

[78] The PEERING testbed requires us to add AS47065 after the originating AS to the AS path.

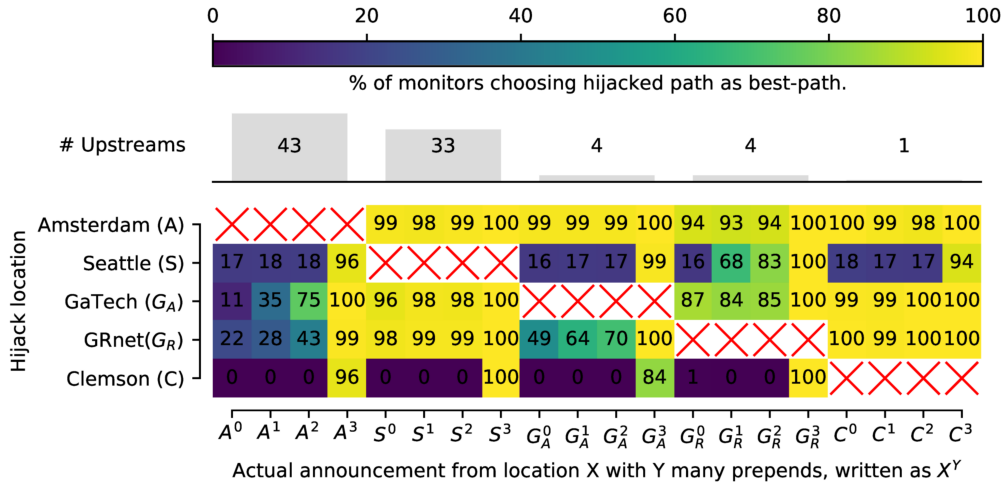


Figure 5.15: Hijacking: Fraction of BGP monitors adopting a hijacked route.

Clemson). Still, connectivity plays a vital role in the success of prefix hijacking. For all cases where we attempted to hijack the prefix from Amsterdam (a highly connected PoP), we succeeded for at least 93% of the monitors. In contrast, when we hijack via Clemson (a poorly connected PoP), we only succeed if the other PoP is prepending three times. Except for Clemson, all other PoPs were able to hijack Seattle mostly. Also, Seattle (with 33 upstreams) had less success in hijacking routes unless they had three prepends, which highlights the complexities of the Internet routing ecosystem.

Our results using a uniform prepending policy are an indication of how ASPP can increase the success of a hijacking attempt. While in § 5.1.4 we show that ASes uniformly prepend many prefixes, most ASes use a binary or diverse prepending policy, whereby one route is often not prepended. This means that the increased risk of hijacking only applies to the part of the Internet that chooses the prepended route.

More than 18% of prepended prefixes include apparently unnecessary prepending, which increases their exposure to hijacking and route leaks.

While for most prepended prefixes (169k) the minimum prepend size is 0, still many ASes originate prefixes with at least one prepend to all their neighbors, which can increase their exposure to hijacks and/or route leaks. For example, on May 4th, 2020, 6.9k ASes originated 38.5k prefixes with this characteristic (18.6% of all prepended prefixes). Among these, 29.4/7.4/2k used a uniform, binary, diverse policy. All these routes contain at least one unnecessary prepend—all their policies can be implemented with less prepending (at least as observable at the BGP monitors).[79] To further understand such potential risks, we use the $\text{BGP}_{\text{Weekly}}$ dataset to analyze the *minimum prepending size* for all prepended prefixes. Based on results in Figure 5.16, we see that the above finding holds across time, and also that the number of affected prefixes has grown.

5.1.6.3 RPKI-covered, Prepended Prefixes

One of the main techniques for enhancing routing security is ROV. RPKI allows ASes to create ROAs (Route Origination Authorizations) for each of their prefixes that other ASes

[79]We confirmed this conclusion in our conversations with network operators.

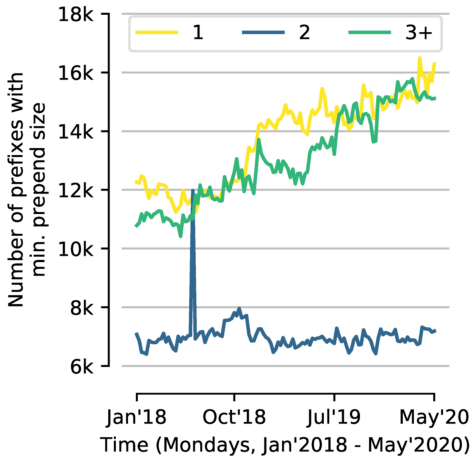


Figure 5.16: Prefix-origin: Pairs with at least X prepends.

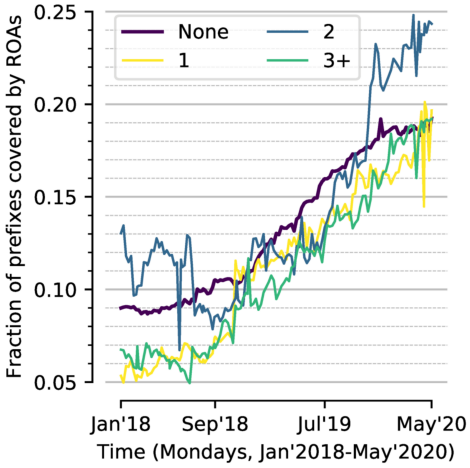


Figure 5.17: Fraction of prepended prefixes with ROAS.

can use to validate routes using ROV (Route Origin Validation) [108, 409, 488]. Although ROV cannot avoid the removal of prepends (see § 5.1.6.1), it can protect against prefix hijacking attacks in which the hijacker alters the origin AS [107, 456]. Given that ASPP potentially increases the exposure during hijacking attacks, we analyze to which degree prepended prefixes are protected by ROAs.

Most prepended prefixes are not covered by ROAs. We use the ROAS dataset to check which of the prepended prefixes in the $\text{BGP}_{\text{Weekly}}$ dataset has a ROA object. Figure 5.17 shows the coverage by ROAs of all prefixes in which all alternatives contain prepend and for those prefixes without prepend (none).

We observe first that the fraction of prefixes covered by ROAs has been increasing in the past years. On the other hand, we note that no more than 25% of the prefixes in each prepending class have ROAs[80]. This indicates that most prepended prefixes are not even partially protected against prefix hijacking attacks, regardless of the minimum number of prepends.

Discussion. Our security related results confirmed the assumptions that most network operators shared with us. Nevertheless, some of them argued that coming close to a specific traffic distribution may be more important to some ASes than reducing the potential impact of prefix hijacks—especially with the added security due to the increasing ROV deployment.

5.1.7 Related Work

Previous studies already focused on characterizing ASPP, understanding its effectiveness, and pointing out possible security aspects.

Characterization. To understand the characteristics of ASPP, previous work analyzed the

[80]We note that the fraction of prepended prefixes whose minimum number of prepends is zero that has ROAs is similar to the ones in the plot.

view of ISPs [164], IXPs [76], and route collectors [77, 174, 503, 525]. Since their numbers were inconsistent, we refreshed and extended their findings by performing a 10-year analysis of the main properties of ASPP. In addition, our work is the first that focuses on prepending policies rather than only utilization rates.

Effectiveness. Swinnen et al. found—in simulations based on a degree-based network model—that ASPP cannot always move all traffic [485]. This finding was later confirmed in 2004 by Quoitin et al. when running measurements from a single vantage point connected to two upstreams [395] (similar to our effectiveness measurements). In contrast to their methodologies, we emulated and tested more than 100 real-world location combinations and showed that the effectiveness of ASPP varies substantially by location and the number of upstreams through which an AS announces the prepended prefix.

Security. Zhang et al. analyzed the potential of interception-attacks exploiting ASPP based on simulations on an AS Graph extracted from the public BGP data of RouteViews and RIPE RIS [524]. They show that well-connected ASes (e.g., Tier-1 ASes) are less prone to this type of attack and that longer prepends amplify their risks. We actively measure the security impact that ASPP has based on hijack emulations from various locations and experiments to identify ASes that remove prepends; we also observed that 18.6 % of prepended prefixes have unnecessary prepend sizes that increase their exposure to attacks.

5.1.8 Final Remarks

Despite mixed opinions about ASPP in the networking community, we find that ASPP is still very present on the Internet, and its utilization is slightly increasing. Surprised by this, we checked with operators and found that the main reasons are the simplicity of ASPP and the fact that it does not have any prerequisites. Our analysis of ASPP reveals that prepending policies are mostly stable over time; that ASes are using a wide range of policies when announcing their prefixes; and that prepend sizes are becoming polarized—with either one or more than three prepends.

We unexpectedly spot many ASes uniformly prepending (all) their prefixes to all neighbors, hence not influencing any remote routing decision. Via our conversations with operators, we identified poor housekeeping of BGP configurations, limited knowledge about BGP, and desire for stability as the possible leading causes. Our complementary analyses with traceroutes and cross-checks with CDN data confirm that, the limited visibility of public route collector projects cannot be the explanation for most of our observations.

During our interviews, many operators pointed out that using ASPP suffices to accomplish their ITE goals. Our active measurements confirm that ASPP is *effective*—since even small prepend sizes can steer the traffic of multiple routes—if used with many upstreams. When using only two upstreams, ASPP's effectiveness is dependent on the AS location.

We also discuss the security implications of ASPP. First, we show through active measurements that some ASes remove prepends, but it appears to be rare at the moment. Second, we find that ASPP can increase the spread of prefix hijacks, since the hijacked route is more attractive (than the actual route) to a larger fraction of ASes. Third, we detect that ASes originate 18% of the prepended prefixes with unnecessary prepends.

ASPP has value, and ASes are using it extensively on the Internet. However, as Internet paths are getting shorter (as the core is getting denser), the need for large prepend sizes

is decreasing. Thus, given the security implications of large prepends and the fact that small prepends are often sufficient for moving traffic, we recommend network operators to review their prepending policies, removing unnecessary prepends and using small prepend sizes when performing ITE.

5.1.9 Ethical Considerations

To conduct our study, we relied on active as well as passive measurements. When we used the PEERING testbed to actively announce prefixes to the Internet, we ensured that we did not overwhelm any networks by waiting 15 minutes between consecutive announcements. When actively sending traffic from the PEERING testbed, we ensured not to cause any harm through the following mechanisms: (1) we sent probing packets at a low rate, i.e., each target IP was probed with one ICMP, one TCP, and one UDP probe once per minute. (2) we avoided traffic bursts by spreading the sending of probes equally throughout a one-minute interval. (3) we included our contact info in the payload of each probe providing details on how to opt-out of the probing process. We have not received complaints nor requests to opt-out of the experiments during the entire duration of our active experiments.

While most of our passive datasets are publicly available, we cannot share any of the data received from CDNs and the European IXP for validation purposes. As this limits the possibility for others to take action based on our results, we tried, whenever possible, to reach the network operators of ASes that consistently announce uniform prefixes.

5.2 Prefix De-Aggregation Attacks

The Internet is an indispensable resource for communication, trade, commerce, education, and entertainment in today's world. Over the past years, the Internet has become more and more important in people's everyday life. Moreover, the reliance of many societies on the Internet has only increased with the COVID-19 pandemic [74, 75, 79, 166, 297].

To counter IP address exhaustion among other things, the IPv6 protocol was designed more than 20 years ago [136]. Although IPv6 usage was low initially, more and more websites, services, and networks are now using IPv6. Around 20% of all websites are IPv6-ready [501], a third of all Autonomous Systems announce IPv6 routes [224], and around 40% of Google users globally access the website via IPv6, with some countries reaching a deployment of more than 60% [191].

However, the additional capabilities provided by IPv6 come with new threats: e.g., targeted probes can find home routers in the vast IPv6 address space [177, 439]; privacy mechanisms can be defeated and devices can be tracked over time [438]; even a single device using legacy IPv6 addressing can foil all privacy extension and prefix rotation efforts [441]. In addition to these attacks on the data plane, IPv6 also introduces new challenges for the control plane. Its vast address space raises questions about the scalability of the Internet's standard interdomain routing protocol: the Border Gateway Protocol (BGP). Some large networks own /19 IPv6 prefixes, each of which contain *half a billion* possible /48 subprefixes that reliably propagate over BGP. As routers have a limited amount of memory available, such a large number of IPv6 prefixes would exhaust the memory of many routers deployed today on the Internet.

In this section, we introduce and analyze a BGP flooding attack named Kirin—standing

for **K**illing **I**nternet **R**outers in **I**Pv6 **N**etworks—that overcomes traditional protection mechanisms (e.g., per-session prefix limits and route flap damping) by originating millions of unique IPv6 routes distributed across many BGP sessions. More specifically, we make the following key contributions:

- **Kirin:** We describe a BGP flooding attack, Kirin, its threat model and recent technology trends that enable it (cf. §5.2.2).
- **Theoretical Feasibility Analysis:** We combine real-world data with an Integer Linear Programming definition of our attack to theoretically analyze its feasibility wrt. (1) required IXP presence, (2) required sessions at each IXP, and (3) the resulting ASes that are affected. We show that Kirin is not only theoretically feasible, but can already affect ASes globally when connecting to just 20 transit providers at 25 IXP peering LANs (cf. §5.2.3).
- **Router Testbed Evaluation:** We test the effects of Kirin on one virtual and one physical router from different vendors. We find that we can exceed router memory already with 109k specially crafted IPv6 prefix announcements (cf. §5.2.4).
- **Real-world Experiment:** To demonstrate how Kirin's requirements can be met in the real world, we deploy the infrastructure needed to perform a small-scale Kirin attack. Our testbed was built from scratch and fully functional in a few weeks and cost only 500 EUR (cf. §5.2.5.1).
- **BGP Testbed Validation:** We validate our assumptions on how routes propagate using BGP data analysis and real-world experiments from our own and the PEERING testbed (cf. §5.2.5.2).
- **Defense Mechanisms & Notification:** We extensively discuss possible defense mechanisms (cf. §5.2.6) and lay-out our plan for a vulnerability notification campaign (cf. §5.2.7).

5.2.1 Background

BGP is the standard interdomain routing protocol, where Autonomous Systems (ASes; groupings of routers) announce and redistribute reachability information between each other according to certain routing policies [156]. When an AS receives an announcement, it usually consists of an IP prefix and a path of ASes to traverse; the term *route* thus is used to refer to a prefix-path pair.

Routers. Routers within ASes establish dedicated TCP sessions over which BGP is run between them. For the IPv4 and IPv6 protocols separately, each router holds a routing information base (RIB) that contains all currently active routes. For each prefix, a router determines its current best-path from all alternatives, and then installs the best-path's next-hop in its Forward Information Base (FIB). The FIB is then used to quickly retrieve the next-hop to which the router forwards a packet. To allow a router to achieve high throughput, the FIB is often stored in expensive, specialized memory formats such as TCAM or DRAM, which are optimized to quickly perform longest-prefix-match operations. This specialized memory is often a scarce resource due to its high cost, a fact that has previously been exploited for theoretical stress attacks [139].

Route Propagation in Theory. Once a router determines the best-path for a given prefix, it may redistribute the new route to its BGP neighbors. Whether a route is redistributed

to a certain neighbor is determined by applying a chain of egress filter rules; the sum of these rules commonly expresses more abstract policies that represent a network's business incentives. In 2001, Gao and Rexford first categorized the business relationships between ASes and identified the consequential redistribution patterns [176]. The Gao-Rexford model describes three types of relationships: (1) transit relationships in which a customer pays a transit provider to forward traffic, (2) peering relationships in which two ASes achieve mutual benefits by forwarding traffic for one another at no cost, and (3) sibling relationships in which two ASes appear as two logically separate AS numbers but are operated by the same organization and hence produce "arbitrary" redistribution patterns. Based on these relationship categories, ASes only redistribute routes that provide monetary benefit. While ASes would redistribute routes they received from their customers to all other neighbors (as the customer ultimately pays for the delivered traffic), they would not forward routes that they received from peers to other peers or transit providers (as the peer would not pay for resulting maintenance or transit costs).

Route Propagation in Practice. While these abstract relationship models still hold today [180, 251, 292], they are partially superseded by more nuanced relationships [154, 185], e.g., partial transit (i.e., for a limited set of routes), paid peering (i.e., one AS pays the other a small fee to access routes towards its customer cone—the set of all direct and indirect customers), or hybrid relationships, where the actual relationship between two ASes depends on the physical location. Besides business relationships, the propagation behavior of an AS can be influenced by various factors including (1) route reputation—some ASes may filter and ignore routes if they or their originating AS appear in block lists [6, 128, 474]—(2) aggregation strategy—ASes may aggregate routes before redistribution to limit routing table growth [255, 280] or provide customers with (partial) default routes [434]—or (3) remote signaling where, e.g., customers instruct their providers to redistribute a route in a certain way using BGP Communities [69, 478].

Propagation Timing. There are also factors that determine *when* a router propagates a route. Many ASes configure a Minimal Route Advertisement Interval (MRAI) during which announcements are aggregated; after this timer expires, only the active best-path is propagated, which reduces the number of propagated updates due to route flapping[81] [161, 181]. Another widely deployed approach which influences the propagation time is Route Flap Damping (RFD) [192, 383]. An RFD-enabled BGP session keeps a penalty counter for each prefix. The counter is incremented for each received update and decremented at fixed time intervals. If the counter exceeds a "suppress" threshold, the router starts to dampen the prefix, i.e., it withdraws it from all its peers and no longer redistributes updates for it. It remains in this state until the counter has decreased to some "reuse" threshold, after which it starts to redistribute the prefix again.

Path Exploration. A router may enter a "path exploration" period once it receives a withdraw. When the origin AS entirely withdraws a prefix, a remote AS receives the withdraw messages from different paths spread across a certain time window—a result of the propagation timings of the routers along a path. If a router knows multiple paths for a prefix and it receives a withdraw for its current best-path first, then it chooses some other path as its new best-path and generates an update that reflects the change. If a router knows N paths for a prefix, it may repeat this cycle up to $N-1$ times (in the worst case) before it finally redistributes the withdraw message: i.e., it "explores" potentially all of the other available paths before it fully withdraws the prefix. Path exploration is present in most (if not all) active route propagation experiments and has been studied extensively [17, 307].

[81] i.e., routes that generate many update messages as they rapidly shift between two or more configurations.

Internet eXchange Points (IXPs). Over the last decade, peering has increasingly gained importance [71]. IXPs allow their members to cost-efficiently establish peering sessions with other members on top of their existing peering LANs, i.e., layer 2 switching infrastructures that are bound to specific geographic locations [15]. Many IXPs also provide route servers to further facilitate peering: using a single BGP session, an IXP member can exchange routes with all other (often 500 or more) ASes that are connected to the route server [412]. As of 2022, there are more than 650 active[82] IXPs worldwide [378]. Some of these IXPs provide access to more than 1000 potential peering partners and routes for more than half of the Internet [71, 390]. These reachability benefits provided by IXPs also make remote participation attractive. Nowadays, "remote peering," i.e., connecting to a peering LAN via some layer 2 connectivity provider, has become the norm rather than an exception [94, 318, 360].

Topology Blindness. While IXPs are highly popular and have been shown to enable hundreds of thousand of interconnections, most of these interconnections are invisible to the existing BGP monitoring platforms [15, 390]. While these platforms, in total, operate 50+ route collectors that receive and dump routing updates from 600+ feeding ASes, they, in general, miss many peering links as those often do not propagate to any feeding AS [15, 49, 188, 365].

Route Aggregation & Filtering. To reduce their routing table size, some ASes perform route aggregation, i.e., they summarize multiple more-specific routes into a single less-specific route and only propagate this summary route [174, 255, 280]. Besides aggregating routes, operators often configure their routers to ignore specific types of routes. Especially routes towards small amounts of address space (i.e., those with CIDR sizes more specific than /24 and /48 for IPv4 and IPv6, respectively) are very commonly filtered [452, 480].

5.2.1.1 Related work

While the option to de-aggregate a prefix has been well-known in the operator community for multiple decades, academic focus on the issue is limited.

Chang et al. experimentally investigated the response of 3 commercial grade routers to large BGP routing tables in 2002 [98]. The authors found significant differences in how routers respond and highlighted that the BGP graceful restart capability could alleviate the effects of BGP malfunctions on IP routing. A deliberate attack and its impact on the Internet are outside the scope of that paper. Yet, similar to Ceasar et al. [83], the authors advocate for the use of prefix limits on BGP sessions. The operator community largely shares this sentiment as prefix de-aggregation often exacerbates the impact of route leaks [170, 386].

In 2013, Schuchard et al. first described the concept of prefix de-aggregation attack for IPv4 [450]. While they describe the same underlying idea, in comparison with Kirin, the paper does not consider various practical details: (1) they assume that the attack is executed by major transit networks with rich peering fabrics; (2) they assume that an AS can obtain enough address space via squatting (illegitimately announcing unused address space) and that filters against squatting are negligibly deployed; and (3) they assume that typical max prefix limits range between tens of thousands of prefixes and the full routing table size. While our work builds upon the same simple idea, it actively addresses these real-world issues, ultimately rendering the attack practically feasible: (1) based on discussions with network operators, we assume that prefix limits are widely deployed and usually range between hundreds to a few thousand prefixes on peering

[82] We consider only IXPs with operational status "active" and at least two participants.

sessions; (2) we leverage IPv6 as an enabler to source millions of legitimately allocated—and hence unfiltered and even RPKI-valid—prefixes; and (3) we leverage remote peering providers, VPS providers, and IXPs to assemble thousands of sessions allowing arbitrary actors to execute Kirin at a minimal cost. Thus, besides a theoretical feasibility analysis, we evaluate the interlinking parts of our improved attack model in practice and on real-world Internet data.

5.2.2 Kirin: Overview

In essence, the Kirin attack is simple and ostensibly obvious: the attacker introduces enough new IP routes to overflow the FIB and/or RIB of the BGP routers within victim ASes. After that, the attacker simultaneously withdraws all previously established routes, which triggers the path-hunting phenomenon that leads to a flood of update messages that impact the performance of routers.

The idea that routers may crash due to memory constraints is not new: many operators already reported crashed routers when the IPv4 routing table reached 512K and 768K routes [2, 158]. Nowadays, high-end devices from major router vendors support ~2–4M routes in total in their FIB: Cisco's Catalyst 8200 and 8500 platforms can store between 800k and 4M routes (depending on the exact model and its respective DRAM storage [117, 118]), Arista's FlexRoute Engine can store up to 2.5M total routes [48], and Juniper's PTX10001 platform can handle 2M total routes [515].

However, it is the new context and availability of new methods that we believe re-enable a well-known attack to be successfully executed on the Internet today, by anyone, and with a limited budget. Although there are various roadblocks built into the routing ecosystem to prevent the exploitation of the FIB/RIB overflow issue, Kirin uses a set of observations and tricks to maneuver the existing roadblocks.

5.2.2.1 Threat Model

Our threat model, which was already introduced in a similar form by Schuchard et al. [450], focuses on highly connected ASes with legitimate BGP speakers that act maliciously. The goal of our adversary is to fill the FIB or RIB within a remote router to the point where it fully exhausts the available memory using millions of prefix announcements. Hereby, the adversarial AS is not limited to transit ASes; as we demonstrate in §5.2.3 that even stub ASes are capable of reaching this goal. In fact, we show in §5.2.5.1 that an adversary can start without any resources or infrastructure and yet is able to perform prefix de-aggregation attacks within less than a month and at a cost bearable for individuals. Notably, an AS may either intentionally decide to become an adversary (and explicitly assemble the required infrastructure) or may be forced in this role by an outside entity that compromised various BGP routers or a global route controller.

While an adversary's router can only send BGP messages to the direct neighbors it established sessions with, it relies on those genuine peers to redistribute these messages according to common BGP policies. Further, our adversary may potentially ignore best common routing practices, yet must assume that all other ASes may implement them.

5.2.2.2 Enablers

IPv6. IPv6 addressing space is so much bigger than IPv4, that instead of assigning 1 IP address to 1 end-user—or even many more end-users through Network Address Translation (NAT)—in IPv6 end-users are typically assigned /64 prefixes each. As a consequence, Internet operators also handle much bigger IP prefixes, e.g., ARIN's allocation policy states that an ISP should never receive less than a /32 prefix allocation [47]. Given that the smallest IPv6 prefix that reliably propagates over BGP is /48 [391, 452, 480], potential bad actors could split their typical IPv6 prefix into *much* more subnets compared with their typical IPv4 prefix. Splitting a /29 IPv6 prefix is enough to inject 1M unique and valid routes into the global routing table. Note that these sub-prefixes can overlap: e.g., a /46 prefix can source 7 routes in total (1x /46, 2x /47, and 4x /48). In general, if C is the difference between the smallest propagating CIDR size (typically 48) and the parent prefix length, an attacker can source up to $2^{C+1} - 1$ unique routes.

Ineffective Route Aggregation. Given that we source all prefixes from the same continuous address space, a wide deployment of aggregation would nullify Kirin's attack potential. To overcome this challenge, Kirin only announces non-aggregatable prefix combinations to each neighbor and may also alternate its origin AS. Please note that the use of small, non-aggregated IPv6 prefixes is already common, and that the average prefix length is increasing over time [225, 226].

Per-Session Max-Prefix Limits. The most commonly recommended approach to prevent the announcements of many routes is to set a maximum number of accepted prefixes for each BGP session. Upon hitting this limit, the session may produce a warning, might be capped—i.e., stop accepting updates for new prefixes yet keep updating existing ones—or can be dropped entirely [111]. Because this approach requires only per-session state, it is simple to implement and requires no cooperation—two key factors that pushed today's wide deployment. Kirin attempts to respect per-session limits by distributing a dedicated set of prefixes to each of *many* BGP sessions: no single prefix is shared between any two sessions. Using this strategy transforms the goal of announcing millions of routes into a session-hunting challenge. We further explore this relation theoretically and experimentally in Sections 5.2.3 and 5.2.5, respectively. Moreover, during our experiments we find IP transit and IXP operators to be permissive about increasing the prefix limits when inquired. One major transit provider stated they do not impose prefix limits on IP transit links; another stated they allow the limit that we set ourselves in the Internet Routing Registry (IRR).

Accessible Internet resources. It is relatively easy to obtain an AS number and a large IPv6 prefix valid for use in the global routing system. A quick and relatively cheap way is to use services of a *sponsoring* LIR, who proxies a request for resources to one of the 5 RIRs (e.g., [14]). LIR operators can *lease* their allocated IP space, e.g., some offer /29 prefixes with a 48h free trial [403], which is enough to launch Kirin. Another essentially free (yet illegal) method for malicious attackers could be *squatting*, a method in which non-announced Internet resources allocated to an unrelated organization are used [351]. Finally, it is also possible to become a regular LIR and gain direct access to legit and large IPv6 allocations. For example, as of 2022, becoming a RIPE member costs under 2500 EUR and allows for /29 IPv6 allocations without providing any justification [344, 345].

Instant and cheap BGP peering. It is no longer true that in order to establish a BGP session neighboring networks must be physically connected [360]. Remote peering at IXPs is an established reality, and a recent study found that already over 10% of members of major IXPs are remote [318]. Commercial services allow for instantly establishing

peering links with dozens of significant IXPs, cloud operators, and data centers [323, 404, 408]. Furthermore, prompt provision of VMs with IXP peering sessions has never been easier: e.g., a VM with NL-IX peering could cost under 30 EUR per month [230], and a VM with BGP IP transit could cost just *a few* USD per month [500]. Moreover, while carrying our experiments for this section, we found it is easy to obtain *free* IPv6 transit—foremost from Hurricane Electric (HE), a major Internet operator, who actively seeks to establish bi-lateral peering sessions with new IXP members. We also inquired a few other major operators and found the cost of a BGP peering port with IP transit would cost around 100–300 USD per month, depending on location and bandwidth.

Circumventable Filtering. While it is hard to enter millions of route-objects into IRR databases, many providers nowadays also accept routes with valid ROAs. As ROA entries allow for CIDR ranges, an adversary may enter a single ROA with CIDR sizes /29–/48, wait for it to propagate, and then would pass, e.g., the route filtering checks of HE. [221].

5.2.2.3 Collateral Damage via Path Hunting

While Kirin itself mainly fills the FIB/RIB of victim ASes, it does so by announcing millions of routes globally that, at some point, need to be withdrawn from the Internet again. If a global route gets fully withdrawn, the path-hunting phenomenon may produces a burst of updates (see §5.2.1 for details).

Given that Kirin triggers this phenomenon simultaneously for millions of prefixes, it "accidentally" generates a distributed update flooding attack. Given that some ASes use route flap damping to ignore these announcements and stop the redistribution, it is hard to provide realistic estimates on the number of produced updates at each AS. Hence, we leave the analysis of collateral damage as future work and focus on Kirin's main idea: propagation of millions of prefixes via thousands of distributed sessions.

5.2.3 Theoretical Feasibility Analysis

In this section, we theoretically analyze Kirin's feasibility. We consider two different scenarios: (1) the adversary obtains (potentially costly) transit from a few providers and (2) the adversary obtains as many (virtually cost-free) bi-lateral and multi-lateral peers as possible. While, in reality, an adversary may use both of these scenarios simultaneously, examining them independently allows us to keep our analysis reasonably simple while still obtaining deep insights into Kirin's cost-benefit trade-off. Further, we assume that an adversary only establishes a single (virtual) port via a single method and service provider at each peering LAN.

We start this section by clearly stating the assumptions we make about route redistribution (§ 5.2.3.1) and the data sources that we build our analysis upon (§ 5.2.3.2). We then define the cost-benefit trade-offs for the first and second scenario as ILP problems (§ 5.2.3.3 and § 5.2.3.4) and finally discuss our analysis results (§ 5.2.3.5).

5.2.3.1 Assumptions & Definitions

Routing Policies and Assumptions. The policies that underpin today's inter-domain routing mostly follow economical incentives [23]. In particular, we assume that:

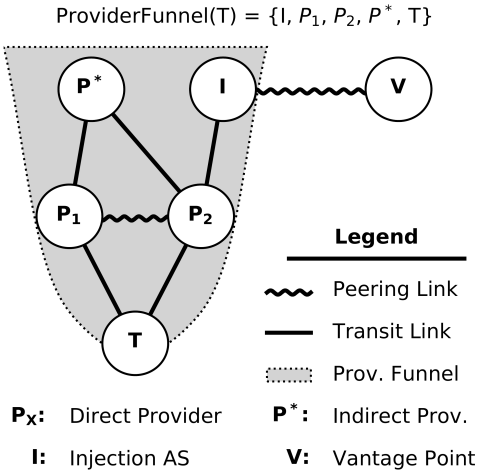


Figure 5.18: Provider funnel example.

1) If an AS receives a route from a customer, it forwards the route to all neighbors.

2) If an AS receives a route from a settlement-free peer or a provider, it forwards the route to customers only.

3) An AS will always forward a route by the above rules to maximize its economical gain.

The first and second assumptions are known as the Gao-Rexford redistribution model [176] and have been standard assumptions for more than two decades in the field of AS relationship inference [167, 185, 248, 251, 292]; the third assumption has frequently (yet usually implicitly) been used for simulating route propagation, e.g., [257, 331, 526]. Notably, these assumptions do not always capture the real-world behavior of all ASes perfectly (see, e.g., complex relationships [185] or non-economic incentives [180]), yet their frequent appearance in the related literature renders them as reasonable abstractions. Based on these assumptions, Luckie et al. introduced the notion of the customer cone, i.e., the set of all direct and indirect customers of an AS [292]. While they introduced multiple methods to calculate this set, we choose the one that only uses routes the AS forwarded to its peers and providers, as it yields more stable and realistic results. By recursively applying our three assumptions, one arrives at the high-level statements: (1) routes sent to a peer will eventually reach all ASes in the peer's customer cone and (2) routes sent to a transit provider will eventually reach all[83] ASes globally.

Provider Funnel & Funneling Degree. In this work, we introduce the concept of *provider funnel* PF_T as the set of all recursively added providers for a given target AS T. We use the example in Figure 5.18 to further illustrate this concept. In our example, T is multi-homed to two direct providers—P_1 and P_2. Neither P_1 nor P_2 are Tier1 ASes, so they also rely on different transit providers P^* and I to reach certain parts of the Internet. When P^* announces a route to P_1, P_1 likely forwards this route to T. Even though P^* and T share no direct connection, P^* is an indirect provider of T.

When executing Kirin, our vantage point V has connections to ASes within T's provider funnel. As these ASes redistribute our routes so they ultimately reach T, we call them *injection ASes*. Moreover, as V might maintain multiple BGP sessions to I (e.g., at different

[83]Notably, there are certain situations in which a route does not propagate, e.g., because it is filtered or because certain ASes only want to have a default route from their providers.

IXPs), we further define an *injection session* as a unique BGP session to an injection AS.

Finally, we call the number of ASes in PF_T as the *funneling degree* of T and denote it as FD_T. Note that we include T in its provider funnel, i.e., $PF_T = \{P_1, P_2, P^*, I, T\}$. We use the term *restricted funneling degree* FD_T^S to refer to the size of the provider funnel when only considering ASes in S, i.e., $FD_T^S = |PF_T \cap S|$.

5.2.3.2 Data Sources & Processing

We estimate funneling degrees using two inputs: (1) the number of sessions that each AS has with each peering LAN and (2) the provider funnel for each AS.

Estimating Peering LAN Sessions. On 2022-09-09, we generated a snapshot of EURO-IX's IXP database [159]. We further obtained a PeeringDB snapshot for that day from CAIDA's daily archive [90]. While the EURO-IX data set does not contain a direct reference to the IXP, it contains the PeeringDB identifier for each co-location facility, which allowed us to merge the `(peering LAN, ASN, IPv6 address)` triplets we extracted from both data sources. The obtained data describes 24k sessions via 725 peering LANs.

Estimating IPv6 Provider Funnels. While CAIDA publishes provider-peer-determined customer cone files on a monthly basis (available at [87]), this data set comes with two problems: (1) it it not available for the IPv6 routing ecosystem and (2) it only uses data from public route collectors which miss significant portions of the AS topology. Hence, we generate this data set (and most of the required tooling) from scratch.

We first extract all IPv6 routes from public route collector data via BGPStream on 2022-09-09 (including routes from all RIB snapshots and update messages). Next, we add routes from 130 IPv6 route servers of 11 IXPs—e.g., DE-CIX, LINX, and IX.br—including both primary and (potentially multiple) secondary servers. All of these route servers provide a public Alice-lg looking glass utility [131] that has a back-end API allowing for obtaining all IPv6 routes received from their peers. We automated the querying process and obtained the IPv6 routes of all route servers throughout 2022-09-09.

To estimate AS relationships, we utilize the publicly available ASRank script [87]. We modify the script to tailor it towards the IPv6 ecosystem [186]. We use the previously collected IPv6 routes and a list of route server ASNs—that we obtained by selecting ASNs with the "Route Server" network type within our PeeringDB snapshot—as input to the modified ASRank script, which leads to the inference of 247K peering links and 32K transit links. Finally, we convert the IPv6 paths and business relationships into peer-provider-determined customer cones [334]. To calculate provider funnels, we inverted these customer cones, i.e., we checked for each AS in which other AS' customer cone it appears.

5.2.3.3 ILP Formulation: Transit Scenario

Now that we obtained the required data sets, we can formalize Kirin's resource needs and attack potential. In our first scenario, we assume that the adversary chooses multiple transit providers and then joins peering LANs to establish additional sessions with the chosen providers. As discussed in § 5.2.3.1, we assume that routes announced to a transit provider propagate globally. As every prefix reaches each AS globally, we can focus on the number of sessions that can be obtained by using P_{max} providers and connecting to L_{max} peering LANs.

Sets. Let A be the set of all IPv6-enabled ASes and L be the set of all peering LANs.

Parameters. Let $\omega_{a,l}$ denote the number of unique sessions that can be established with AS $a \in A$ at peering LAN $l \in L$. We can then build the following session matrix:

$$S = \begin{pmatrix} \omega_{a_1,l_1} & \omega_{a_2,l_1} & \cdots & \omega_{a_{|A|},l_1} \\ \omega_{a_1,l_2} & \omega_{a_2,l_2} & \cdots & \omega_{a_{|A|},l_2} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{a_1,l_{|L|}} & \omega_{a_2,l_{|L|}} & \cdots & \omega_{a_{|A|},l_{|L|}} \end{pmatrix}$$

We further provide the parameters $L_{max} \in \mathbb{N}$ and $P_{max} \in \mathbb{N}$ that reflect the maximum number of peering LANs and providers that can be chosen.

Variables. We first introduce a binary decision matrix D that contains a binary decision variable $d_{a,l}$ for each $\omega_{a,l}$ that denotes whether provider $a \in A$ at peering LAN $l \in L$ is part of the solution. Further we introduce two sets of binary decision variables that help us to realize our constraints: CL contains a variables cl_l for each $l \in L$ that determines whether the adversary has to connect to peering LAN l while CP contains a variable cp_a for each $a \in A$ that determines whether a is chosen as a transit provider

ILP Problem Formulation. Given S, L_{max}, and P_{max}, our goal is to chose a set of providers and a set of LANs such that we can obtain the maximum number of sessions, i.e.,

$$\text{maximize} \quad \sum_{l \in L} \sum_{a \in A} \omega_{a,l} * d_{a,l}$$

To ensure that only L_{max} LANs and P_{max} ASes are chosen, we add the following two constraints:

$$\text{wrt.} \quad \sum_{l \in L} cl_l \leq L_{max}$$
$$\sum_{a \in A} cp_a \leq P_{max}$$

Next, we need to make sure that $d_{a,l}$ is always 0 whenever either cl_l or cp_a are 0—if a LAN/AS is not chosen, its entire line/row should only contain zeros. If both, cl_l and cp_a, are set to 1, we want $\omega_{a,l}$ to be arbitrarily large (the more sessions can be obtained, the better). To represent this circumstance we introduce a "large enough" number, B, and formulate the following constraints:

$$\forall a \in A: \quad \sum_{l \in L} \omega_{a,l} * d_{a,l} \leq cp_a * B$$
$$\forall l \in L: \quad \sum_{a \in A} \omega_{a,l} * d_{a,l} \leq cl_l * B$$

For our calculations, we set $B = 10^{10}$ which is multiple orders of magnitude larger than the sum over all entries in the session matrix S. Using this ILP formulation, we can now calculate the maximum number of sessions that can be obtained for at most P_{max} providers when connecting to at most L_{max} peering LANs.

5.2.3.4 ILP Formulation: Peering Scenario

In our second scenario, we assume that the adversary chooses multiple settlement-free peers as injection ASes and then joins peering LANs to establish additional sessions with them. This case differs from the previous one, as routes are no longer propagated globally but rather only into the customer cone of the injection AS. We reuse the notation from § 5.2.3.3.

While we already defined the funneling degree, FD_a, of an AS $a \in A$ in § 5.2.3.1, we need to extend this concept to incorporate the number of sessions that can be established with the injection ASes. We can calculate the Session-Multiplied Funneling Degree (SMFD), $f^P_{a,l}$, for AS a using only injection ASes in $I \subset A$ that are present at peering LAN l:

$$f^I_{a,l} = \sum_{i \in I} \omega_{i,l} \cdot \mathbb{1}_{PF_a}(i)$$

where $\mathbb{1}_Y(x)$ represents the indicator function that returns 1 if $x \in Y$ and otherwise 0.

Parameters. After calculating $f^I_{a,l}$ for each (peering LAN, ASN)-pair, we build the matrix F as our first parameter:

$$F = \begin{pmatrix} f^I_{a_1,l_1} & f^I_{a_2,l_1} & \cdots & f^I_{a_{|A|},l_1} \\ f^I_{a_1,l_2} & f^I_{a_2,l_2} & \cdots & f^I_{a_{|A|},l_2} \\ \vdots & \vdots & \ddots & \vdots \\ f^I_{a_1,l_{|L|}} & f^I_{a_2,l_{|L|}} & \cdots & f^I_{a_{|A|},l_{|L|}} \end{pmatrix}$$

We also provide the parameters $R \in \mathbb{N}$ and $N \in \mathbb{N}$ and a set of potential injection ASes, I. R describes the required SMFD to count an AS as *fully affected*, and N describes the required number of fully affected ASes.

Variables. We add two binary decision variables, $d_l \in \{0,1\}, l \in L$ and $c_a \in \{0,1\}, a \in A$; d_l determines whether the adversary should participate at peering LAN l while c_a tracks whether the current peering LAN selection introduce a session-multiplied funneling degree of at least R for AS a.

ILP Problem Formulation. Given I, F, N, and R, our goal is to minimize the resources—i.e., the number of peering LANs with which we have to establish a connection—needed to perform the Kirin attack, i.e., our objective function is:

$$\text{minimize} \quad \sum_{l \in L} d_l$$

Every valid solution should have a least N fully affected ASes. Hence, we first add this constraint:

$$\sum_{a \in A} c_a \geq N$$

Next, we want to assure that the combined SMFD (across all chosen LANs) of an AS is larger than R for at least N many ASes. Here, we utilize the fact that at least N many c_a variables are set to 1 (by the previous condition) while all other are set to 0. When we multiply R by c_a we effectively generate a switch that either does nothing or conditions the session-multiplied funneling degree of a to be larger than R. As the

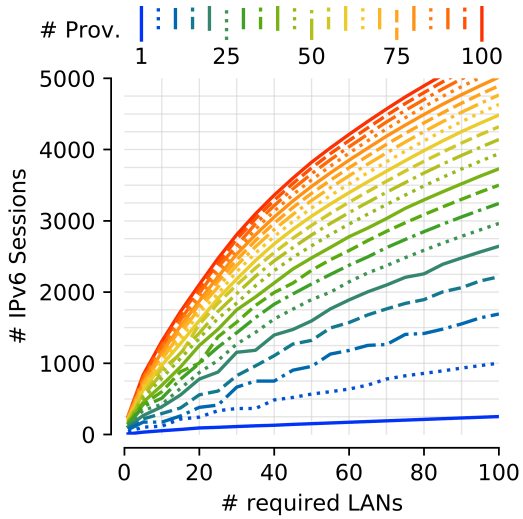


Figure 5.19: Transit Scenario: trade-off landscape.

described condition works only for a single AS, we have to add it once for each AS:

$$\forall a \in A: \quad \sum_{l \in L} d_l * f^I_{a,l} \geq R c_a$$

Notably, this formulation does not incentivize the ILP solver to arrive at the solution with the largest number of set c_a variables—each solution that sets at least N of them is seen as equally good by the solver.

5.2.3.5 Analysis & Results

Now that we have formulated our two models, we can run an ILP solver with varying input parameters to explore Kirin's cost-benefit trade-off landscape.

Implementation and Execution. We implement the ILP program using Python3's PuLP library [394]. We configure PuLP to use the CBC C++ solver [123] and time out (i.e., return the current best, potentially sub-optimal solution) after three hours. We refine sub-optimal solutions whenever possible, i.e., when an optimal run with stricter requirements produced a better objective value than a sub-optimal run, we copy the results from the optimal run over to the sub-optimal run.[84]

Transit Scenario We solve the ILP problem defined in § 5.2.3.3 for L_{max} and P_{max} values between 1 and 100 and obtain the maximum number of sessions that can be established using each pair. Figure 5.19 shows different lines for the number of transit providers (P_{max}), the number of peering LANs (L_{max}) on the x-axis, and the resulting number of obtainable sessions on the y-axis.

Peering Scenario We first observe that we can establish more than a thousand transit sessions by choosing 20 providers and join 25 peering LANs. Given the many possibilities

[84] e.g., when you need X peering LANs to affect 1000 ASes, you do not need more than X to affect 900 with otherwise identical configuration.

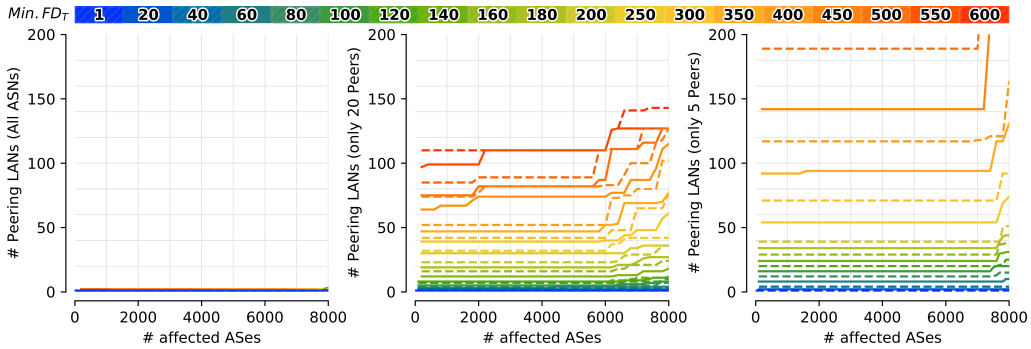


Figure 5.20: Peering Scenario: trade-off landscape for I_{all} (left), I_{20} (middle), and I_5 (right).

to remotely connect to a peering LAN as well as the cheap (in fact, often free) IPv6 transit options, deploying such an infrastructure is not a major hurdle. If each sessions allows us to send 1000 prefixes (which is not uncommon for transit sessions), this setup would already allow us to inject 1M routes into the global routing table.

We further observe that we need to contract at least 35, 45, and 60 transit providers while joining at least 40, 60, and 80 peering LANs to establish 2000, 3000, and 4000 sessions via just a single port per peering LAN, respectively. While certainly harder to achieve, these scenarios are not out of reach for, e.g., state-backed adversaries.

We solve the ILP problem defined in § 5.2.3.4 for different required SMFDs (R), required fully affected ASes (N), and three different sets of injection ASes (I). We first choose I_{all} to be the set of all IPv6-enabled ASes, which corresponds to setting up a bi-lateral peering link with each AS that participates at a peering LAN. While accomplishing this connectivity setup is unrealistic for new and small ASes, it provides us with a lower bound for the number of needed peering LANs. After that, we choose restricted sets of injection ASes, i.e., a scenario in which the adversary convinces a limited number of ASes to setup bi-lateral peering. In this scenario, choosing peers with large customer-cones and many sessions is the most ideal; hence, we rank ASes by the product of their customer-cone size and their total session count across all peering LANs and then choose the top 5 and top 20 ASes to represent the injection sets I_5 and I_{20}, respectively.

Figure 5.20 shows the resulting trade-off landscapes for I_{all} (left), I_{20} (middle), and I_5 (right). Each subplot shows the number of fully affected ASes (N) on the x-axis, different curves for the minimal required session-multiplied funneling degree (R), and the resulting minimal number of required peering LANs on the y-axis. The I_{all} subplot shows that if an adversary could establish bi-lateral peering connections to all ASes at IXP LANs, connecting to a single (or few) peering LAN(s) is sufficient to generate $R = 600$ for 8000 (and probably more) ASes. If the adversary can only establish peering with the injection ASes in I_{20} or I_5, it is realistic to connect to enough peering LANs to introduce $R = 200$ for 5000+ ASes, yet further increasing the required session-multiplied funneling degree might become a significant obstacle.

While a real adversary would realistically arrive at a setup somewhere between I_{all} and I_{20}, properly representing the full spectrum of possibility, which is probably highly dependent on case-by-case, non-technical aspects (e.g., access to the right contacts, marketing, justification of need, "prestige" in the operator community, etc.), goes beyond the scope of this section. Yet, our analysis shows that running Kirin *solely* based on peering connections—which often have max-prefix limits of ~100—seems unrealistic. This insight

is further substantiated by our experiments in § 5.2.5.2 which show that announcements via bi-lateral peering sessions do not necessarily propagate to all ASes within a peer's customer cone, which means that our calculated SMFDs are likely overestimates.

Discussion & Feasibility While it is unlikely that an adversary acquires enough sessions via bi-lateral peering alone, we demonstrated it is possible to get thousands of sessions from various transit providers. Notably, our analysis took a very conservative approach for estimating the session count. In reality, an adversary could use 5, 10, or even more different VPS and remote peering providers simultaneously to establish multiple ports at each peering LAN, which would provide a linear multiplication factor to the number of sessions that can be established. Hence, a highly motivated adversary could potentially end up with 10k+ sessions, most of which capable to reach a significant portion of the IPv6 routing ecosystem. Even if each session would be tightly limited to 100 prefixes, such a setup could still produce an increase of 1M prefixes; hence, we conclude that performing Kirin is clearly feasible.

5.2.4 Testing Router Behavior

As the "512k day" in August 2014 (as well as its successors) received substantial media coverage [2, 158], router vendors are well aware of the possibility and potential impact of exceeding a router's available RIB or FIB memory. In this section, we examine how routers react to a large number of announced non-aggregatable IPv6 routes.

We perform our evaluation in our testbed with one popular enterprise router—the Juniper MX5 [252]—and one virtual version of a popular core router—the Cisco Virtual Router XRv9k [110]. We use ExaBGP [160], a stateless BGP speaker, to quickly announce a large number of routes from a measurement machine to each of the two routers and assess the impact of those announcements over time. We devise two different scenarios for our experiments: (1) the best-case scenario (from the victim's perspective), where each route contains the shortest possible AS path (i.e., a single AS, resulting in a path length of 1) and no BGP communities attached at all; (2) the worst-case scenario, where each route contains the longest possible AS path and maximum number of large BGP communities[85]. For both AS numbers as well as BGP communities we choose 32 bit values to maximize the impact on router memory. For the hardware and the virtual router we use a minimal configuration whenever possible. The Juniper MX5 does not have any prefix limit configured by default, while the Cisco Virtual Router XRv9k has a default prefix limit of 524,288 for IPv6 [113]. We increase XRv9k's prefix limits for our experiments. Note that these prefix limits do not make Kirin infeasible (cf. §5.2.2.2), in fact they can be easily circumvented by announcing prefixes over multiple sessions. While we continuously announce new routes via ExaBGP, we monitor the resource usage of the system under test.

5.2.4.1 Juniper MX5

We begin our testbed experiments with the Juniper MX5 router. In Figure 5.21 we show the results of our memory exhaustion experiments. In the best-case scenario, the router is able to accept around 2.04 million prefixes, before running out of memory. In the

[85]The maximum possible AS path length and number of BGP communities that can be sent with ExaBGP is 251, even though the BGP [405] and BGP large communities [207] specifications allow even longer path attributes.

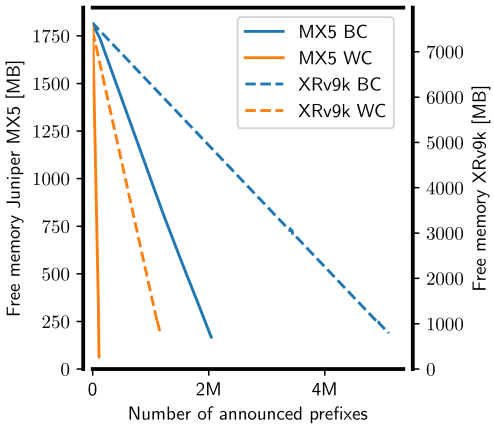


Figure 5.21: Juniper MX5 and Cisco XRv9k memory exhaustion for best-case (BC) and worst-case (WC) announcements.

worst-case scenario this number drops to a low 109k prefixes—which is substantially lower than the current number of all announced IPv6 prefixes (164k) [217].

Once the router's memory is exhausted it will trigger an out-of-memory exception, which results in the BGP routing process being killed. This results in a core dump of the routing process[86], a complete loss of all established BGP sessions, and a purge of all entries in the RIB and FIB.

5.2.4.2 Cisco Virtual Router XRv9k

Next, we perform our experiments with the Cisco Virtual Router XRv9k. We show the results of our memory exhaustion experiments in Figure 5.21. In the best-case scenario, the virtual router accepts slightly more than 5 million prefixes before running out of memory. In the worst-case scenario, it only accepts around 1.16 million prefixes.

The virtual Cisco router deploys different levels of memory alerts [437]. (1) a *minor alert* is triggered at 85% memory occupancy which leads to rejection when trying to establish new eBGP sessions, whereas already established sessions are not affected. (2) a *severe alert* is raised at 90% memory usage and at that point the BGP daemon shuts down already established eBGP sessions until the memory threshold becomes minor. The daemon shuts down BGP sessions with the lowest percentage of best paths selected (# best paths from peer/# total paths from peer). (3) a *critical alert* will be triggered at 95% memory usage, which leads to a shutdown of all established BGP sessions. In our experiments we trigger all of these alerts sequentially, leading to a complete shutdown of all established BGP sessions.

[86] Interestingly, the core file can be so large that it leads to the `/var` directory on the router becoming full, which can not be written to anymore, unless cleaned manually.

5.2.4.3 Theoretical Lower Bound Memory Usage

We can also calculate the lower bound RIB memory usage of our worst-case announcements as follows:

$$MEM = \begin{array}{l}(PREFIX\_SIZE + (255 \times ASN\_SIZE) + \\ (255 \times COMM\_SIZE)) \times NUM\_PFX\end{array}$$

Assuming a prefix size of 16 bytes for the IPv6 prefix and 1 byte for the IPv6 prefix length, an ASN size of 4 bytes, and a BGP large community size of 12 bytes, we get a lower bound of $MEM = 4097 \times NUM\_PFX$, i.e. every worst-case prefix needs at least 4kB of RIB memory. Given that today's core routers (e.g., Cisco ASR 9000, Juniper MX960, or Arista 7280CR2K) have RIB memory sizes of 32 or 64 GB, a large number of worst-case prefixes can still bring a router with lots of memory to its knees: 8M prefixes—which can be obtained from de-aggregating a /26—suffice to fill up 32 GB. Finally, as the IPv4 Internet is approaching the 1M route threshold [223] and the increasing deployment of technique such as RPKI [466], fewer routes from an attacker will be needed to further fill up a router's RIB memory.

Takeaway 1: *Enterprise routers can already be overwhelmed with as little as ≈ 100k announcements, whereas core routers can at least handle around 1M. In the worst case, a route needs at least 4KB router memory to be stored.*

5.2.5 Real-World Experiments

Due to ethical concerns as well as economical and social consequences, we can not simply perform a large scale attack on the Internet to provide a proof-of-concept. Instead, we opt for multiple small-scale experiments that provide interlocking insights into the viability of different parts of the attack.

5.2.5.1 Obtaining Resources and Connectivity

We state in Section 5.2.2.2 that it is fairly easy to (1) receive the resources needed to execute the proposed attack, (2) join multiple IXP peering LANs, and (3) establish additional sessions to large transit providers. Below we report on our experience in building and operating a proof-of-concept network capable of performing a small-scale Kirin attack at negligible cost.

Internet Resources. We obtained an AS number (AS39282) and a few IPv6 address blocks (2a10:cc47:100::/40, 2a0e:b107:e80::/44, and 2a10:2f00:15d::/48) through a sponsoring LIR (Securebit), at a total cost of 270 EUR (valid for 1 year). It took only a few days from requesting these resources until obtaining them for use on the Internet.

Takeaway 2: *It is possible to obtain ASNs and IP prefixes within days and at a low cost.*

Peering LANs. We built our proof-of-concept network using 2 VMs with IXP access: one in Frankfurt (provided by vServer.site) and another in Dusseldorf (provided by Securebit). This allowed us to directly access all route servers and peering LANs of 4 medium-to-large size IXPs: DEC-IX, NL-IX, KleyReX, and LocIX. In total, we paid an initial setup fee of 160 EUR and a monthly operating fee of 60 EUR. It took a day till we connected to the first IXP and a few weeks until we connected to the last IXP.

Takeaway 3: *IXP connectivity providers let new ASes quickly join many peering LANs at a small cost.*

Transit Sessions. We decided to use Hurricane Electric (HE, AS6939) as our main transit provider, as it is one of the most important IPv6 networks [247]. Surprisingly, HE reached out to us about setting up bilateral peering sessions at our IXPs—with a free IPv6 transit option—before we even knew the IXP on-boarding process finished. Additionally, we obtained a VM in Amsterdam from Vultr (AS20473), which provides BGP transit to its customers at no additional cost. We paid no setup fee and a monthly operating fee of 5 USD. The VM was available in a few minutes.

Takeaway 4: *It is possible to instantly get cheap IP transit.*

Prefix Limits. After finding out our sessions have low prefix limits, we asked if our providers could raise them. As a result, in less than 24h, most operators increased the limits by an order of magnitude without asking for explanation. Other operators stated they could arbitrarily raise the limits given a reasonable justification.

Takeaway 5: *Increasing prefix limits is a matter of asking, and often requires no justification.*

5.2.5.2 Propagating Announcements

Below we take a closer look at the routing ecosystem itself. In particular, we analyze the correctness of the claims we made earlier in Sections 5.2.2 and 5.2.3. We use the infrastructure described in the previous subsection and the PEERING testbed to run real-world experiments for a limited number of ASes and contrast our findings with insights obtained from the routing information captured by the route collector projects.

Setup Specifications We make use of the proof-of-concept network that we built in the previous subsection to produce IPv6 route announcements. Besides the thousands of (implicitly gained) multilateral peering sessions via route servers, our network only has few direct sessions (most of which connect to HE). To improve our coverage of large IPv6 transit providers and, thereby, improving the generalizability of our results, we also utilize the PEERING testbed [444, 448]. The PEERING testbed is a research network that allocates resources (i.e., ASNs and prefixes) to submitted and accepted project proposals. It has 207 direct IPv6 sessions to 150 different networks distributed across 9 physical locations as well as dedicated IPv6 sessions to 12 route servers at 5 IXPs. All announcements from the PEERING testbed were originated from AS 47065 and sourced from the 2804:269c:10::/44 IPv6 address block. In addition to the standard project capabilities we received the additional capability to announce BGP communities that control the redistribution behavior of the connected route servers.

Announcement Schedules. We announced a dedicated /48 IPv6 prefix via each session. As we control fewer unique /48 prefixes than we have sessions, we first organize the sessions into groups and then reuse the same prefixes across groups (but not within each group). To substantially reduce the likelihood that two successive groups are influenced by one another (e.g., as the first one triggers Route Flat Damping), we adopt a two hour announcement schedule—we announce all prefixes within a group, then wait 30 minutes for route convergence, then withdraw all prefixes, and then wait another 90 minutes before repeating the cycle with the next group. While, e.g., MRAI timers [181] or similar update minimization techniques may introduce few minutes of delay to the propagation of our our announcements, we have to wait additional 60 minutes in the last step to ensure that accidentally triggered Route Flap Damping penalties expire [192] and can hence no longer influence the next group of announcements.

Routing Information. We utilize the route collector projects RIPE RIS and Routeviews as

our vantage points. In total, they operate 47 IPv6-enabled route collectors that connect to 305 full-feed ASes via 555 IPv6 sessions. For our analysis, we utilize all available RIB snapshots at 2022-09-26, 00:00 UTC+0 using the BGPStream utility.

	Routes	Paths	Prefixes
Total	58.2M	13.9M	223K
AS set	12K (0%)	10K (0%)	57 (0%)
ATOM.	4.2M (7%)	1.0M (7%)	161K (72%)
AGGR.	5.1M (8%)	1.3M (9%)	16K (6%)
Any Hint	6.4M (10%)	1.6K (11%)	162K (72%)

Table 5.1: Results of aggregation analysis.

Route Aggregation In this first experiment, we announce pairs of aggregatable routes via all our transit providers, i.e., HE at our infrastructure and 7 different transit providers at the PEERING testbed. We repeat this experiment twice. The first time, we announce two consecutive prefixes (i.e., A:B:C::/48 and A:B:C+1::/48) via each session. As both routes are entirely identical, a transit network may decide to aggregate these two routes and only redistribute the resulting /47 route that covers both announcements. The second time, we announce a /47 covering prefix and the /48 sub-prefix with the same network address (i.e., we announce A:B:C::/47 and A:B:C::/48 but not A:B:C+1::/48). In this scenario, a transit AS may decide to not redistribute the more-specific /48 route given that the AS path is identical. While we see all announcements propagate globally (i.e., each prefix is seen by at least 95% of all route collector peers), we see no signs of aggregation.

Analysis. When an AS aggregates a route, it may leave up to three clues in the BGP messages that it redistributes. First of all, AS paths may consist of AS sequences and AS sets [405]. A set is generated whenever two routes with different AS paths are aggregated; they represent a summary of the non-matching parts of the two initial AS paths. If an AS aggregates a route and generates no AS set during this process, it should add the `ATOMIC_AGGREGATE` attribute to the message. Finally, an AS may set the `AGGREGATOR` field to indicate that it produced this route aggregate. We searched all IPv6 routes seen by the route collectors for these three hints and display our findings in Table 5.1. While we observe that 72 % of prefixes have at least one path with an aggregation hint, we only observe 11 % of paths and 10 % of routes with aggregation hints; hence, we believe that only few ASes actively perform route aggregation. While we did not find any signs of route aggregation during our own experiments, an adversary could also make routes less aggregatable by announcing neither neighboring nor covering prefixes to the same neighbor, and alternating the origin AS.

Takeaway 6: *While aggregation is a theoretical challenge, it is rare in practice and can be circumvented.*

Route Redistribution Next, we want to analyze whether our assumptions for the route propagation behavior of transit, bi-lateral, and multi-lateral sessions are accurate. While the number of transit providers for both testbeds is limited, applying our schedule to all bi-lateral and multi-lateral peers connected to the PEERING testbed would require extensive amounts of time; hence, we select a smaller set of important ASes.

Tested Networks. The importance of a network for our attack can be characterized by two dimensions: the number of sessions we can establish with it, and the number of networks it redistributes our announcements to. Figure 5.22 shows the customer cone

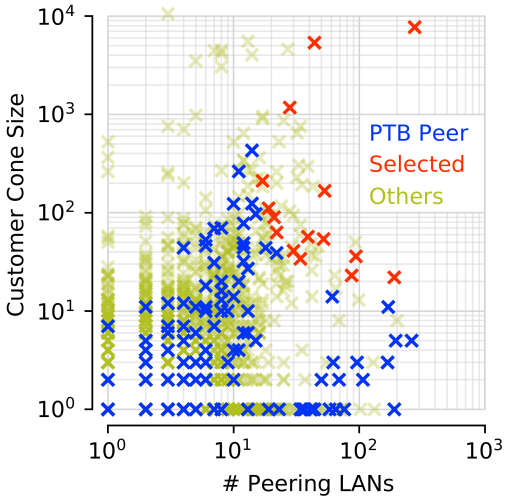


Figure 5.22: PEERING testbed peers: customer cone vs. peering LANs.

size (y-axis) against the number of peering LANs to which a network is connected (x-axis) as a scatter plot for all networks with PeeringDB entries. We mark networks that connect to the PEERING testbed in blue ("PTB Peer") or red ("Selected") and all other networks in green ("Others"). As both dimensions are equally important to Kirin, we select the 15 PEERING peers with the highest harmonic mean[87] of customer cone size and the number of potential sessions.

Experiment. Figure 5.23 shows the fraction of route collector peers (y-axis) reached by /48 announcements via each of the three different session types (on the x-axis). We calculate this fraction twice: once relative to all IPv6 route collector peers (green, "total") and once relative to the peers within the customer cone of the neighbor to which we announced the prefix (blue, "within customer cone"). We can first verify that announcements towards transit providers always propagated globally and that announcements via multi-lateral peers barely propagates at all. Yet, contrary to our assumption, not a single bi-lateral peering sessions redistributed our announcements into even half of its customer cone. Hence, we likely over-estimated the achievable funneling degrees in § 5.2.3.4, which we already noted in that section.

Analysis. To further test the validity of our transit propagation assumption, we analyze the public BGP data. After removing path-prepending [312], we select all prefixes for which all paths have the same first-hop AS, i.e., that were announced via a single transit provider. Figure 5.24 shows the minimal, median, and maximal propagating route for each of these transit providers as an ECDF. We observe that for 80 % of transit providers every route propagates globally (i.e., to more than 80 % of route collector peers), while for 89 % and 94 % at least the median and best route propagated globally, respectively.

Takeaway 7: *While bi- and multi-lateral peers do not necessary redistribute into their entire customer cones, announcing to a transit provider leads to global redistribution.*

[87] Compared with arithmetic mean, the harmonic mean leans towards lower numbers, which penalizes networks that appear large in only one dimension.

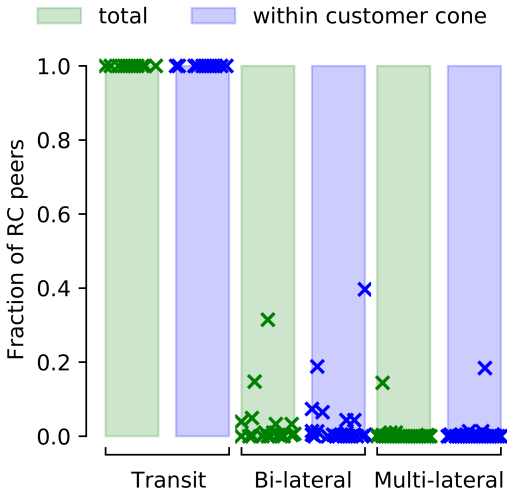


Figure 5.23: Redistribution behavior of different session types.

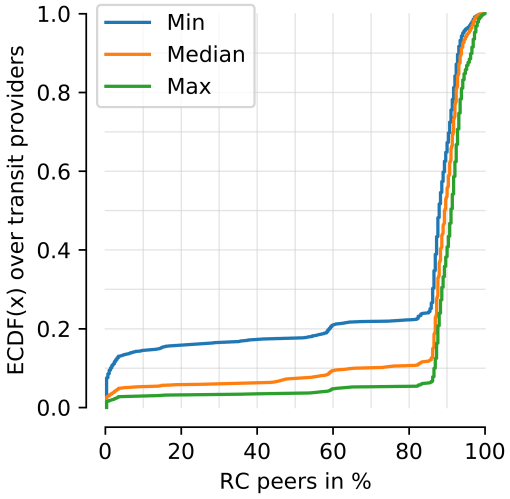


Figure 5.24: Redistribution behavior for transit providers of single-homed ASes.

5.2.6 Discussion

Targetability & Collateral damage. While we introduced Kirin as a global attack, BGP has many mechanisms that allow an adversary to steer the redistribution of a route. Many transit providers allow their customers to directly decide to which neighbors their routes get redistributed by attaching specifically encoded BGP community attributes [69, 96, 478]. In addition to this collaboration-based technique, the adversary may also "poison" the AS path to avoid that certain ASes accept the route. The poisoning method leverages cyclic route filters implemented by most routers: if the adversary A forges a route with the path AXA and this route propagates to AS X, X will likely drop it [258]. As the Internet's routing hierarchy has flattened drastically over the last decade, it is likely that a combination of these two mechanisms could be sufficient to steer routes towards most regional networks. Yet, even if the adversary succeeds in steering the majority of the attack towards a single AS, the increase in redistributed routes at the intermediate ASes should still be very noticeable providing an opportunity to detect the attack and limit the redistribution.

Detection & Mitigation. Kirin is easily detectable as it introduces multiple times more routes than the current IPv6 routing table contains in total. Hence, even operators that do not monitor their own network could detect it by checking Twitter notifications from the IPv6 routing table size bot or the BGPStream bot [109, 130]. Once some operator detected the attack and shared the origin ASes involved in it via some high-visibility operator mailing list such as NANOG, the attack can be mitigated by adding simple ingress filters for the covering prefix and origin ASes. Once these filters are added, the routers no longer import any routes related to the attack which should prevent the router from running out of memory and also drastically lower the CPU load. Due to the simplicity of the mitigation process, Kirin's attack duration is effectively limited to how quickly network operators (especially those of intermediate ASes) can co-ordinate the mitigation efforts—a time that we hope to substantially reduce by raising awareness via this work and our carefully designed disclosure process.

Traceability & Repercussions. Kirin's resources are easily traceable to the RIRs that

allocated them and, from there, might be directly accountable to a specific person or organization. While this seems like a large issue, there are no real sanctions or direct repercussions for "routing vandalism." Bitcanal illustrates this issue nicely: besides loosing some "reputation" via call-outs from researchers and operators [451, 487], Bitcanal continued to frequently hijack the resources of other ASes over multiple years, until Spamhaus added all related ASNs to their "Don't route and peer" list [474].

5.2.6.1 Potential Defense Mechanisms

While Kirin can be mitigated quickly, we ideally would like to entirely prevent it from being feasible. Yet, based on its distributed nature, there is no simple solution that fully prevents the attack; however, there are multiple technical and non-technical mechanisms that may limit Kirin's impact or increase its requirements.

Dynamic Yet Tight `Max-Prefix` Limits. Transit providers should introduce dynamically growing yet tight per-session limits on their eBGP sessions. We recommend to allow customers and peers to announce at most 1.5x the number of prefixes they announced the previous day. Similarly, the IPv6 routing table currently grows <50k new prefixes per year [217]; hence, we further recommend to allow a maximum daily increase of at most few thousand prefixes on transit sessions. Automatic imports of max-prefix limits from, e.g., PeeringDB should be sanity checked and not be allowed to surpass a certain pre-defined limit—otherwise adversaries could enter arbitrary high numbers and abuse the prevention automation.

Per-Origin and Per-Block Prefix Limits. We highly recommend transit providers to stop the redistribution once too many routes within the same covering prefix or by the same origin AS are announced. While covering prefixes would optimally be determined by analyzing the daily IRR delegation files, counting on a /29 or /32 basis might be easier to implement. Currently, the AS with most announcements is AS9808 with 3870 IPv6 routes and the covering prefix with the most more-specific announcements is 2409:8000::/20 with 9807 more-specifics. As implementing these limits on each router is costly (and may still be insufficient if different routers receive unique sets of routes), we highly recommend (if available) to introduce these limits on a route reflector.

Tight Resource Monitoring & Filtering. We recommend transit providers to monitor the number of sessions that other ASes establish with them—especially if their peering policy is fully open or they employ a fully automated session establishment service. If they automatically generate filter lists from third party data sources (e.g., RPKI [304], IRR [325], or Team CYMRU [128]), we recommend them to carefully monitor the resulting filter size; checking the number of acceptable prefixes may reveal the preparation for a Kirin attack early. Further, we recommend transit providers to only redistribute what is correctly registered and avoid loose filtering, i.e., do not assume that more-specific versions of route objects or ROA records are valid by default. While this will not directly prevent the attack, it will increase the effort on the adversary's side to register the resources correctly.

Delayed Propagation of Unfamiliar Routes. The concept behind Pretty Good BGP [257] is to avoid propagation of anomalous routes, not seen in a window of historical data. Thus, the use of previously unseen routes is delayed, with the hope of identifying and neutralizing any attacks in the meantime, if the route was really malicious. In the context of Kirin, if the attacker used a hijacked prefix, the idea presented in that paper could stop the attack from propagating further, yet note that prefix history tracking needs memory anyway. On the other hand, as Kirin does not need IP prefix hijacking, the attacker can use a large pool of addresses that has never been announced on the Internet before.

Thus, we suggest modifying the Pretty Good BGP idea so that it *also* delays accepting announcements of new prefixes that are not contained in already propagated, larger address blocks.

5.2.7 Ethical Considerations

The attack framework we present in Section 5.2.2 has the potential to cause serious harm. Hence, our research naturally raises several ethical concerns, which we discuss in this section.

Real-world Experiments. While we performed a thorough theoretical evaluation of our attack's potential impact (Section 5.2.3) and assess the behavior of various hardware implementations exclusively in a non-Internet-connected lab environment (Section 5.2.4), the Internet is a dynamic system, and—given the issue of deaggregation is well known—might not be susceptible to the attack after all. Hence, we also conducted real-world experiments, see Section 5.2.5.

When designing our experiments, we closely followed the Menlo report [261] and work by Partridge and Allman [377] to mitigate potential harm to the Internet. This includes a thorough harm benefit analysis after assessing the theoretically possible impact. Here, we weighted that the underlying techniques of our attack are generally known in the community, and it is likely that others with potentially malicious intentions may independently develop our scaling methods for prefix deaggregation. At the same time, the networking community considers existing techniques—like per session prefix limits—sufficient to mitigate the threat, and is unlikely to consider our attack as serious and implement prevention mechanisms, unless the feasibility can be practically demonstrated.

Hence, we decided to conduct a small-scale experiment using 500 prefixes via Vultr. Given the size of the IPv6 routing table (~160k prefixes), we believe that that these 500 prefixes (~0.3%) were well inside the daily BGP IPv6 table size churn. Furthermore, we limit the duration of our announcements, made them unlikely to trigger route flapping, and ensured that our announcements are properly withdrawn after we have completed the experiments. In our PEERING testbed experiments we never announced more than 20 IPv6 prefixes simultaneously. Similarly as with Vultr, we ensure proper withdrawal of our announcements in the PEERING testbed.

Independent Reproduction by Unknown 3rd Party. Despite our best efforts to design an experiment that does not cause harm, it was still visible in the global routing table. Six days after we conducted our experiments—which did not cause noticeable load at an independent leaf AS we operate as well—we observed an unknown entity that replicated our experimental setup executing Kirin with over 8,000 prefixes from one /32 via Vultr. This caused noticeable load on the independent leaf AS we operate and was widely noticed in the operator community.

We hence decided to accelerate the initial disclosure process we had planned to take place (see below). Furthermore, it demonstrates that, by now, threat actors are actively monitoring the global routing table. Researchers conducting experiments for potential vulnerabilities in the routing ecosystem *must* consider that even small-scale experiments may reveal attack opportunities to third parties. This leads to substantial problems when the "attack" opportunity is (technically) well-known in the community, yet is currently not considered "exploitable enough" [147].

Disclosure Schedule. After an independent third part potentially replicated our experiments on a significantly larger scale, we immediately launched a two-stage notification

process. While technically, a lengthy coordinated vulnerability disclosure process [215] would have been preferred, also to have more time to carefully discuss with operators *why* this well-known vector is a higher threat *now*, we opted for this path due to the actions of the unknown third party around the 5^{th} of October, 2022 [61].

- **Private Disclosure Phase (2022/10/11–19).** We first disclosed the details of our attack via a whisper-network of well connected Tier-1 network operators and IXPs. In this process, we distributed the document enclosed in Figure C.4. This process included 8 major IXPs, 20 Tier-1 ASes, and 7 major content providers. We followed-up the initial notification with a clarifying statement, highlighting that an independent third party potentially already executed the attack on a larger scale. We received the feedback that this clarification made the severity of the problem apparent.
- **Public Disclosure Phase (2022/10/20 and onward).** After sufficient reaction time and no signals to further delay the disclosure, we publicly disclosed our findings via 13 different operator mailing lists (including, e.g., NANOG, DENOG, and the RIPE Routing Working Group) as well as via different social media platforms.

During our disclosure phases, we continuously discussed our findings with network operators, integrated their experiences, and assisted them in deploying prevention mechanisms whenever possible. From private e-mail exchanges, we know that at least two Tier-1 ASes, three cloud providers, and various smaller networks actively configured prevention mechanisms against our new form of prefix de-aggregation attacks.

5.2.8 Summary

In this section, we presented Kirin, an attack that overwhelms BGP routers by globally distributing millions of IPv6 routes via thousands of distributed sessions. We demonstrated that Kirin can bypasses traditional prevention mechanisms via its distributed nature and showed that its required infrastructure and resources can be obtained swiftly and at a cost bearable even for single individuals. We tested our assumptions in lab experiments, real-world measurements, and by analyzing passively obtained routing information. Finally, we launched a two-stage disclosure campaign to notify network operators and expedite the deployment of prevention mechanisms.

5.3 Chapter Summary

We analyzed the prevalence of AS Path Prepending and found that it is still very present on the Internet, with its utilization slightly increasing over time. Our analysis further reveals that prepending policies are mostly stable over time; that ASes use a wide range of policies when announcing their prefixes; and that prepend sizes are becoming polarized—with either one or more than three prepends. We unexpectedly spot many ASes uniformly prepending (all) their prefixes to all neighbors, hence not influencing any remote routing decision. Our complementary analyses with traceroutes and cross-checks with CDN data confirm that our observations are likely valid despite the limited visibility of public route collector projects. Our active measurements confirm that ASPP is effective—since even small prepend sizes can steer the traffic of multiple routes—if used with many upstreams. When using only two upstreams, ASPP's effectiveness is dependent on the AS location. Yet, we find that ASPP has security considerations as it

can increase the spread of prefix hijacks, as the hijacked route becomes more attractive than its prepended counterpart to a larger fraction of ASes. This issue is emphasized by the observation that for approximately 18 % of the prepended prefixes the prepending does not yield any traffic steering benefits.

We revisited the viability of prefix de-aggregation attacks in the current hyper-connected routing ecosystem. We showed that these attacks may overwhelms BGP routers by globally distributing millions of IPv6 routes via thousands of distributed sessions. We demonstrated that they can bypasses traditional prevention mechanisms when executed in a highly distributed manner and showed that their required infrastructure and resources can be obtained swiftly and at a cost bearable even for single individuals. We tested our assumptions in lab experiments, real-world measurements, and by analyzing passively obtained routing information. Finally, we proposed a set of updated prevention mechanisms and performed a two stage vulnerability notification campaign involving 8 major IXPs, 20 Tier-1 ASes, and 7 major content providers.

Discussion. While we demonstrated that the routing ecosystem's constant evolution has weakened previously deployed prevention mechanisms for prefix de-aggregation attacks, it's key enablers—the routing ecosystem's increase in scale and the vast address space offered by IPv6—may also affect other prevention mechanisms in the future. One explicit example—that was recently discussed by Richter et al. in [411]—is the usage of block-lists to drop traffic from a specific source IP (e.g., to protect against email spam or phishing attacks). While it is technically still possible to block individual IPv6 addresses, IPv6 weakens the association between devices and addresses as it allows end-users to quickly pick new addresses. At the same time, blocking an entire IPv6 prefix is also infeasible as, on the one hand, it is unclear at what CIDR size operators should block while, on the other hand, the blockage of entire network prefixes would incurs large amounts of collateral damage, i.e., legitimate sources with genuine traffic might also be blocked.

Chapter 6
Summary & Future Directions

In this dissertation, we measured the Internet's routing ecosystem, a complex and dynamic system that constantly evolves to accommodate the needs of its tens of thousands of stakeholders and billions of users. This constant evolution challenges operators, academics, and policy-makers when maintaining, optimizing, securing, and regulating networks. In Chapter 3, we first measured our ability to model and track the routing ecosystem and its evolution accurately and found that our observation infrastructure and inference methods tend to be more incomplete, biased, and sensitive to variance than we initially assumed. In Chapter 4, we then explored different ways to handle one of the Internet's recent major changes, IPv4 exhaustion, and found that leased IPv4 address space and hyper-specific announcements are economically and operationally attractive options to bridge the gap until IPv6 deployment has fully matured. Finally, we showed in Chapter 5 that tracking the evolution of the routing ecosystem allows us to identify new security issues and also helps us to understand whether previously deployed prevention mechanisms still provide the defensive properties we expect.

6.1 Summary

Throughout this dissertation, we addressed three major challenges posed by the Internet routing ecosystem's continuous and rapid evolution. First, we focused on the question:

(1) How accurately can we model and track the Internet's routing ecosystem and its evolution with our deployed observation infrastructures and commonly used inference methods?

To address this question in Chapter 3, we initially started with a case study at a large European IXP. We carefully inferred the IXP's peering fabric and found that less than a fourth of its AS links are visible from public route collectors while, in total, the IXP's peering fabric alone contains more AS links than the route collector view. We further observe that at least 19.8, 57.1, and 57.4 % (37.3, 37.4, 37.8 %) of all routed IPv4 (IPv6) address space

can be reached at our IXP via multi-lateral, bi-lateral, and private peering, respectively. Those results provide practical contrast to the 70+ % reachability theoretically estimated by Böttger et al. in 2018. Finally, we observe that many publicly available routes at the route server are unattractive (i.e., they lead to out-of-continent with relatively many AS-level hops) and hence see little to no traffic. Putting these findings into perspective, we find a public route collector view that not only diminishes in completeness but also loses in importance as the visible routes may be similarly unattractive as route server routes (given their length and potential geographic routing detours) and hence may carry lower volumes of traffic.

We then focused on the biases and sensitivity of business relationship inference algorithms. We performed two orthogonal studies: In the first, we evaluated validation coverage and inference performance on varying subsets of AS links (chosen based on common features). This study revealed that our validation data has significant topological and geographical biases and that the inference precision may drop by up to 25 % for certain combinations of algorithms and link classes. This result drastically reduces our confidence in the correctness of inferred business relationships in narrow contexts, e.g., when applied to a single IXP or a small set of ASes. In the second study, we generated tens of thousands of slightly altered (e.g., shifted by two hours) input data sets and compared the inferences that the ASRank algorithm produced for them. While we found that paying attention to these kinds of short-term routing dynamics may uncover algorithmic issues, we further found that it had a non-negligible impact on performance evaluation. Even though we show that only 6 % of AS links are inconsistently inferred, 55 % and 85 % of all inference errors (for the median and worst snapshot) are transient (i.e., appeared only in some snapshots), respectively. To put this result into perspective: ASRank had a $5.4\times$ lower error rate for the best compared to the worst performing input within a three-month window, yet recent works claimed that they improved upon ASRank when only lowering the error rate for a handful of inputs by $1.4\times$. In summary, our two studies suggest that our inference methods for business relationships are more biased and sensitive to short-term routing dynamics than we previously thought.

We acknowledged the multi-dimensional nature of bias and proposed a framework to quantify the AS-level bias of vantage point placements along various dimensions. Our framework not only confirmed previously known biases but further provided new insights, e.g., while highly-peered networks are over-represented as route collector peers, the distribution of their peering policies is representative of the Internet as a whole (according to PeeringDB). We further extended our framework with tools to subsample the existing infrastructure and predict the bias reduction that can be obtained by acquiring a given new AS as a vantage point. Finally, we show that unbiasing the route collector infrastructure can go hand-in-hand with use-case-specific goals, such as reducing the minimum number of hops between origin ASes and vantage points across prefixes.

With the uncovered limitations of our observation infrastructure and inference methods in mind, we focused on one of the most critical routing ecosystem changes throughout the last decade by asking the question:

(2) How can network operators cope with the exhaustion of IPv4 addresses while parts of the Internet still lack sufficient IPv6 adoption?

In Chapter 4, we explored two options to address this question. First, we analyzed at which cost operators may be able to (temporarily) obtain IPv4 addresses via the buying and leasing markets based on data from four major brokers and tens of leasing services.

While we showed that public data sources only provide a spotty view of market size, both markets appear to thrive. In 2020, the cost of an average IP address circulated around $22.50, with little to no difference among regions. In contrast to the buying market, the leasing market has not yet converged with offers ranging from $0.30 to $2.33 per IP per month depending on, e.g., the contractual details of the service, the geographic region, and the purpose of usage. Given that these observations project the amortization time of an IPv4 address to anywhere between ten months and more than six years, leasing agreements may be the economically preferred option to bridge the gap until full IPv6 deployment for some networks.

Besides obtaining additional address space, we further explored the option of announcing the already available address space as smaller, hyper-specific prefixes. While these prefixes are commonly filtered and do not propagate globally, we can observe tens of thousands of them in the routing ecosystem. We found that hyper-specific prefixes most commonly represent blackholing and peering subnet announcements in IPv4, whereas they are primarily associated with blackholing and address re-assignments in IPv6. Despite not redistributing them, many larger networks accept hyper-specific prefixes internally. This characteristic may popularize the usage of hyper-specific prefixes as the Internet's routing ecosystem continues to flatten.

After this look at one of the more prominent changes, we moved our focus to our final question:

(3) How does the evolution of the routing ecosystem affect the security of routing operations?

To approach this question in Chapter 5, we first focused on an easy-to-use, well-known traffic engineering technique: AS path prepending. We first measured the usage of path prepending and the frequency of different prepending sizes over time in the wild, then identified common usage patterns and discussed and experimentally tested their traffic engineering benefits and potential security issues. Hereby, we took special care of our previous findings and validated our results extensively with additional active measurements and the help of three CDNs. We found that the usage of AS path prepending, despite already being high, is steadily increasing. While we found that prepending is effective in many scenarios, we also found that around 18 % of prepended prefixes employ a uniform prepending policy, i.e., they have the same prepending policy on all paths. While this policy may only achieve traffic steering effects in rare corner cases (e.g., when a neighbour uses a route optimizer that ignores prepending), it equally de-prioritises all paths, effectively aiding the spread of BGP hijacks. In this case, the routing ecosystem's evolution aided the secure operation of AS path prepending: The introduction of additional networking infrastructure and IXPs over the last decade led to shorter prepending sizes, making it harder for adversaries to take advantage of currently deployed prepending policies.

Next, we wanted to stress the security impact yielded by the evolution of the Internet's routing ecosystem. Hence, we focused on the synergies of two major recent changes—the rapid deployment of peering infrastructures and the vastness of the available IPv6 address space—and how these changes may be exploited to orchestrate globally distributed prefix de-aggregation attacks. Using a mix of a theoretical Integer Linear Programming formulation, the deployment of a real-world testbed, various BGP data analyses, real-world route propagation measurements, and router testbed experiments, we demonstrated that these attacks are capable of bypassing deployed prevention mechanisms and can be arranged at relatively low cost and effort. To prevent the misuse of our findings,

we proposed a set of updated prevention mechanisms and performed a two-stage vulnerability notification campaign involving eight major IXPs, twenty Tier-1 ASes, and seven major content providers. Our results clearly indicated that the impact of the routing ecosystem's evolution is unpredictable—while it may aid secure routing operations, it also has the potential to impede them substantially.

6.2 Discussion & Future Directions

Internet Modeling. Our findings from Chapter 3 paint a rather dark picture for the future of tracking the Internet's hierarchy. As the flattening of the Internet's hierarchy continues, our visibility and ability to track trends will ultimately diminish. While our current solution—the acquisition of additional vantage points—likely slows this blinding process down, it not only introduces storage issues but also makes it hard to actually derive insights from the large volume of raw data. To not lose our ability to track the Internet's routing ecosystem accurately, we should clearly define the use cases for which we want to use route collectors and, based on those, develop tailored data capturing and compression approaches.

Further, many promising avenues exist for improving our inference of business relationships. From a conceptual perspective, we should infer relationships on a more fine-grained level than the AS level and also embrace the diversity of real-world business relationships beyond the three main abstractions of peering, transit, and sibling relationships. From a practical perspective, we should put more effort into evaluating these models as we showed our best current practices to be both biased and sensitive to short-term routing dynamics. Finally, from a data perspective, we should stop aggregating routing information over time and sessions before starting the inference process. This aggregation step introduces conflicting information and may be avoided by inferring relationships in each session after each update. In this way, we could identify incoherences across sessions and time. This would allow us to handle these cases appropriately (e.g., using the orthogonal information available in BGP communities) and gain a deeper understanding of our inference algorithms' underlying issues.

Addressing. Despite our findings from Chapter 4, bridging the gap between IPv4 exhaustion and full IPv6 adoption remains challenging as different networks have unique business environments that often prompt individually tailored solutions. While IPv4 prices are still rising, IPv6 adoption will eventually render IPv4 the 'legacy' protocol, ultimately leading to a price crash. This dissertation aided in the process of making this market behaviour more transparent to all involved parties as various brokers (e.g., Brandergroup, IXPO, and IPv4marketgroup) started to openly disclose their pricing information on regular intervals after we first presented actual pricing information from multiple major brokers in our 2020 study.

While IPv6 adoption has come a long way over the last two decades, global adoption still remains a major challenge. Additional research is needed to identify the areas that primarily lack adoption, deeply understand the factors that impede adoption in those areas, and explicitly tailor incentives and solutions to those areas—a journey that requires joined efforts from the operator and academic community. Until IPv6 is fully adopted, the coexistence of IPv4 and IPv6 poses multiple challenges:

First, IPv6 is rarely deployed independently but is part of dual-stack deployments. Once IPv4 has become the 'legacy' protocol, these dual-stack deployments are wasteful regarding processing power, available memory, and maintenance efforts. Hence, a set of

questions arises: How can we reduce (or entirely remove) IPv6's current "dependency" on IPv4 to deslag our future networking infrastructure? What are the major pitfalls of a separated deployment approach, and how could these be mitigated? How transparent does this transition have to be to upper layers or users?

Second, the IPv6 routing table size currently increases almost exponentially, with approximately 25k new prefixes being added globally across 2021. Over the next decade, RIBs that contain IPv4 and IPv6 routes from many peers may exceed currently available memory sizes and increase the processing overhead. This issue would be further exacerbated by the increasing popularity of hyper-specific prefix announcements. In fact, larger networks such as PCCW not only accept them but also redistribute these prefixes already. Hence, the current evolution of the routing ecosystem may potentially lead to a natural RIB memory exhaustion.

Secure routing operations. In general, routing security was more an afterthought than a fundamental principle for the design of BGP. This resulted in a lack of authenticity and legitimacy verification and a continuously increasing patchwork of best common practice documents that detail how to identify, mitigate, or prevent specific security issues. In Chapter 5, we demonstrated that the routing ecosystem's change could positively and negatively impact the security of routing operations and the strength of prevention mechanisms. While we specifically focused on prefix de-aggregation attacks, the new opportunities that IXP interconnections and IPv6 introduced into the routing ecosystem may likely weaken the protection of more current prevention mechanisms. While this observation would require us to check and potentially improve all best common practice documents, it is unrealistic that this due diligence is meticulously performed, potentially leaving us open to routing-based attacks in the future.

Appendix A
Modelling the Routing Ecosystem

A.1 Case Study: Alternative Traffic Metrics

This appendix section provides the interested reader with variants of the plots provided in section § 3.1 using alternate metrics (e.g IPv4/6 Bytes or Packets). Notably, the conclusions drawn in the section are equally supported by these plots.

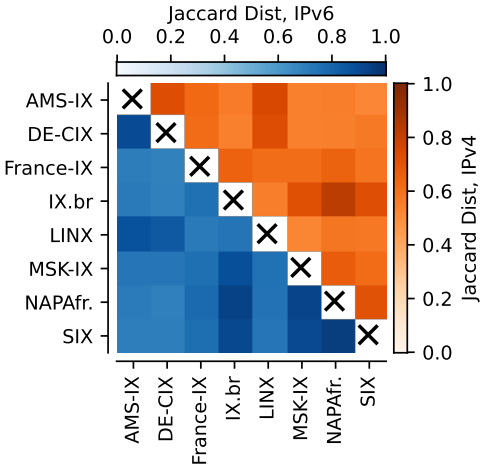


Figure A.1: Similarity of addresses between Route Servers without HE's 2002::/16 route

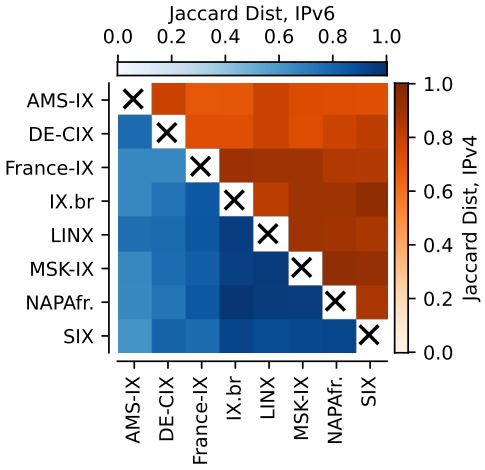


Figure A.2: Similarity of prefixes between Route Servers for common peers

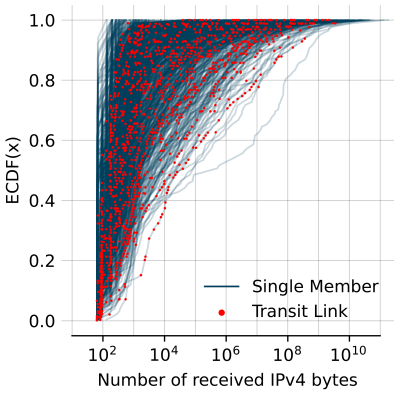


Figure A.3: ECDF for the number of received IPv4 Bytes per member

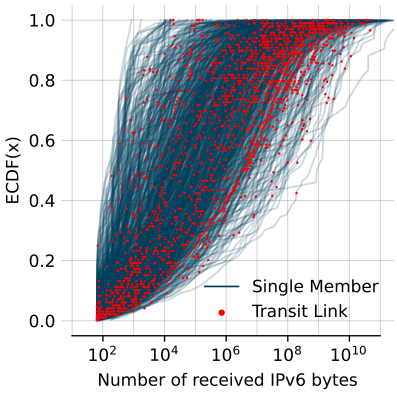


Figure A.4: ECDF for the number of received IPv6 Bytes per member

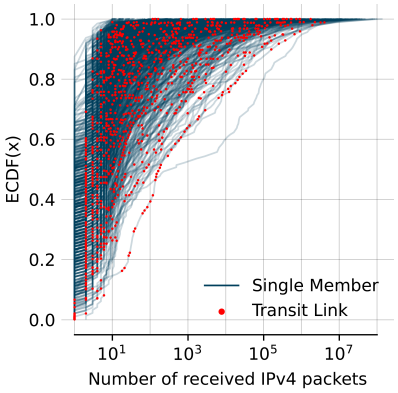


Figure A.5: ECDF for the number of received IPv4 Packets per member

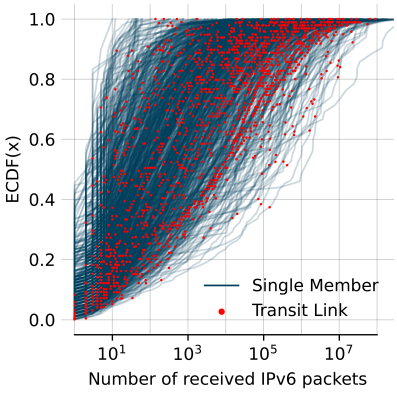


Figure A.6: ECDF for the number of received IPv6 Packets per member

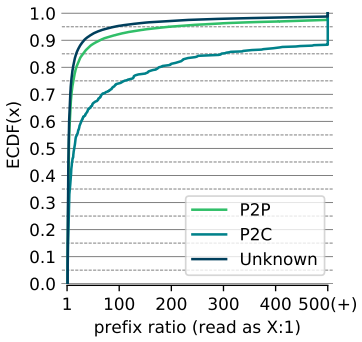


Figure A.7: ECDF for the ratio of /24 receiving traffic per member pair

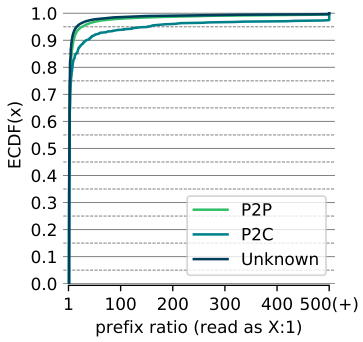
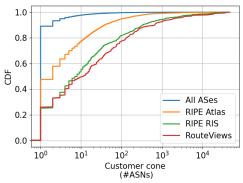


Figure A.8: ECDF for the ratio of /48 receiving traffic per member pair

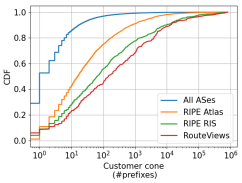
A.2 Distributions of the IMP Infrastructure Characteristics

Figures A.10 and A.9 present the detailed distributions of the characteristics we consider (see §3.2.2.2) for the entire population of ASes and for the ASes in the IMPs.

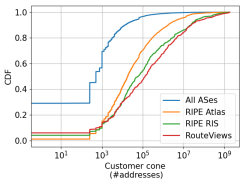
Network size dimensions



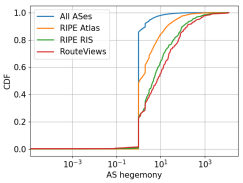
(a) Customer cone (#ASNs)



(b) Customer cone (#prefixes)

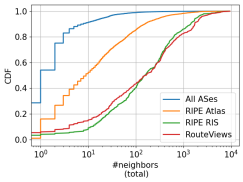


(c) Customer cone (#addresses)

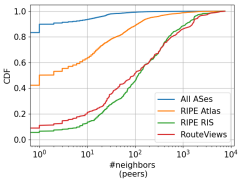


(d) AS hegemony

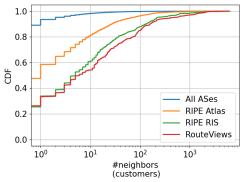
Topology dimensions



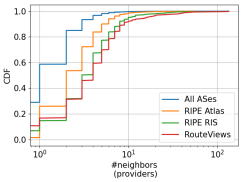
(e) #neighbors (total)



(f) #neighbors (peers)

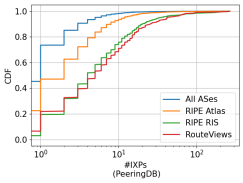


(g) #neighbors (customers)

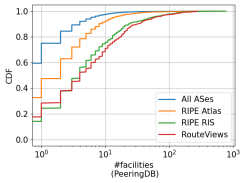


(h) #neighbors (providers)

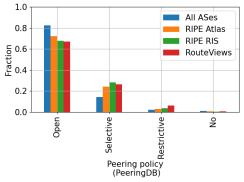
Interconnection (IXP-related) dimensions



(i) #IXPs

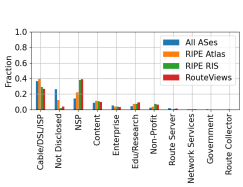


(j) #facilities

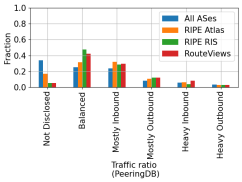


(k) Peering policy

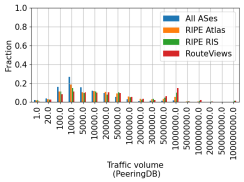
Network type dimensions



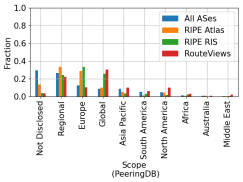
(l) Network type



(m) Traffic ratio

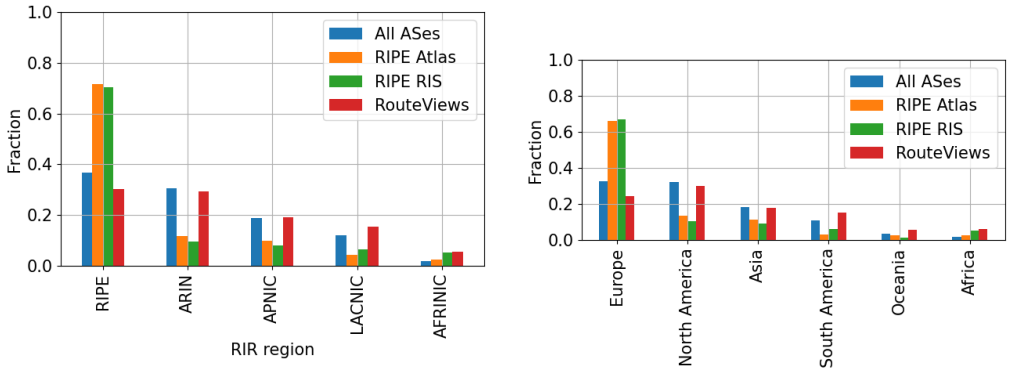


(n) Traffic volume



(o) Scope

Figure A.9: Network size, Topology, Interconnection (IXP-related) and Network type dimensions.



(a) RIR region (b) Location (continent)

Figure A.10: Location dimensions

A.3 Business Relationship Inferences

A.3.1 Does Performance Correlate with Validation Coverage?

Some of the link classes for which the inference algorithms perform poorly have a higher validation coverage. In this section, we show that there is no correlation between these two metrics. We set up the following experiment: We uniformly sample a subset of relationships and track their evaluation performance using the metrics discussed in §3.3.5. We vary the subset size between 50 % and 99 % of the original set size by increments of 1 %. To get a more stochastically robust result, we repeat this process 100 times for each sample size. While we have done this analysis for all link classes mentioned in §3.3.4, we now discuss the results for the $T1 - TR$ class as it produced low-performance results while containing more than 600 peering relationships.

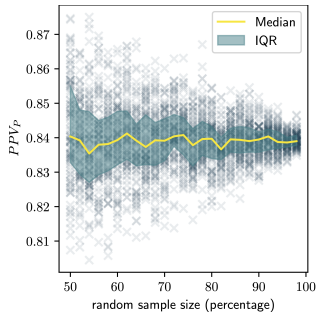


Figure A.11: Precision (P2P)

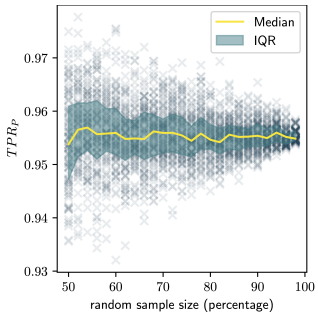


Figure A.12: Recall (P2P)

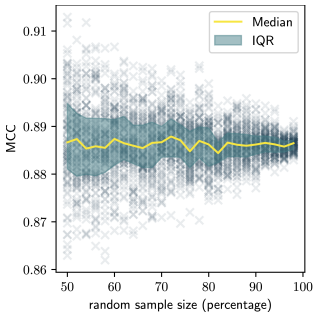


Figure A.13: MCC

Figure A.11, A.12, and A.13 show the sample size on the x-axis against the precision (PPV_P), recall (TPR_P), and MCC on the y-axis. While the figures mark the individual performance measurements for each sampled set with a cross, they also show the interquartile-range (IQR) and median across all 100 sampled sets. Even though we observe that the variance increases with decreasing sample size, we neither observe an increasing nor a decreasing trend for the performance metrics. Notably, the other link classes (not shown) allow for similar conclusions.

A.3.2 Plots for Alternative Metrics

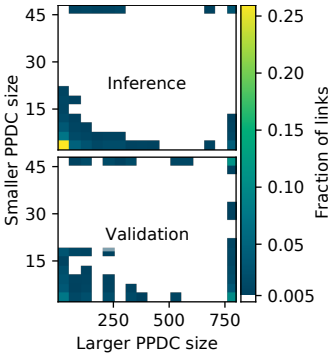


Figure A.14: Customer Cone Imbalance for transit links

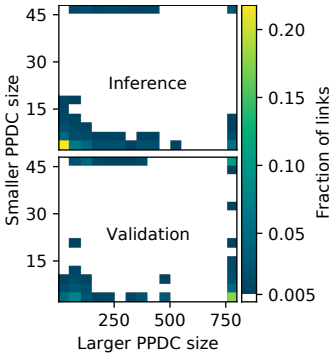


Figure A.15: Customer Cone Imbalance for transit links (ignoring links with incident Route Collector Peers)

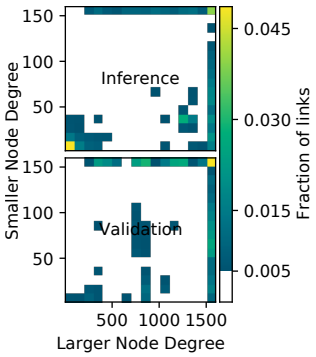


Figure A.16: Node degree Imbalance for transit links

Figures A.14, A.15, and A.16 show alternate variants of figure 3.31 for the customer cone size (CCS), CCS when ignoring links incident to route collector peers (i.e., vantage point ASes), and the node degree.

A.3.3 List of Potential Bias Dimensions

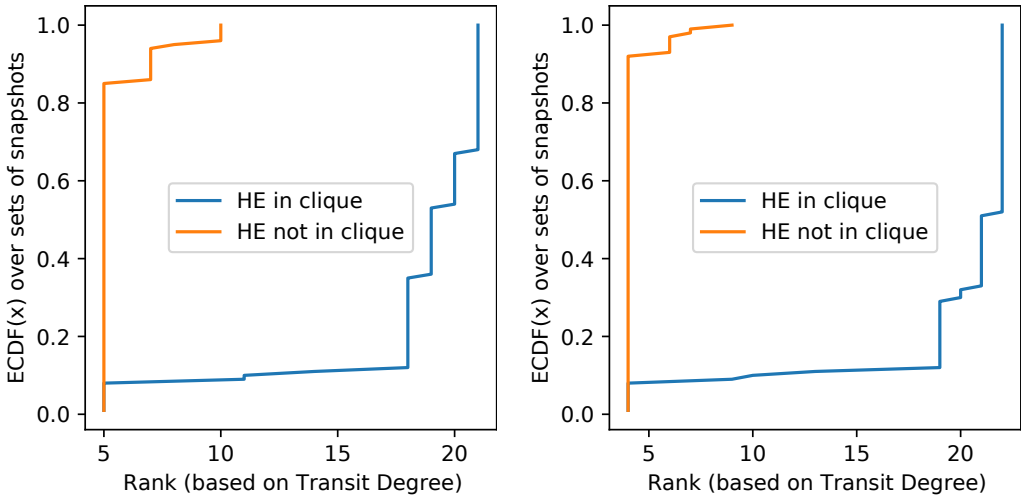
The following per-link metrics might help to identify additional groups of 'hard links':

1. visibility over time
2. number of prefixes redistributed via link
3. number of addresses covered by those prefixes
4. number of prefixes *originated* through the link
5. number of addresses covered by those prefixes
6. number of ASes that can observe (i.e., occur left from) the link
7. number of ASes that might receive traffic via (i.e., occur right from) the link
8. the relative difference in transit degree between the incident ASes
9. the relative difference in PPDC size between the incident ASes
10. and the number of common IXPs where both incident ASes are present
11. and the number of common peering facilities where both incident ASes are present
12. how the incident ASes behave, e.g., BGP serial hijackers [487] vs MANRS participants [305]

A.3.4 Ethical Considerations

This work relies only publicly available routing information and privately obtained validation data. Despite being privately obtained, the validation data is comprised of only public information—only the parsing and processing scripts are not publicly available–from various operator data bases. The individuals that provided routing information to route collector projects as well as the individuals that entered their information into operator databases knowingly offered their information to the public. Both data sources have been used extensively for academic research throughout the last two decades, which shows that their collection and analysis results in substantial benefits for the research community. Based on these observations and the fact that this work does not conduct any form of research on human subjects, we argue that this work raises no ethical concerns.

A.3.5 Ranks & Cliques



(a) AS1299 (b) AS2914

Figure A.17: Ranking of potential clique ASes across snapshots

To illustrate the clique dynamics between AS1299, AS2914, and AS6939, we explicitly calculated the transit degree of all ASNs (again using the `asrank.pl` script) for 200 $30D$ snapshots. The first 100 are randomly chosen from the set of snapshots for which AS6939 is part of the clique while the other 100 are chosen from snapshots for which AS6939 is *not* part of the clique. Figure A.17 shows the rank (based on largest transit degree) that AS1299 and AS2914 have w/o AS6939 (HE) as clique member. The plot indicates that most of the time that HE is part of the clique neither AS1299 nor AS2914 are part of the top-10 ASes ranked by transit degree—an indication we verified by manually inspecting various snapshots.

A.3.6 Untangling Clique Choices

To illustrate why each clique is chosen, we consider the outcomes of two exemplary snapshots for $ws = 30D$[88]. *2018-01-31, 10 a.m. snapshot*: The two cliques that compete for becoming the C_1 seed clique are {6939, 3356, 7018} and {174, 2914, 3356, 7018} with summed transit degrees of $7428 + 4671 + 2123 = 14222$ and $5240 + 1890 + 4671 + 2123 = 13924$, respectively. *2018-02-01, 08 a.m. snapshot*: The competing cliques are {6939, 3356, 7018} and {174, 1299, 2914, 3356, 7018} with summed transit degrees of $7408 + 4679 + 2122 = 14209$ and $5488 + 2082 + 2224 + 4679 + 2122 = 16595$, respectively. Across all snapshots, AS6939 has a transit degree difference of ~2000 ASes to the second largest AS while some of the top-10 ASes not even have a transit degree of 2000. Hence, if a clique does not include AS6939, it must include at least one extra top-10 AS to have a comparable transit degree sum to that of cliques with AS6939. For input formats with shorter window size (e.g., $8H$), the slight changes in the transit degrees of the top-10 ASes across snapshots lead to AS6939 repeatedly entering and leaving the seed C_1 clique (and, after extension, the final clique). The second snapshot illustrates yet another problem: The 6th to 20th ranked AS often have comparable (~500 ASes difference) transit degrees. Hence, the lower half of the top-10 ASNs is very volatile across time. Figure A.17(a) and Figure A.17(b) (both in appendix A.3.5) show that AS6939 (Hurricane Electric (HE)) never gets into the clique when AS1299 and AS2914 make it both into the top-10 (which represents the second scenario above). Hence, the magic number of $N = 10$ ASes among which the seed clique is formed ends up substantially impacting the clique inference outcomes.

[88] Please note that while we would have expected AS3549 to be part of all four competing cliques—as it always was in the top-10 and had links to all other top-10 members—the `asrank.pl` code included it in none. So far, we were not able to identify the exact code responsible for this behavior.

Appendix B
Managing IPv4 Exhaustion

B.1 Delegation Consistency

Delegations and RPKI. In order to observe delegations in BGP data the delegated address space needs to be announced. Since the deployment of route origin validation has increased significantly [108, 409, 488], the Resource Public Key Infrastructure (RPKI) database has become a valuable source to infer delegations. Rather than taking the announcements of P and P', we now check whether those prefixes have Route Origin Authorizations (ROAs) assigned to different ASes. When inferring delegations based on the preprocessed RPKI snapshots by [108], we observe an order of magnitude less delegations compared to BGP. Yet RPKI-based delegations provide a different view on delegation consistency: If S has a ROA assigned for P and delegates P' to T, than T *continuously* needs to have a ROA for P throughout the entire delegation period, otherwise many ASes would filter and not propagate the delegated prefix P'.

We utilize this characteristic of RPKI-based delegations to evaluate the correctness of different consistency rules and then pick one rule to compensate for the on-off-patterns observed for BGP delegations. In general, we analyze rules of the form: "If we observe a delegation on day X and on day $X + M$, the delegation also exists for all but N days in between." In Figure B.1, we present the fail rate (i.e. the fraction of possibilities for which a rules' premise is valid but its conclusion is violated) for rules with different values for N on the y-axis against an increasing values of M (i.e., increasing time frame) on the x-axis. First, we observe that ~90 % of delegations that are seen at least 90 days apart are visible for the entire time frame except for at most 3 days. We also observe that the fail rate never reaches 30 % even when picking extremely large time frames of 100 days. Finally, we observe that even when N is 0 (i.e., the delegation must be visible for *all* days within the time frame) the rule is only violated for ~5 % of all possibilities when choosing a time frame of ten days; therefore, we decided to apply the following consistency rule to all BGP delegations: When we observe the same delegation ten days apart and we do not observe a conflicting delegation (i.e. we observe P being delegated to another delegatee AS T') in the meantime, the delegation also exists for all days in between.

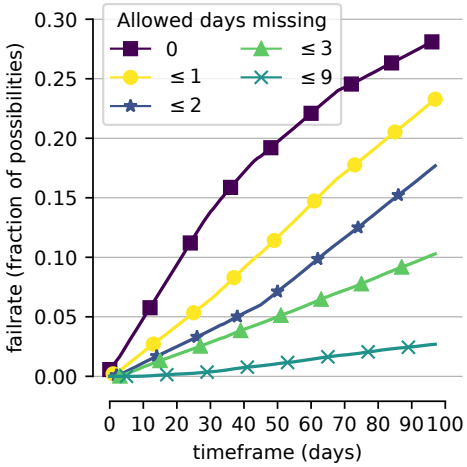


Figure B.1: Validation of different consistency rule values on RPKI delegations.

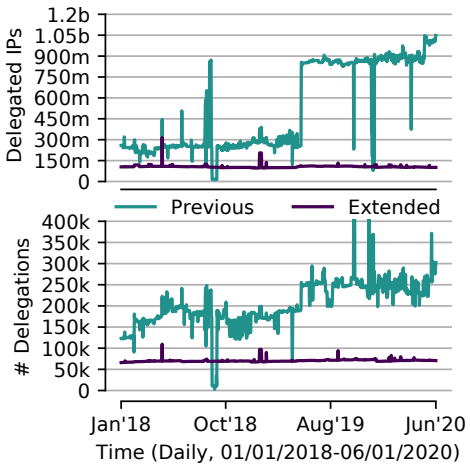


Figure B.2: Number of BGP delegations and delegated addresses w/wo our extensions.

Delegations. Figure B.2 shows the number of delegations as well as the amount of delegated address space inferred by the previously proposed algorithm [268] as well as our extended algorithm over time. While we observe that our extensions significantly reduced the number of delegations inferred for each day, we also see that it almost completely eliminated the high variance that can be observed when using the previously proposed inference algorithm. While the previous approach suggests a significant increase in both, the number of delegations as well as the number of delegated addresses, our extended algorithm only yields an increase of delegations by ~7 % with a negligible change in delegated IPs. When looking further into this result we found that delegation sizes decreased: While the fraction of /20 delegations decreased from ~7% to ~3%, the fraction of /24 delegations increased from ~66 % to ~72 %.

B.2 Orthogonal Analyses for HSPs

B.2.1 Definition: MSPs vs HSPs

In this section, we want to briefly contrast the definitions of More-Specific Prefixes (MSPs) and Hyper-Specific Prefixes (HSPs). P is an MSP of P' when the address space that P describes is entirely contained in the address space that P' describes, e.g., 1.0.0.0/24 is an MSP of 1.0.0.0/22. In contrast, we call a prefix hyper-specific if its CIDR size is larger than /24 or /48 for IPv4 and IPv6, respectively. While labelling a prefix as an MSP requires another (covering) prefix, the HSP label relies entirely on the CIDR size of a given prefix and does not require a second, related prefix. Notably, many—but not all—hyper-specific prefixes are also MSPs of less-specific prefixes. As the definitions of MSPs and HSPs are very different, further classifications of HSPs (as in, e.g., Geoff Huston's blogpost [227]) are not directly applicable to HSPs.

B.2.2 Route Collector Consistency

In order to analyze representative route collector snapshots of the three RC projects Isolario [244], RIPE RIS [349], and Routeviews [364], we first analyze their consistency over time. To estimate the consistency, we initially retrieve data for all days in 2010, 2013, 2016, and 2020. For each day, we download the first routing information base (RIB) snapshot as well as all available update messages produced by each RC. If an update file is missing, we, additionally, download the first available RIB snapshot after the missing update file. After extracting the HSPs for each day, we analyze consistency as the fraction of HSPs seen at day $n+w+1$ that are also visible within the observation period $[n, n+w]$. Notably, we try all possible window size positions, i.e., $n \in \{0, ..., d-w-1\}$ where d is the number of days in the given year.

Figure B.3 shows the mean as well as the interquartile range (IQR) across all possible n for window size w between 1 and 60 days for IPv4 and IPv6 HSPs in 2020. We observe that a seven-day window allows us to achieve a consistency of 97 % and 98 % for IPv4 and IPv6, respectively. Notably, further expanding the window size to 60 days would only increase the consistency by ~0.5 %. Given that we now have a snapshot aggregation window, we still need to pick a snapshot interval. When comparing the number of visible HSPs for different snapshot intervals, we observe that a three-month interval provides an optimal balance: While the number of data points is still capable of capturing all visible trends in more-frequent snapshot intervals, the reduced amount of data (i.e., only seven days every three months) still allows us to perform computationally expensive observations for the entire decade promptly.

B.2.3 In-depth Visibility Analysis

How prominent are HSPs? To understand the prevalence of hyper-specific prefixes, we aggregate the routing tables of all collector peers and compare the distribution of prefixes depending on CIDR sizes. Figure B.4 shows those distributions as stacked bar plots for each snapshot. We observe that up to 13 % (in 2015) and 25 % (in 2018) of totally visible prefixes are hyper-specific for IPv4 and IPv6, respectively. Yet, the usual contribution of HSPs is approximately 10 % for most months. Note that this does not mean that any single routing table contains that many HSPs on its own.

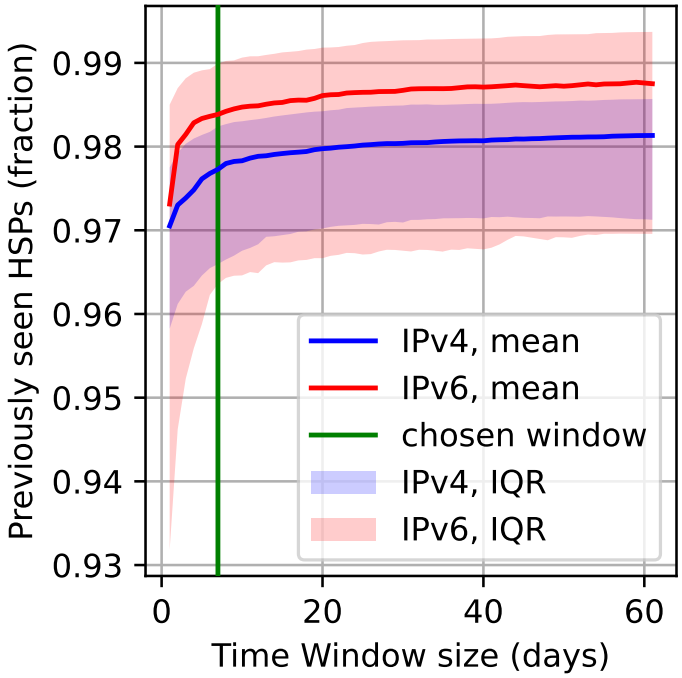


Figure B.3: Impact of window size on visible HSPs.

How visible are HSP? To further elaborate on this point, Figure B.5 shows the number of hyper-specific prefixes per IPv4 (left) and IPv6 (right) snapshot separated based on the number of route collector peers that can see them. For IPv6, we observe that most hyper-specific prefixes can be seen by two or more peers, with around a fifth of all HSPs being visible by 11+ peers for most snapshots. Similar to the previous plot, we again observe a peak of (~20K) hyper-specific prefixes at around 2018. While we are not able to account this peak to a single factor, we observe that the increase is rather uniform across collector peers, origin ASes, intermediate ASes, and address space and, hence, is unlikely to stem from a measurement artifact or some local misconfiguration. When comparing the situation before and after the peak, we still can see an increase from ~7K HSPs in 2016 to ~11K HSPs in 2021. In contrast to IPv6, many HSPs in IPv4 can only be seen by one peer. While we observe few HSPs that can be seen by 100+ peers, the vast majority of HSPs can only be seen by 10 or less peers. Even though the number of low-visibility HSPs strongly fluctuates between snapshots, it increases rather continuously across many snapshots. Both such characteristics are significantly less pronounced for IPv4 HSPs that can be seen by 6+ peers. This difference may be accountable to various reasons including the association of a prefix to a certain function or a prefix's lifetime.

HSP aggregation. ASes often have economical incentives to keep their BGP routing table size low. To realize this goal, some ASes aggregate (multiple) more-specific routes into a single less-or-equally-specific route [120]. If an anchor-prefix results from aggregating prefixes with different CIDR sizes (prefix-based aggregation), we know that one of such pre-aggregation prefixes must have been hyper-specific. Yet, confidently identifying such aggregations is challenging. According to RFC 4271 [405], a router *MAY* set the `AGGREGATOR` field when it performs prefix-aggregation—which can serve as indication that *some* form of aggregation must have happened. Thus, we first extract all routes for anchor-prefixes which have the `AGGREGATOR` field set. At this stage, our selected

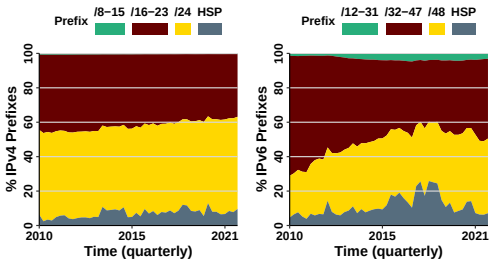


Figure B.4: HSP prefix contribution over time

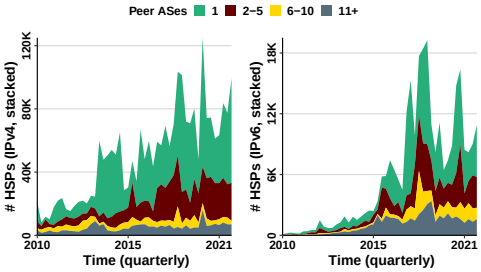


Figure B.5: HSPs by # Peer ASes over time

routes might be a result prefix-based aggregation or the aggregation of different routes—e.g., with different AS_PATH attributes (path-based aggregation)—for the same prefix (or both). To reduce the likelihood of falsely identifying HSP usage due to path-based aggregation, we rely on the ATOMIC_AGGREGATE field as well as the presence of AS_SET elements in the AS_PATH attribute. A router *SHOULD* set the ATOMIC_AGGREGATE field if the newly generated AS_PATH attribute of the post-aggregation route does not contain all AS numbers present in the pre-aggregation routes, e.g., the paths AB and AC can be aggregated to AB (which hides the existence of C). If the ATOMIC_AGGREGATE field is not set, ASes often use AS_SETs to signal path-aggregation, e.g., the paths AB and AC can be aggregated to $A\{B, C\}$ (where {...} denotes the AS_SET containing all ASes after A). As the ATOMIC_AGGREGATE field and AS_SETs indicate path-based aggregation, we remove all anchor-routes that contain at least one of them.

Where does HSP aggregation happen? Now that we have a set of anchor-prefixes that are likely the result of prefix-based aggregations, we can analyze how close to the origin HSPs are aggregated. We compare the AS number in the AGGREGATOR field with the AS_PATH and differentiate between the following cases: (1) **Origin**—the origin itself performed the aggregation, (2) **On-path**—an AS within the AS path that is not the origin performed the aggregation, and (3) **Off-path**—some AS that does not occur in the AS path performed the aggregation[89]. Figure B.6 shows the number of anchor prefixes in each class over time. Notably, the figure also contains the class **Multiple** that contains anchor prefixes for which there are multiple paths with inconsistent classes. We observe that the the vast majority of anchors are actually aggregated at the origin with only few hundreds of anchors being aggregated on-path. Origin and off-path (especially AGGREGATOR fields with private ASNs) aggregation often occurs due to the use of BGP confederations [116, 254] where the AS is internally split into multiple private sub-ASes. Depending on how an AS border router handles the aggregation of internal confederation routes, it might either correctly set the external AS number or leak the internal confederation AS Number in the AS_PATH or AGGREGATOR attribute. Notably, those HSP routes are likely not available to other ASes (including neighbors of the origin).

Projected actual usage. While our IRR snapshots produced actual HSPs, our final prefix-aggregation and ROAs only produced a list of anchor-prefixes that is likely to contain HSPs. Therefore, we decided to analyze the potential extent of HSP usage on the basis of anchor-prefixes. Figure B.7 shows the number of IPv4 (left) and IPv6 (right) anchor-prefixes per data set (stacked) over time. Notably, the aggregated class only contains on-path aggregated anchor prefixes and the RPKI class only contains anchor prefixes for explicit HSP ROAs. The "multiple" class covers those entries that are visible via

[89]This class also includes reserved AS numbers.

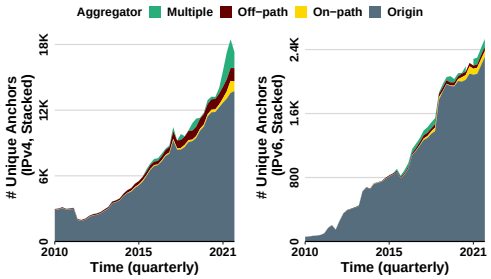


Figure B.6: Position of HSP Aggregation

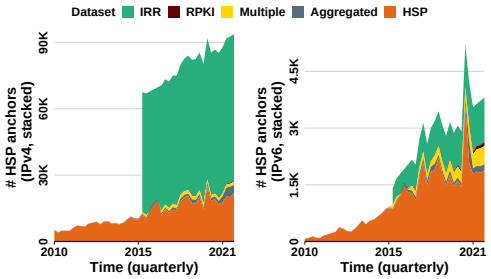


Figure B.7: HSP anchors across data sets

multiple data sources. We observe that the current route collector infrastructure misses roughly one-third of the of the anchor prefixes that potentially contain HSPs. We further observe a less noisy, linear increase in the number of anchor prefix for which HSPs are visible compared to the raw count of visible HSPs. Notably, some part of this increase can potentially be accounted to the increasing numbers of route collectors and route collector peers over time.

Who uses hyper-specific prefixes? We leverage the "AS Classification Inferences" dataset described in ASDB [528] to classify ASes as Content, Education, Hypergiant, ISP (Stub), ISP (Transit), Tier 1, and Others. Figure B.8 compares the classes of all BGP-visible ASes (left) to HSP origin ASes (right) over time. We find that in contrast to all origin ASes, HSP origins are more likely to be ISP (Transit) ASes. Interestingly, the majority of Tier 1 ASes is also originating HSPs. During the period of January 2019 until October 2021, we identify between 12 and 15 of the total 19 Tier 1's as HSP origins. In contrast to the high share of Tier 1 HSP origins, we find that most hypergiants do not originate HSPs.

B.2.4 Real-World Propagation Experiment

Does BGP reflect control plane reachability? Finally, we want to understand how much the lack of additional BGP vantage points impacts our observations on reachability. Hence, we configure a real-world experiment using the PEERING testbed [445] in which we announce an anchor prefix as well as multiple hyper-specific prefixes. Once those prefixes have converged, we run traceroutes from RIPE Atlas [433] probes and compare their resulting paths to those visible at route collectors.

Vantage points & resources. The PEERING testbed allocates Internet resources (specifically, IPv4/IPv6 address space and AS numbers) to its users based on approved experiment proposals. Once allocated, users can announce those resources via the testbed's infrastructure. Given that the PEERING testbed strongly relies on third party resources (e.g., for hosting infrastructure), announcements must be designed carefully to not cause trouble or irritation for other network operators. For our experiment we use the address ranges 184.164.240.0/23 and 2804:269c:4::/46. More specifically, we utilize 184.164.240.0/24 and 2804:269c:4::/48 as anchor prefixes (i.e., they represent our control group) and announce HSPs only from the remaining address space[90].

RIPE Atlas [433] is a measurement platform with probing devices (henceforth called probes) all over the world. To maximize probing coverage and minimize probing load, we choose at most one probe per AS. To reduce the likelihood of probe outage, we

[90] In particular, we announce 184.164.241.0/25, 184.164.241.128/28, 184.164.241.255/32, 2804:269c:5::/49, 2804:269c:6::/64, 2804:269c:6:8000::/65, and 2804:269c:7::/128.

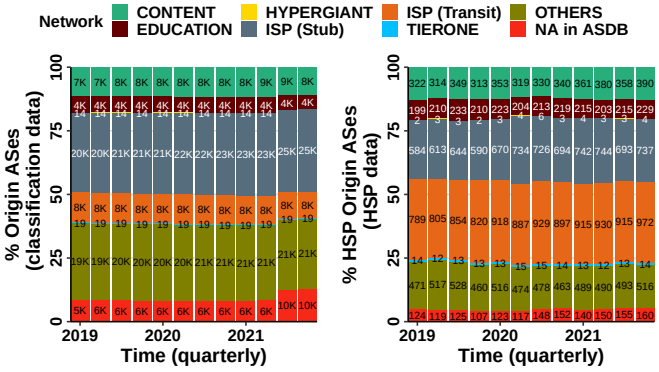


Figure B.8: HSP origin AS classification over time.

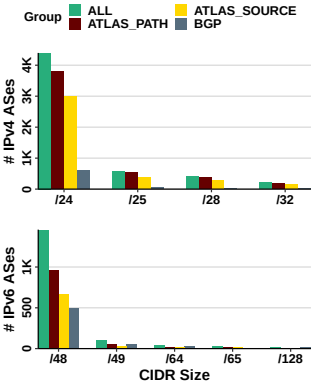


Figure B.9: PEERING testbed propagation results

select only probes that are not tagged with system-problematic tags[91]. We further validate that selected IPv4 and IPv6 probes are tagged with `system-ipv4-works`, and `system-ipv6-works`, respectively. If an AS hosts multiple probes, we prefer dual-stack probes (such that we can use a consistent probe for our IPv4 and IPv6 measurements) over anchor probes (i.e., better equipped probes) over any other probes. If we still have multiple choices, we pick the probe that is tagged with the highest stability tag (e.g., `system-ipv4-stable-90d`). Our final probe set consists of 3097 probes distributed across 2990 IPv4 and 1239 IPv6 ASes.

Experimentation environment. The PEERING testbed currently has a total of 180 IPv4 and 152 IPv6 neighboring ASes. Yet, most neighbors do not support/redistribute HSPs. We identify supportive neighbors by iteratively announcing a /25 or /49 prefix from our allocated address space through each neighbor and analyzing the resulting update stream from RIPE RIS and Routeviews. Since, at this point, we only care about a "life sign" (i.e., whether or not *any* update was received) rather than full convergence, we adopt a short announcement cycle: We announce a prefix at the beginning of every full hour and withdraw it 30 minutes later[92]. We identify a set of 8 IPv4 and 9 IPv6 neighboring ASes that redistribute HSPs. Notably, those ASes are distributed across 4 and 3 geographically separate Points of Presence.

Technical realization. Throughout May 21 and 22, 2021, we announce /24, /25, /28, and /32 IPv4 prefixes and /48, /49, /64, /65, /128 prefixes through a single neighbor at the beginning of every even hour. After the announcement, we wait 40 minutes to allow the prefix to converge[93]. After those 40 minutes, we run active measurements for 10 minutes, and then withdraw the prefixes again. Notably, we choose 70 minutes between a withdrawal and the next announcement on purpose such that we out-wait the expiration of potential Route Flap Damping hold-down timers, which have been shown to usually expire after 60 or less minutes [192].

[91] tags: `system-flash-drive-filesystem-corrupted`, `system-v1`, `system-no-flash-drive`, `system-flash-drive-bad-or-too-small`, `system-firewall-problem-suspected`, `system-trying-to-connect` `system-readonly-flash-drive`, `system-no-controller-connection`, `system-bad-firmware-signature`, `system-flakey-connection`, `system-flakey-power`, `system-flash-drive-problem-detected`, and `system-v2`

[92] These experiments ran between the May 1, 2021 and the May 3, 2021.

[93] During previous experiments we observed that usually the 95th percentile of updates reach the collector peers already in the first 15 minutes.

During our 10 minute active measurement period, we run paris-traceroutes from all probes towards either the network address or the first non-network address of all prefixes (which are configured to be pingable). To reduce the dependence of our results on the underlying protocol, we simultaneously issue ICMP, TCP, and UDP probing. To keep the induced load for the RIPE Atlas platform as well as for the peering testbed manageable, we reduce the number of probing packets used per per hop by paris-traceroute from RIPE ATLAS' default of 3 packets to one packet. Notably, as the resulting load still exceeds the default limitations (e.g., for measurement results per day) for a single RIPE Atlas account, we coordinate our probing efforts with the RIPE Atlas team who generously raised the limits for our experiments.

We map traceroutes to AS Paths using the state-of-the-art mapping tool bdrmapit [313]. As bdrmapit requires a large corpus of traceroutes as input to perform well, we use traceroute data from CAIDA's IPv4 Prefix-Probing data set [493], CAIDA's IPv4 Routed /24 Topology Dataset [493], CAIDA's IPv4 Routed /48 Topology Dataset [494], and RIPE's hourly archives of Atlas traceroutes [426] between May 17, 00:00 and May 24, 00:00. For all the other inputs (e.g., prefix-to-origin mappings or business relationship inferences) we use recent snapshots from the recommended data sources. Finally, we use bdrmapit's output to map our successful (i.e., only those that actually reached the respective target host) traceroutes to AS paths.

Comparison. Figure B.9 compares the the number of ASes (aggregated over all iterations) that (1) hosted Atlas probes that reached the target (ATLAS_SOURCE, yellow), (2) appeared along the path between ATLAS_SOURCE ASes and the Peering Testbed (ATLAS_PATH, dark red), (3) are visible from route collector peers (BGP, gray). The most drastic observation is that hyper-specific prefixes see a very sharp drop in reachability. Even the best performing CIDR size, /25, only reached ~15 % of of the ASes that are reached by its respective anchor prefix. Especially for IPv6 we observe that most PEERING neighbors redistribute our prefixes (including the anchor prefix) only towards their customers, hence, some of our Atlas probes are unable to reach the peering testbed even for the anchor prefix. We further find that the more-specific the prefix gets, the less likely it propagates. This finding is interesting as most recommended filtering guides [153, 156, 306, 356, 359] treat all hyper-specific CIDR sizes equally. Our third observation is that the reachability reflected by route collector peers substantially underestimates data plane reachability. While we are able to observe approximately a third of the total ASes for our /48 prefix via BGP, this fraction lies at around 14 % for our /24 prefix.

B.2.5 Filtering Pipeline

When an AS peers with a Route Collector, the router that feeds the collector may provide all routes that are not removed during (or before) egress filtering. Hence, misconfigured egress filters can lead to misinterpretations. For our analysis, we filter out HSPs which are originated by feeder AS directly connected to a route collector. However, we use the HSP if it has been propagated to at least 2 AS hops, including feeder AS. In addition, we filter all private, reserved, multicast, and experimental IP prefixes. Furthermore, we also filter prefixes originated by a private AS. Finally, we remove the HSPs we identify as outliers during the data cleaning process. Appendix B.2.6 provides detail information on HSPs we have filtered out.

B.2.6 Applied Data Isolation Rules

We applied the data isolation rules summarized in table B.1. ASes marked with ⋆ contributed/announced either (1) an extraordinarily high number of HSPs (i.e., 100 or more times higher than in other snapshots) or (2) HSPs in an extraordinarily high number of anchor prefixes for a limited amount of time.

Timeframe	Filter name	Filter Details	Reason
entire period	Private Origin ASes 2 Bytes	Origin AS number from 64512 to 65534	private IPv4 ranges.
entire period	Private Origin ASes 4 Bytes	Origin AS number from 4200000000 to 4294967294	private IPv4 ranges.
entire period	Private IPs	IPv4 Private IP ranges	private IPv4 ranges.
entire period	Class D and E	IPv4 Prefixes > 223.x.x.x	IPv4 multicast & class E.
entire period	Abnormal Prefixes	for IPv4 prefix > /32 for IPv6 prefix > /128	abnormal IPv4 prefixes. abnormal IPv6 prefixes
entire period	No Origin Internal	Routes having no origin AS Feeder AS is the Origin AS	AS-internal routes.
2015/10/01-07	IPv4 Noisy Origins	Origin AS == 9498	origin AS.⋆
2016/10/01-07	IPv4 Noisy Origins	Origin AS == 36937	origin AS.⋆
2017/04/01-07	IPv4 Noisy Origins	Origin AS == 9498	origin AS.⋆
2019/07/01-07	IPv4 Noisy Origins	Origin AS 7122	origin AS.⋆
entire period	IPv4 Noisy Origins	Origin AS 12400	origin AS.⋆
2016/07/01-07	IPv4 Noisy Peer AS	Peer AS 35908	peer AS.⋆
2017/01/01-07	IPv4 Noisy Peer AS	Peer AS 60924 and 27630	peer AS.⋆
2017/10/01-07	IPv4 Noisy Peer AS	Peer AS 37497	peer AS.⋆
2018/10/01-07	IPv4 Noisy Peer AS	Peer AS 14361	peer AS⋆
2019/01/01-07	IPv4 Noisy Peer AS	Peer AS 262757	peer AS.⋆
2020/04/01-07	IPv4 Noisy Peer AS	Peer AS 268430	peer AS.⋆
2021/04-07/01-07	IPv4 Noisy Peer AS	Peer AS 398465	peer AS.⋆
2021/01-10/01-07	IPv4 Noisy Peer AS	Peer AS 203125	peer AS.⋆
2020/04-07/01-07	IPv4 Noisy Peer AS	Peer AS 268430	peer AS.⋆
entire period	IPv6 Noisy Origins	Origin AS 4761	origin AS.⋆
2017/07/01-07	IPv6 Noisy Origins	Origin AS 17451 and 45899	origin AS.⋆
2019/04/01-07	IPv6 Noisy Origins	Origin AS 7713	origin AS.⋆
2021/07/01-07	IPv6 Noisy Origins	Origin AS 8100	origin AS.⋆
2018/07/01-07	IPv6 Noisy Peer AS	Peer AS 199036	peer AS.⋆

Table B.1: Applied filtering and isolation rules.

Appendix C
Securing Routing Operations

C.1 Path Preprending

C.1.1 Timeline for the Effectiveness Experiments

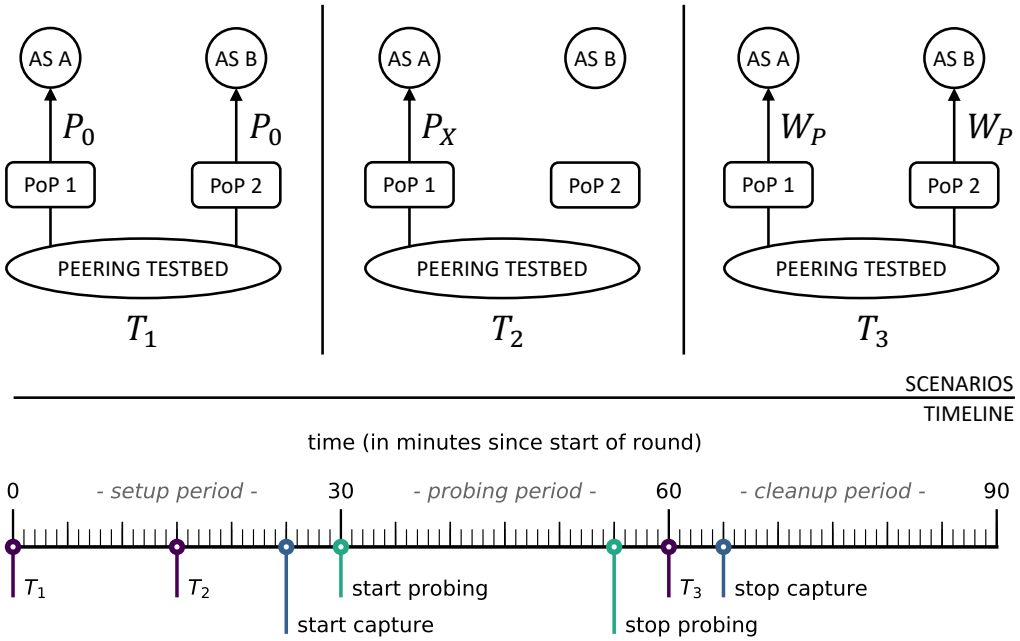


Figure C.1: Effectiveness experiment: Timeline and experimental setting (P_X: announcement of prefix P with $X \in 0, \ldots, 3$ prepends; W_P: withdraw P.

In Figure C.1, we depict the timeline of events and the configuration scenarios from each iteration of our effectiveness experiment. First, see Scenario T_1, we create a baseline by announcing our prefix P via all upstreams without any prepending. After waiting 15 minutes to allow BGP to converge, we announce P with X prepends via the chosen upstreams, see Scenario T_2. After again waiting for 15 minutes to allow BGP to converge, we conclude the *setup period* and start a 25-minute *probing period*. Each probe consists of ICMP, TCP, and UDP pings triggered once per minute to all targets. To reduce probing bursts, we spread the packets evenly across the one-minute time interval. Before the cleanup, we have a 5-minute break to ensure that the last responses can arrive before we withdraw the prefix. After the break, we start the *cleanup period* with the withdrawal of the announcements in every upstream, see Scenario T_3. To allow for BGP to converge and to minimize the risk of BGP Route Flap Damping, we wait for 30 minutes before starting a new iteration.

C.1.2 Monitor Filtering

Figure C.2 shows the distribution of the fraction of monitors that observe a given prefix based on all routing information available on January 15th of each year. While prefix visibility has increased over the decade, we find a clear separation between two distinct regions in the plot regardless of the year. On the leftmost side, we have locally visible prefixes (seen by less than 20 % of monitors), and on the rightmost side, globally visible prefixes (seen by over 80 % of monitors). Based on this finding, we decided to remove all routes to prefixes observed by less than one-third of all monitors, as indicated by the threshold line. Notably, picking any other threshold between 0.2 and 0.8 only results in negligible differences.

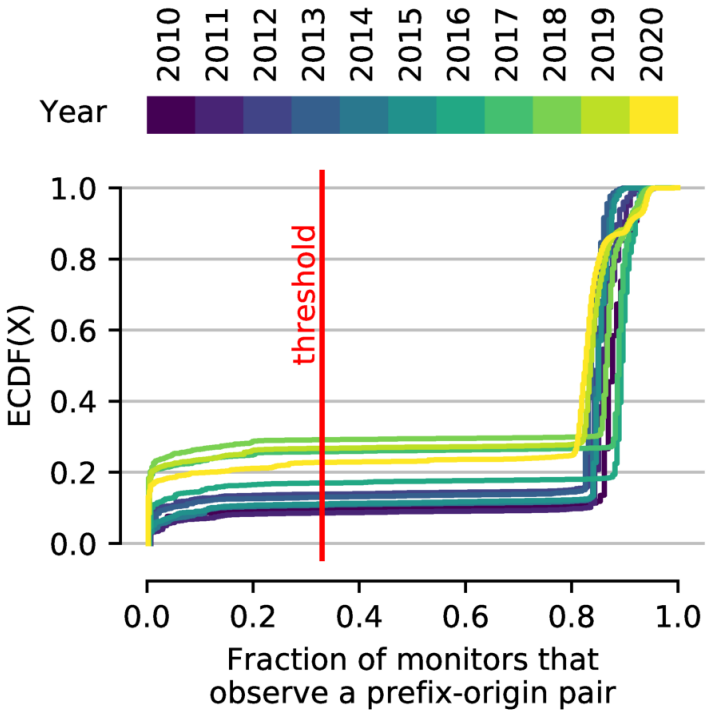


Figure C.2: Fraction of monitors that observe each prefix-origin pair.

C.2 Private Disclosure Notification

Figure C.4 shows the initial email that we sent out in the private disclosure notification. In Figure C.3 we show the follow-up email highlighting why the attack can cause serious harm and has already been run on a larger-scale by an unknown third party.

```
Dear colleagues,

we received some feedback that the message we provided you with is simply stating the
obvious, and noticed an important piece of information missing:

Note, that we conducted experiments with a limited (<=500 prefixes) test-setup around the
29th of September. On the 5th of October an entity unknown to us replicated our experiments
 via AS20473 with around 8k prefixes, already causing noticeable load but yet staying below
 the potential of this technique. We hence assume that our technique is by now known--not
only commonly known in the community but potential attackers being consciously aware--to
third parties, which is why we are sending out these notifications for something
technically well known. We plan to notify the wider networking community in one week from
now.

With best regards,
Lars Prehn
```

Figure C.3: Follow-up email text of private disclosure notification.

Dear <Person>,

I'm a researcher at the Max Planck Institute for Informatics in Germany and received your contact from <Person>, who believes that you might be the right contact at < Company> for the following issue:

We started the private disclosure process for an IPv6–based routing attack discovered in a research collaboration between the Max Planck Institute for Informatics in Germany and the Institute of Theoretical and Applied Informatics, Polish Academy of Sciences. We'd highly appreciate your valuable insights and hope you join our efforts in globally deploying effective prevention mechanisms. To keep the Internet and its users safe, it is important to keep the attack details confidential until prevention mechanisms are in place; we count on you not to publicly share this information prior to the public disclosure, which we currently plan for Wednesday, 19th October 2022.

What is the problem?
Routers either crash, drop sessions, or behave in other unintended ways when their FIB or RIB runs out of memory. While newer routers can store up to 4M prefixes, many ASes still run (at least some) older hardware that may only be able to store 1M routes or even less. TL;DR: We found an attack that allows an adversary to introduce very quickly more than 1M new and unique IPv6 prefixes into the global routing table and is only preventable with the help of major transit networks and IXPs. If, afterwards, these prefixes also get withdrawn simultaneously, the resulting path–hunting behavior additionally results in a massive flooding attack.

How does the adversary even obtain 1M unique prefixes?
After obtaining a /29 address block from any of the RIRs (e.g., RIPE this does not even require need–based justification) the adversary announces every possible /48, /47,... /29 route leading to the announcement of 1.048.575 unique routes–––if C is the difference between the minimal propagating CIDR size, /48, and the CIDR size of the address block from which an attacker sources routes, the adversary can announce up to 2^(C+1) – 1 unique routes, e.g., a /46 block can source seven routes in total: one /46 route, two /47 routes, and four /48 routes.

Don't we have per–session prefix limits that prevent such attacks?
If the average per–session limit is X, an adversary 'simply' has to distribute its routes via 1M/X many sessions, i.e., per–session limits do not eliminate the issue, they only transform it into a session–hunting challenge. During our real–world experiments and discussions, we noticed that while many ASes set tight (often 100–500 prefixes) per–session limits on their peering sessions, it's less common that ASes on either side of a transit session enforce prefix limits.

Why does ROV not protect us from this attack?
It is possible to set a single ROA entry that specifies that the /29 prefix can be announced with CIDR sizes up to /48. If the adversary generates such a ROA and waits some days for it to propagate to all validating ASes, each of the more than 1M prefixes would be a valid announcement.

How can an adversary even get hundreds or thousands of sessions?
The idea is that remote peering providers and VPS providers (e.g., Vultr) enable the adversary to quickly and cheaply 'click together' (virtual) ports at many (think 20+) different peering LANs. The adversary obtains transit by picking providers that also establish transit sessions over peering LANs (Hurricane Electric being the prime example), many bi–lateral peering sessions via openly/aggressively peering networks (that can be identified via, e.g., PeeringDB), and additional (less effective) sessions via multi–lateral peering with Route Servers. Surprisingly, while it would be hard to assemble enough sessions with just one port at each peering LAN (yet eventually doable), this limitation does not exist in reality; while certain providers directly allow clicking multiple ports for a single peering LAN, there are also multiple providers–––this allows the adversary to obtain a 5X to 10X factor for its session counts by establishing multiple sessions to each neighbor (in fact each port of each neighbor).

Do these routes even propagate far enough?
TL;DR: yes. As a rule of thumb: The routes announced via transit sessions usually propagate globally, routes announced to bi–lateral peers usually propagate into the peer's customer cone, and routes announced via multi–lateral peering usually propagate only to the peer's regional customers. As part of our research, we analyzed the propagation behavior and found that an adversary that combines announcements via all three peering types can inject lethal amounts of IPv6 routes into routers of 8k+ ASes, i.e., yes, enough of these routes propagate far enough.

Don't ASes along the path aggregate the individual routes?
While some ASes do aggregate routes, it is possible to launch the attack in such a way that routes can not be aggregated: the adversary would have to choose the prefixes in each session in such a way that neither two consecutive prefixes nor a prefix and its covering prefix are announced via the same session and/or neighbor. To be extra safe, the adversary could switch between multiple origin ASNs for the announcements or use path–poisoning to alter a route's AS path.

What can IXPs do to help prevent the attack?
Ensure that your route servers have tight prefix limits and that they only accept a small number of sessions from each participant.

If applicable, monitor your members' session acquisition behavior (e.g., by looking for BGP–session related packets in the peering LAN's traffic data) to identify potential adversaries early.

What can transit providers do to help prevent the attack?
Introduce dynamically growing yet tight per–session limits on all of your sessions. Allow, e.g., customers and peers to announce at most 1.3x the number of prefixes they announced yesterday. Similarly, the IPv6 routing table currently grows at a rate of <50k new prefixes per year; hence, one could limit the maximum daily growth to, e.g., at most 10k prefixes.

Closely monitor the number of sessions that other ASes establish with you–––especially if your peering policy is fully open or you employ a fully automated session establishing service.
Given that the attack model is highly distributed, the best position to install protection mechanisms is your route reflectors, as they often have a complete view of the globally redistributed routes. If possible, implement the following two limiters:

(i) ensure that you only accept and redistribute a certain number of routes per origin AS
(ii) ensure that you only accept and redistribute a certain number of more–specific routes for each assigned address block.
(iii) accept only what is correctly registered. Do not allow an automatic "or longer" for any registered prefix. This will not prevent the attack but add more effort on the attackers' side to register the resources correctly.
(iv) monitor your generated filter size. A simple check on the number of acceptable prefixes can reveal the preparation of such an attack.

If you have any further questions, please don't hesitate to contact me!
Best regards,
Lars Prehn

Figure C.4: Private disclosure notification email text.

List of Abbreviations

AS Autonomous System 1

ASN Autonomous System Number 10

ASPP AS path-prepending 104

BGP Border Gateway Protocol 1

CAIDA Center for Applied Internet Data Analysis 21

CDN Content Delivery Network 1

CGN Carrier-Grade NAT 82

CPE Customer-Premises Equipment 97

DDOS Distributed Denial of Service 4

FIB Forwarding Information Base 11

FQDN Fully Qualified Domain Name 33

HSP Hyper-Specific Prefix 6

IANA Internet Assigned Numbers Authority 10

IP Internet Protocol 9

IPv4 Internet Protocol Version 4 1

IPv6 Internet Protocol Version 6 1

IQR Inter Quartile Range 28

IRR Internet Routing Registry 14

ISP Internet Service Provider 1

ITE Inbound Traffic Engineering 104

IXP Internet Exchange Point 1

LAN Local Area Network 12

LG Looking Glass 30

LIR Local Internet Registry 10

List of Figures

List of Tables

Bibliography

[1] Lida Abdi and Sattar Hashemi. To combat multi-class imbalanced problems by means of over-sampling techniques. *IEEE transactions on Knowledge and Data Engineering*, 28(1):238–251, 2015.

[2] Emile Aben. 768k day. will it happen? did it happen? https://labs.ripe.net/author/emileaben/768k-day-will-it-happen-did-it-happen/, 2019. last accessed: 19th Sept. 2021.

[3] Emile Aben. RIPE RIS route collectors map. https://observablehq.com//ris-route-collectors-and-peer-locations, 2022. Accessed: 07 May 2022.

[4] Emile Aben and Colin Petrie. Propagation of Longer-than-/24 IPv4 Prefixes, 2014.

[5] Emile Aben and Colin Petrie. Has the Routability of Longer-than-/24 Prefixes Changed?, 2015.

[6] AbuseIPDB. making the internet safer, one ip at a time. https://www.abuseipdb.com/, 2022. Last accessed: March 29th, 2022.

[7] Network Academy. Ipv4 header vs ipv6 header. https://www.networkacademy.io/ccna/ipv6/ipv4-vs-ipv6, 2023.

[8] AFRINIC. stats. Available at https://ftp.afrinic.net/pub/stats/afrinic/ Last accessed: May 31st, 2020.

[9] AFRINIC. Afrinic ipv4 exhaustion. https://afrinic.net/exhaustion, 2020. last-accessed: June 3rd, 2020.

[10] AFRINIC. Latest transfers. https://ftp.afrinic.net/pub/stats/afrinic/transfers/transfers_latest.json, 2020. Last accessed: June 25th, 2020.

[11] AFRINIC. Membership fee and payment facilities. https://afrinic.net/membership/cost, 2020. last-accessed: 03/29/2020.

[12] AFRINIC. Extended delegations, 20180405. https://ftp.apnic.net/stats/afrinic/2018/delegated-afrinic-extended-20180405, 2021. Last accessed: 24th April, 2021.

[13] AFRNIC. Afrinic enters ipv4 exhaustion phase 2. https://afrinic.net/20200113-afrinic-enters-ipv4-exhaustion-phase-2, 2020. last-accessed: June 3rd, 2020.

[14] Securebit AG. Internet resources. https://www.securebit.ch/internet/resources, 2022. Last access: October 2022.

[15] Bernhard Ager, Nikolaos Chatzis, Anja Feldmann, et al. Anatomy of a large european ixp. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, pages 163–174, 2012.

[16] Adnan Ahmed, Zubair Shafiq, Harkeerat Bedi, and Amir Khakpour. Peering vs. transit: Performance comparison of peering and transit interconnections. In *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, pages 1–10. IEEE, 2017.

[17] Eatedal A Alabdulkreem, Hamed S Al-Raweshidy, and Maysam F Abbod. Using a fight-or-flight mechanism to reduce bgp convergence time. In *Fourth International Conference on Communications and Networking, ComNet-2014*, pages 1–4. IEEE, 2014.

[18] Alexa. The top 500 sites on the web. `https://www.alexa.com/topsites`, 2021. last accessed: 21st. June. 2021.

[19] Thomas Alfroy, Thomas Holterbach, and Cristel Pelsser. Mvp: Measuring internet routing from the most valuable points. In *Proceedings of the 22nd ACM Internet Measurement Conference*, IMC '22, page 770–771, New York, NY, USA, 2022. Association for Computing Machinery.

[20] Mark Allman, Robert Beverly, and Brian Trammell. Principles for measurability in protocol design. *ACM SIGCOMM Computer Communication Review*, 47(2):2–12, 2017.

[21] AMS-IX. AMS-IX breaks through 8 Tbps barrier, 2020. Available at `https://www.ams-ix.net/ams/news/ams-ix-breaks-through-8-tbps-barrier` Last accessed: April 14th, 2020.

[22] AMS-IX. Total traffic statistics. `https://stats.ams-ix.net/index.html`, 2021. Last accessed: 27th June, 2021. Archived version available at:`https://web.archive.org/web/20210627072325/https://stats.ams-ix.net/index.html`.

[23] Ruwaifa Anwar, Haseeb Niaz, David Choffnes, et al. Investigating Interdomain Routing Policies in the Wild. In *Proceedings of the 2015 Internet Measurement Conference*, pages 71–77. ACM, 2015.

[24] AnyIP. Ipv4 address lease plans. `http://anyip.com/`, 2020. last-accessed: June 3rd, 2020.

[25] APNIC. stats. Available at `https://ftp.apnic.net/apnic/stats/apnic/` Last accessed: May 31st, 2020.

[26] APNIC. Apnic ipv4 address pool reaches final /8. `https://mailman.apnic.net/mailing-lists/apnic-announce/archive/2011/04/msg00002.html`, 2011. last-accessed: June 3rd, 2020.

[27] APNIC. Apnic guidelines for ipv4 allocation and assignment requests. `https://www.apnic.net/about-apnic/corporate-documents/documents/resource-guidelines/ipv4-guidelines/`, 2014. Last accessed: June 28th, 2020.

[28] APNIC. When will apnic's ipv4 pool be exhausted? https://www.apnic.net/community/ipv4-exhaustion/ipv4-exhaustion-details/, 2014. last-accessed: June 3rd, 2020.

[29] APNIC. Waiting list for unmet ipv4 requests. https://www.apnic.net/get-ip/get-ip-addresses-asn/unmet-ipv4-requests/, 2019. last-accessed: June 3rd, 2020.

[30] APNIC. How much does it cost? https://www.apnic.net/get-ip/apnic-membership/how-much-does-it-cost/, 2020. last-accessed: 03/29/2020.

[31] APNIC. Latest transfers. https://ftp.apnic.net/stats/apnic/transfers/transfers_latest.json, 2020. Last accessed: June 25th, 2020.

[32] APNIC. Extended delegations, 20180405. https://ftp.apnic.net/stats/apnic/2018/delegated-apnic-extended-20180405.gz, 2021. Last accessed: 24th April, 2021.

[33] Maria Apostolaki, Gian Marti, Jan Müller, and Laurent Vanbever. Sabre: Protecting bitcoin against routing attacks. *arXiv preprint arXiv:1808.06254*, 2018.

[34] Malte Appel, Emile Aben, and Romain Fontugne. Metis: Better atlas vantage point selection for everyone. In *2022 Network Traffic Measurement and Analysis Conference (TMA)*, 2022.

[35] APRICOT. Future APRICOTs. Available at https://www.apricot.net/, 2021. last-accessed: Tuesday, 20th April 2021.

[36] Lorenzo Ariemma, Simone Liotta, Massimo Candela, and Giuseppe Di Battista. Long-lasting sequences of bgp updates. In *International Conference on Passive and Active Network Measurement*, pages 213–229. Springer, 2021.

[37] ARIN. stats. Available at https://ftp.arin.net/pub/stats/arin/ Last accessed: May 31st, 2020.

[38] ARIN. Fee schedule. https://www.arin.net/resources/fees/fee_schedule/, 2020. last-accessed: 03/29/2020.

[39] ARIN. Ip address blocks arin issues from. https://www.arin.net/reference/research/statistics/ip_blocks/, 2020. last-accessed: June 3rd, 2020.

[40] ARIN. Latest transfers. https://ftp.arin.net/pub/stats/arin/transfers/transfers_latest.json, 2020. Last accessed: June 25th, 2020.

[41] ARIN. Number resource policy manual. https://www.arin.net/participate/policy/nrpm/4-1-5-resource-request-size, 2020. Last accessed: June 28th, 2020.

[42] ARIN. Registered ipv4 brokers. https://www.apnic.net/manage-ip/manage-resources/transfer-resources/transfer-facilitators/, 2020. Last accessed: June 29th, 2020.

[43] ARIN. Registered transfer facilitators. https://www.arin.net/resources/registry/transfers/stls/registered_facilitators/, 2020. Last accessed: June 29th, 2020.

[44] ARIN. Swip. https://www.arin.net/resources/registry/reassignments/#swip, 2020. Last accessed: October 23rd, 2020.

[45] ARIN. Waiting list status report. https://www.arin.net/resources/guide/ipv4/waiting_list/, 2020. last-accessed: June 3rd, 2020.

[46] ARIN. Extended delegations, 20180405. https://ftp.arin.net/pub/stats/arin/delegated-arin-extended-20180405, 2021. Last accessed: 24th April, 2021.

[47] ARIN. Number Resource Policy Manual. https://www.arin.net/participate/policy/nrpm/#6-5-2-1-size, 2022. Last access: October 2022.

[48] ARISTA. Flexroute engine ip forwarding - network efficiency. https://www.arista.com/en/solutions/flexroute-engine-ip-forwarding, 2022. Last access: September 2022.

[49] Todd Arnold, Jia He, Weifan Jiang, et al. Cloud provider connectivity in the flat internet. In *Proceedings of the ACM Internet Measurement Conference*, pages 230–246, 2020.

[50] Brice Augustin, Xavier Cuvellier, Benjamin Orgogozo, et al. Avoiding traceroute anomalies with paris traceroute. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pages 153–158, 2006.

[51] Alexander Azimov, Eugene Bogomazov, Randy Bush, et al. BGP AS_PATH Verification Based on Resource Public Key Infrastructure (RPKI) Autonomous System Provider Authorization (ASPA) Objects. Internet-Draft draft-ietf-sidrops-aspa-verification-11, Internet Engineering Task Force, October 2022. Work in Progress.

[52] Vaibhav Bajpai, Steffie Jacob Eravuchira, and Jürgen Schönwälder. Lessons learned from using the ripe atlas platform for measurement research. *ACM SIGCOMM Computer Communication Review*, 45(3):35–42, 2015.

[53] Vaibhav Bajpai, Steffie Jacob Eravuchira, Jürgen Schönwälder, et al. Vantage point selection for ipv6 measurements: Benefits and limitations of ripe atlas tags. In *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pages 37–44. IEEE, 2017.

[54] Hitesh Ballani, Paul Francis, and Xinyang Zhang. A Study of Prefix Hijacking and Interception in the Internet. *ACM SIGCOMM Computer Communication Review*, 37(4):265–276, 2007.

[55] Shehar Bano, Philipp Richter, Mobin Javed, et al. Scanning the internet for liveness. *ACM SIGCOMM Computer Communication Review*, 48(2):2–9, 2018.

[56] T. Bates, R. Chandra, D. Katz, and Y. Rekhter. Multiprotocol extensions for bgp-4. RFC 4760, RFC Editor, January 2007. http://www.rfc-editor.org/rfc/rfc4760.txt.

[57] Giuseppe Di Battista, Tiziana Refice, and Massimo Rimondini. How to extract bgp peering information from the internet routing registry. In *Proceedings of the 2006 SIGCOMM workshop on Mining network data*, pages 317–322, 2006.

[58] BBC. 'hacking attacks' hit russian political sites. https://www.bbc.com/news/technology-16032402, 2012.

[59] Dimitris Bertsimas, Angela King, and Rahul Mazumder. Best subset selection via a modern optimization lens. *The annals of statistics*, 44(2):813–852, 2016.

[60] bgp4.as. Bgp looking glasses for ipv4/ipv6, traceroute & bgp route servers. `https://www.bgp4.as/looking-glasses`, 2021. last accessed: 21st. June. 2021.

[61] BGP6-Table. Weekly BGP table movement. `https://twitter.com/bgp6_table/status/1579562700392103937`. Last access: October 2022.

[62] BGPKIT. Fast, extensible, on-premise global bgp monitoring. `https://bgpkit.com/`, 2023.

[63] bgplookingglass.com. Bgp looking glass database. `http://www.bgplookingglass.com/`, 2021. last accessed: 21st. June. 2021.

[64] BGP.Tools. Browse the internet ecosystem. Available at `https://bgp.tools/`, 2022. last-accessed: Sunday, 8th May 2022.

[65] BGP.Tools. Networks that have the following tag: Personal ASN. Available at `https://bgp.tools/tags/perso`, 2022. last-accessed: Sunday, 8th May 2022.

[66] bgpview.io. BGPVIEW. Available at `https://bgpview.io/`, 2022. last-accessed: Sunday, 8th May 2022.

[67] Debopam Bhattacherjee, Waqar Aqeel, Ilker Nadi Bozkurt, et al. Gearing up for the 21st century space race. In *Proceedings of the 17th ACM Workshop on Hot Topics in Networks*, pages 113–119, 2018.

[68] Debopam Bhattacherjee, Waqar Aqeel, Gregory Laughlin, et al. A bird's eye view of the world's fastest networks. In *Proceedings of the ACM Internet Measurement Conference*, pages 521–527, 2020.

[69] Henry Birge-Lee, Liang Wang, Jennifer Rexford, and Prateek Mittal. Sico: Surgical interception attacks by manipulating bgp communities. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 431–448, 2019.

[70] L. Blunk, M. Karir, and C. Labovitz. Multi-threaded routing toolkit (mrt) routing information export format. RFC 6396, RFC Editor, October 2011.

[71] Timm Böttger, Gianni Antichi, Eder L Fernandes, et al. The elusive internet flattening: 10 years of ixp growth. *CoRR*, 2018.

[72] Timm Böttger, Felix Cuadrado, Gareth Tyson, et al. Open connect everywhere: A glimpse at the internet ecosystem through the lens of the netflix cdn. *ACM SIGCOMM Computer Communication Review*, 48(1):28–34, 2018.

[73] Timm Böttger, Felix Cuadrado, and Steve Uhlig. Looking for hypergiants in peeringdb. *ACM SIGCOMM Computer Communication Review*, 48(3):13–19, 2018.

[74] Timm Böttger, Ghida Ibrahim, and Ben Vallis. How the internet reacted to covid-19: A perspective from facebook's edge network. In *Proceedings of the ACM Internet Measurement Conference*, pages 34–41, 2020.

[75] Xander Bouwman, Victor Le Pochat, Pawel Foremski, et al. Helping hands: Measuring the impact of a large threat intelligence sharing community. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1149–1165, 2022.

[76] Samuel Henrique Bucke Brito, Mateus AS Santos, Ramon dos Reis Fontes, et al. Dissecting the Largest National Ecosystem of Public Internet eXchange Points in Brazil. In *International Conference on Passive and Active Network Measurement*, pages 333–345. Springer, 2016.

[77] Andre Broido, Evi Nemeth, and kc claffy. Internet Expansion, Refinement and Churn. *European Transactions on Telecommunications*, 13(1):33–51, 2002.

[78] Coen Bron and Joep Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 1973.

[79] Francesco Bronzino, Nick Feamster, Shinan Liu, et al. Mapping the digital divide: before, during, and after covid-19. In *TPRC48: The 48th Research Conference on Communication, Information and Internet Policy*, 2021.

[80] Tian Bu, Lixin Gao, and Don Towsley. On characterizing bgp routing table growth. In *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, volume 3, pages 2185–2189. IEEE, 2002.

[81] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.

[82] Randy Bush, Olaf Maennel, Matthew Roughan, and Steve Uhlig. Internet optometry: assessing the broken glasses in internet reachability. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*, pages 242–253, 2009.

[83] Matthew Caesar and Jennifer Rexford. Bgp routing policies in isp networks. *IEEE network*, 19(6):5–11, 2005.

[84] CAIDA. Archipelago (Ark) Measurement Infrastructure. Available at `https://www.caida.org/projects/ark/` Last accessed: June 2nd, 2020.

[85] CAIDA. The CAIDA UCSD AS Classification Dataset, 1st February 2020. Available at `https://www.caida.org/data/as-classification` Last accessed: May 30th, 2020.

[86] CAIDA. Routeviews prefix to as mappings dataset for ipv4 and ipv6, 2021/06. `https://www.caida.org/datasets/routeviews-prefix2as/`, 2021. last accessed: 21st. June. 2021.

[87] CAIDA. AS Relationships. `https://www.caida.org/catalog/datasets/as-relationships/`, 2022.

[88] CAIDA. asrank.pl. Available at `https://publicdata.caida.org/datasets/as-relationships/2013-asrank-data-extra/asrank.pl`, 2022. last-accessed: Monday, 25th April 2022.

[89] Caida. The caida as organizations dataset. `http://www.caida.org/data/as-organizations`, 2022.

[90] CAIDA. PeeringDB Dataset. Available at `https://publicdata.caida.org/datasets/peeringdb/`, 2022.

[91] CAIDA. Python API (PyBGPStream). Available at `https://bgpstream.caida.org/docs/api/pybgpstream`, 2022. last-accessed: Monday, 25th April 2022.

[92] Juan Camilo Cardona Restrepo and Rade Stanojevic. Ixp traffic: a macroscopic view. In *Proceedings of the 7th Latin American Networking Conference*, pages 1–8, 2012.

[93] Esteban Carisimo, Alexander Gamero-Garrido, Alex C Snoeren, and Alberto Dainotti. Identifying ases of state-owned internet operators. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 687–702, 2021.

[94] Ignacio Castro, Juan Camilo Cardona, Sergey Gorinsky, and Pierre Francois. Remote peering: More peering without internet flattening. In *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, pages 185–198, 2014.

[95] CH-CENTER. Select a lir affiliation plan. `https://www.ch-center.com/ipv4-allocation`, 2020. last-accessed: June 3rd, 2020.

[96] R. Chandra, P. Traina, and T. Li. Bgp communities attribute. RFC 1997, RFC Editor, August 1996.

[97] Balakrishnan Chandrasekaran, Mingru Bai, Michael Schoenfield, et al. Alidade: Ip geolocation without active probing. *Department of Computer Science, Duke University, Tech. Rep. CS-TR-2015.001*, 2015.

[98] Di-Fa Chang, Ramesh Govindan, and John Heidemann. An empirical study of router response to large bgp routing table load. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 203–208, 2002.

[99] Hyunseok Chang, Ramesh Govindan, Sugih Jamin, et al. Towards capturing representative as-level internet topologies. *Computer Networks*, 44(6):737–755, 2004.

[100] Rocky KC Chang and Michael Lo. Inbound traffic engineering for multihomed ass using as path prepending. *IEEE network*, 19(2):18–25, 2005.

[101] Nikolaos Chatzis, Georgios Smaragdakis, Jan Böttger, et al. On the benefits of using a large ixp as an internet vantage point. In *Proceedings of the 2013 conference on Internet measurement conference*, pages 333–346, 2013.

[102] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[103] Kai Chen, David R Choffnes, Rahul Potharaju, et al. Where the sidewalk ends: Extending the internet as graph using traceroutes from p2p users. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 217–228, 2009.

[104] Davide Chicco, Niklas Tötsch, and Giuseppe Jurman. The matthews correlation coefficient (mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData mining*, 14(1):1–22, 2021.

[105] Marco Chiesa, Roberto di Lallo, Gabriele Lospoto, et al. Prixp: Preserving the privacy of routing policies at internet exchange points. In *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pages 435–441. IEEE, 2017.

[106] Yi-Ching Chiu, Brandon Schlinker, Abhishek Balaji Radhakrishnan, et al. Are we one hop away from a better internet? In *Proceedings of the 2015 Internet Measurement Conference*, pages 523–529, 2015.

[107] Shinyoung Cho, Romain Fontugne, Kenjiro Cho, et al. Bgp hijacking classification. In *2019 Network Traffic Measurement and Analysis Conference (TMA)*, pages 25–32. IEEE, 2019.

[108] Taejoong Chung, Emile Aben, Tim Bruijnzeels, et al. RPKI is coming of age: a longitudinal study of RPKI deployment and invalid route origins. In *Proceedings of the Internet Measurement Conference*, pages 406–419, 2019.

[109] Cisco. BGPStream. https://twitter.com/bgpstream. Last access: October 2022.

[110] Cisco. Cisco IOS XRv 9000 Router Data Sheet. https://www.cisco.com/c/en/us/products/collateral/routers/ios-xrv-9000-router/datasheet-c78-734034.html. last accessed: March 1, 2022.

[111] Cisco. Configuring the BGP Maximum-Prefix Feature. https://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/25160-bgp-maximum-prefix.html. Last access: October 2022.

[112] Cisco. Ipv6 over borderless networks. https://web.archive.org/web/20120128040834/http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html, 2001.

[113] Cisco. Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide, Release 5.3.x . https://www.cisco.com/c/en/us/td/docs/routers/asr9000/software/asr9k_r5-3/routing/configuration/guide/b_routing_cg53xasr9k/b_routing_cg53xasr9k_chapter_010.html, 2015.

[114] Cisco. Influencing Inbound Path Selection by Modifying the AS_PATH Attribute, 2018. Available at https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_bgp/configuration/xe-3se/3850/irg-xe-3se-3850-book/irg-prefix-filter.html Last accessed: April 14th, 2020.

[115] CISCO. Cisco umbrella 1 million. https://umbrella.cisco.com/blog/cisco-umbrella-1-million, 2021. last accessed: 21st. June. 2021.

[116] Cisco. Removing private as numbers from the as path in bgp, 2021.

[117] Cisco. Cisco catalyst 8200 series edge platforms data sheet. https://www.cisco.com/c/en/us/products/collateral/routers/catalyst-8200-series-edge-platforms/nb-06-cat8200-series-edge-plat-ds-cte-en.html, 2022. Last access: October 2022.

[118] Cisco. Cisco catalyst 8500 series edge platforms data sheet. https://www.cisco.com/c/en/us/products/collateral/routers/catalyst-8500-series-edge-platforms/datasheet-c78-744089.html, 2022. Last access: October 2022.

[119] Cisco. Global internet adoption and devices and connection, 2022. available at: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[120] Luca Cittadini, Wolfgang Mühlbauer, Steve Uhlig, et al. Evolution of internet address space deaggregation: Myths and reality. *IEEE Journal on Selected Areas in Communications*, 28(8):1238–1249, 2010.

[121] Luca Cittadini, Stefano Vissicchio, and Benoit Donnet. On the quality of bgp route collectors for ibgp policy inference. In *2014 IFIP Networking Conference*, pages 1–9. IEEE, 2014.

[122] Cloudflare. Project myriagon: Cloudflare passes 10,000 connected networks. https://blog.cloudflare.com/10000-networks-and-beyond/, 2021.

[123] COIN-OR Foundation. Cbc. https://github.com/coin-or/Cbc, 2022. Last access: September 2022.

[124] COUGAR. Cougar looking glass utility. https://github.com/Cougar/lg, 2021. last accessed: 21st. June. 2021.

[125] Ben Cox. https://blog.benjojo.co.uk/post/bgp-battleships. https://blog.benjojo.co.uk/post/bgp-battleships, 2018.

[126] John Curran. Announcing address resources. https://www.ripe.net/participate/mail/forum/address-policy-wg/PDUyOEU3QzkzLjkwOUBpbmV4LmllPg, 2013. Last accessed: June 28th, 2020.

[127] John Curran. Arin ipv4 free pool reaches zero. https://www.arin.net/vault/announcements/2015/20150924.html, 2015. last-accessed: June 3rd, 2020.

[128] TEAM CYMRU. What is a bogon, and why should i filter it? https://team-cymru.com/community-services/bogon-reference/, 2022. Last accessed: March 29th, 2022.

[129] M. Daniele, B. Haberman, S. Routhier, and J. Schoenwaelder. Textual conventions for internet network addresses. RFC 4001, RFC Editor, February 2005.

[130] Darren O'Connor. BGP6-Table. https://twitter.com/bgp6_table. Last access: October 2022.

[131] DE-CIX. Alice-lg - your friendly looking glass. https://github.com/alice-lg/alice-lg. last accessed: Sept. 2022.

[132] DE-CIX. Highest jump ever: DE-CIX Frankfurt reaches 9.1 Tbps, 2020. Available at https://www.de-cix.net/de/news-events/news/de-cix-frankfurt-reaches-9-1-tbps Last accessed: April 14th, 2020.

[133] DE-CIX. Bgp announcement filtering, 2021.

[134] DE-CIX. De-cix frankfurt statistics. https://www.de-cix.net/en/locations/germany/frankfurt/statistics, 2021. Last accessed: 27th June, 2021. Archived version available at:https://web.archive.org/web/20210620110006/https://www.de-cix.net/en/locations/germany/frankfurt/statistics.

[135] DE-CIX. Traffic frankfurt – 1 year. https://www.de-cix.net/en/locations/frankfurt/statistics, 2023.

[136] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. RFC 2460 (Draft Standard), December 1998. Obsoleted by RFC 8200, updated by RFCs 5095, 5722, 5871, 6437, 6564, 6935, 6946, 7045, 7112.

[137] Julián M Del Fiore, Pascal Merindol, Valerio Persico, et al. Filtering the noise to reveal inter-domain lies. In *2019 Network Traffic Measurement and Analysis Conference (TMA)*, pages 17–24. IEEE, 2019.

[138] DELL. set as-path, 2020. Available at `https://www.dell.com/support/manuals/de/de/debsdt1/networking-z9100/z9100-on-9.14.2.6-cli-pub/set-as-path?guid=guid-f2652337-11a3-4dce-bc31-6bd3729bfbf3&lang=en-us` Last accessed: April 4th, 2020.

[139] Wenping Deng, Peidong Zhu, Xicheng Lu, and Bernhard Plattner. On evaluating bgp routing stress attack. *J. Commun.*, 5(1):13–22, 2010.

[140] DENOG. Ipv6 capability status for the top 500 websites according to alexa. `http://www.delong.com/ipv6_alexa500.html`, 2020. Last accessed: June 29th, 2020.

[141] Deploymentcode. Lease ipv4 space. `https://whmcs.deploymentcode.com/ip-subnets.php`, 2020. last-accessed: June 3rd, 2020.

[142] DevelApp. Price overview. `https://develapp.me/en/rent-ip-address`, 2020. last-accessed: June 3rd, 2020.

[143] DeviceAtlas. Which devices have browsers? `https://deviceatlas.com/blog/which-devices-have-browsers`, 2023.

[144] Amogh Dhamdhere, David D Clark, Alexander Gamero-Garrido, et al. Inferring persistent interdomain congestion. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, pages 1–15, 2018.

[145] Giuseppe Di Battista, Maurizio Patrignani, and Maurizio Pizzonia. Computing the types of the relationships between autonomous systems. In *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428)*, volume 1, pages 156–165. IEEE, 2003.

[146] Giuseppe Di Battista, Maurizio Patrignani, Maurizio Pizzonia, and Massimo Rimondini. Towards Optimal Prepending for Incoming Traffic Engineering. In *3rd International Workshop on Internet Performance, Simulation, Monitoring, and Measurement (IPS MoMe 2005)*, 2005.

[147] Constanze Dietrich, Katharina Krombholz, Kevin Borgolte, and Tobias Fiebig. Investigating system operators' perspective on security misconfigurations. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 1272–1289, 2018.

[148] Christoph Dietzel, Anja Feldmann, and Thomas King. Blackholing at ixps: On the effectiveness of ddos mitigation in the wild. In *International Conference on Passive and Active Network Measurement*, pages 319–332. Springer, 2016.

[149] Christoph Dietzel, Matthias Wichtlhuber, Georgios Smaragdakis, and Anja Feldmann. Stellar: network attack mitigation using advanced blackholing. In *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*, pages 152–164, 2018.

[150] Xenofontas Dimitropoulos, Dmitri Krioukov, Marina Fomenkov, et al. As relationships: Inference and validation. *ACM SIGCOMM Computer Communication Review*, 37(1):29–40, 2007.

[151] Benoit Donnet and Olivier Bonaventure. On BGP Communities. *ACM SIGCOMM Computer Communication Review*, 38(2):55–59, 2008.

[152] Gert Döring. Pv6 peering thoughts. `https://www.space.net/~gert/RIPE/EPF4-IPv6-panel.pdf`, 2009.

[153] Gert Döring. Ipv6 bgp filter recommendations, 2013.

[154] DrPeering. The art of peering: The peering playbook. `http://drpeering.net/white-papers/Art-Of-Peering-The-Peering-Playbook.html`, 2022. Last accessed: March 29th, 2022.

[155] Ben Du, Cecilia Testart, Romain Fontugne, et al. Mind your manrs: measuring the manrs ecosystem. In *Proceedings of the 22nd ACM Internet Measurement Conference*, pages 716–729, 2022.

[156] J. Durand, I. Pepelnjak, and G. Doering. BGP Operations and Security. RFC 7454 (Best Current Practice), February 2015.

[157] Benjamin Edelman and Michael Schwarz. Pricing and efficiency in the market for ip addresses. *American Economic Journal: Microeconomics*, 7(3):1–23, 2015.

[158] Marco Hogewoning Emile Aben. 512k-mageddon? `https://labs.ripe.net/author/emileaben/512k-mageddon/`, 2014. last accessed: 19th Sept. 2021.

[159] Euro-IX. Ixpdb. `https://www.euro-ix.net/`, 2022. last accessed: Sept. 2022.

[160] Exa Networks. Exabgp on github. `https://github.com/Exa-Networks/exabgp`. last accessed: March 1, 2022.

[161] Alex Fabrikant, Umar Syed, and Jennifer Rexford. There's something about mrai: Timing diversity can exponentially worsen bgp convergence. In *2011 Proceedings IEEE INFOCOM*, pages 2975–2983. IEEE, 2011.

[162] Adriano Faggiani, Enrico Gregori, Alessandro Improta, et al. A study on traceroute potentiality in revealing the internet as-level topology. In *2014 IFIP Networking Conference*, pages 1–9. IEEE, 2014.

[163] Rodérick Fanou, Pierre Francois, and Emile Aben. On the Diversity of Interdomain Routing in Africa. In *International Conference on Passive and Active Network Measurement*, pages 41–54. Springer, 2015.

[164] Nick Feamster, Jay Borkenhagen, and Jennifer Rexford. Guidelines for Interdomain Traffic Engineering. *ACM SIGCOMM Computer Communication Review*, 33(5):19–30, 2003.

[165] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, et al. The lockdown effect: Implications of the covid-19 pandemic on internet traffic. In *Proceedings of the ACM Internet Measurement Conference*, pages 1–18, 2020.

[166] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, et al. A year in lockdown: How the waves of covid-19 impact internet traffic. *Communications of the ACM Research Highlights*, 64, June 2021.

[167] Guoyao Feng, Srinivasan Seshan, and Peter Steenkiste. Unari: an uncertainty-aware approach to as relationships inference. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, pages 272–284, 2019.

[168] Alberto Fernández, Salvador Garcia, Francisco Herrera, and Nitesh V Chawla. Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, 61:863–905, 2018.

[169] Fernando Ferri, Patrizia Grifoni, and Tiziana Guzzo. Online learning and emergency remote teaching: Opportunities and challenges in emergency situations. *Societies*, 10(4):86, 2020.

[170] Jérôme Fleury, Tom Strickx, and Martin J. Levy. Anatomy of a route leak. `https://ripe84.ripe.net/presentations/36-Cloudflare-Anatomy-of-a-Route-Leak-RIPE84-1.pdf`, 2022.

[171] Romain Fontugne, Esteban Bautista, Colin Petrie, et al. Bgp zombies: An analysis of beacons stuck routes. In *International Conference on Passive and Active Network Measurement*, pages 197–209. Springer, 2019.

[172] Romain Fontugne, Anant Shah, and Emile Aben. The (thin) bridges of as connectivity: Measuring dependency using as hegemony. In *International Conference on Passive and Active Network Measurement*, pages 216–227. Springer, 2018.

[173] FRRouting. Using AS Path in Route Map. Available at `http://docs.frrouting.org/en/latest/bgp.html#bgp-router-configuration` Last accessed: April 4th, 2020, 2020.

[174] Julien Gamba, Romain Fontugne, Cristel Pelsser, et al. Bgp table fragmentation: what & who? In *Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication*, 2017.

[175] Lixin Gao. On Inferring Autonomous System Relationships in the Internet. *IEEE/ACM Transactions on networking*, 9(6):733–745, 2001.

[176] Lixin Gao and Jennifer Rexford. Stable Internet Routing Without Global Coordination. *IEEE/ACM Transactions on Networking*, 9(6):681–692, 2001.

[177] Oliver Gasser, Quirin Scheitle, Pawel Foremski, et al. Clusters in the expanse: understanding and unbiasing ipv6 hitlists. In *Proceedings of the Internet Measurement Conference 2018*, pages 364–378, 2018.

[178] GetIPAddresses. Our pricing. `https://getipaddresses.com/`, 2020. last-accessed: June 3rd, 2020.

[179] Petros Gigis, Matt Calder, Lefteris Manassakis, et al. Seven Years in the Life of Hypergiants' Off-Nets. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, pages 516–533, 2021.

[180] Phillipa Gill, Michael Schapira, and Sharon Goldberg. A survey of interdomain routing policies. *ACM SIGCOMM Computer Communication Review*, 44(1):28–34, 2013.

[181] Rajvir Gill, Ravinder Paul, and Ljiljana Trajković. Effect of mrai timers and routing policies on bgp convergence times. In *2012 IEEE 31st International Performance Computing and Communications Conference (IPCCC)*, pages 314–323. IEEE, 2012.

[182] Vasileios Giotsas, Amogh Dhamdhere, and Kimberly C Claffy. Periscope: Unifying looking glass querying. In *International Conference on Passive and Active Network Measurement*, pages 177–189. Springer, 2016.

[183] Vasileios Giotsas, Christoph Dietzel, Georgios Smaragdakis, et al. Detecting peering infrastructure outages in the wild. In *Proceedings of the conference of the ACM special interest group on data communication*, pages 446–459, 2017.

[184] Vasileios Giotsas, Ioana Livadariu, and Petros Gigis. A first look at the misuse and abuse of the ipv4 transfer market. In *International Conference on Passive and Active Network Measurement*, pages 88–103. Springer, 2020.

[185] Vasileios Giotsas, Matthew Luckie, Bradley Huffaker, and KC Claffy. Inferring complex as relationships. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 23–30, 2014.

[186] Vasileios Giotsas, Matthew Luckie, Bradley Huffaker, and Kc Claffy. Ipv6 as relationships, cliques, and congruence. In *International Conference on Passive and Active Network Measurement*, pages 111–122. Springer, 2015.

[187] Vasileios Giotsas, Georgios Smaragdakis, Christoph Dietzel, et al. Inferring bgp blackholing activity in the internet. In *Proceedings of the 2017 Internet Measurement Conference*, pages 1–14, 2017.

[188] Vasileios Giotsas, Shi Zhou, Matthew Luckie, and Kc Claffy. Inferring multilateral peering. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, pages 247–258, 2013.

[189] Google. COVID-19 Community Mobility Reports, 2020. Available at `https://www.google.com/covid19/mobility/` Last accessed: June 1st, 2020.

[190] Google. Announcing the blue and raman subsea cable systems. `https://cloud.google.com/blog/products/infrastructure/announcing-the-blue-and-raman-subsea-cable-systems?hl=en`, 2021.

[191] Google. IPv6 Adoption. `https://www.google.com/intl/en/ipv6/statistics.html`, 2023. Last accessed: January 25th, 2023.

[192] Caitlin Gray, Clemens Mosig, Randy Bush, et al. Bgp beacons, network tomography, and bayesian computation to locate route flap damping. In *Proceedings of the ACM Internet Measurement Conference*, pages 492–505, 2020.

[193] Thomas Green, Anthony Lambert, Cristel Pelsser, and Dario Rossi. Leveraging inter-domain stability for bgp dynamics analysis. In *International Conference on Passive and Active Network Measurement*, pages 203–215. Springer, 2018.

[194] Enrico Gregori, Alessandro Improta, Luciano Lenzini, et al. On the incompleteness of the as-level graph: a novel methodology for bgp route collector placement. In *Proceedings of the 2012 Internet Measurement Conference*, pages 253–264, 2012.

[195] Enrico Gregori, Alessandro Improta, Luciano Lenzini, et al. A novel methodology to address the internet as-level data incompleteness. *IEEE/ACM Transactions on Networking*, 23(4):1314–1327, 2014.

[196] Brander Group. Buy ipv4 addresses, wholesale ipv4 connectivity and cloud solutions. `https://brandergroup.net/`, 2020. last-accessed: June 3rd, 2020.

[197] IPv4 Market Group. Setting the standard for ipv4 transfers. https://ipv4marketgroup.com/, 2020. last-accessed: June 3rd, 2020.

[198] Arpit Gupta, Matt Calder, Nick Feamster, et al. Peering at the Internet's Frontier: A First Look at ISP Interconnectivity in Africa. In *International Conference on Passive and Active Network Measurement*, pages 204–213. Springer, 2014.

[199] CNN Business Hadas Gold. Netflix and YouTube are slowing down in Europe to keep the internet from breaking, 2020. Available at https://edition.cnn.com/2020/03/19/tech/netflix-internet-overload-eu/index.html Last accessed: June 1st, 2020.

[200] Coleen Haggerty. Google is planning a $9.5 billion expansion of its offices and data centers. https://www.popsci.com/technology/google-plan-facilities-expansion/, 2022.

[201] Hodjat Hamidi and Maryam Jahanshahifard. The role of the internet of things in the improvement and expansion of business. *Journal of Organizational and End User Computing (JOEUC)*, 30(3):24–44, 2018.

[202] Lasse Haugen. Alexa top 1 million websites - tagged by ipv6-compatibility. https://whynoipv6.com/, 2023.

[203] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pages 1322–1328. IEEE, 2008.

[204] Yihua He, Georgos Siganos, Michalis Faloutsos, and Srikanth Krishnamurthy. Lord of the links: a framework for discovering missing links in the internet topology. *IEEE/ACM Transactions On Networking*, 17(2):391–404, 2008.

[205] HEFICED. Service pricing. https://www.heficed.com/pricing, 2020. last-accessed: June 3rd, 2020.

[206] John Heidemann and Christos Papdopoulos. Uses and challenges for network datasets. In *Proc. IEEE CATCH*, 2009.

[207] J. Heitz, J. Snijders, K. Patel, et al. Bgp large communities attribute. RFC 8092, RFC Editor, February 2017.

[208] HETZNER. Ipv4 addresses. https://wiki.hetzner.de/index.php/IP-Adressen/en, 2020. last-accessed: June 3rd, 2020.

[209] R. Hinden and S. Deering. Ip version 6 addressing architecture. RFC 4291, RFC Editor, February 2006. http://www.rfc-editor.org/rfc/rfc4291.txt.

[210] Robert M. Hinden and Stephen E. Deering. Ip version 6 addressing architecture. RFC 2373, RFC Editor, July 1998. http://www.rfc-editor.org/rfc/rfc2373.txt.

[211] Rahul Hiran, Niklas Carlsson, and Phillipa Gill. Characterizing Large-scale Routing Anomalies: A Case Study of the China Telecom Incident. In *International Conference on Passive and Active Network Measurement*. Springer, 2013.

[212] Thomas Holterbach, Cristel Pelsser, Randy Bush, and Laurent Vanbever. Quantifying interference between measurements on the ripe atlas platform. In *Proceedings of the 2015 Internet Measurement Conference*, pages 437–443, 2015.

[213] Michio Honda, Yoshifumi Nishida, Costin Raiciu, et al. Is it still possible to extend tcp? In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 181–194, 2011.

[214] HOSTHONEY. Ipv4 lease / rent addresses. https://www.hosthoney.com/ipv4-lease.php, 2020. last-accessed: June 3rd, 2020.

[215] Allen D Householder, Garret Wassermann, Art Manion, and Chris King. The cert guide to coordinated vulnerability disclosure. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Pittsburgh United States, 2017.

[216] R. Housley, J. Curran, G. Huston, and D. Conrad. The internet numbers registry system. RFC 7020, RFC Editor, August 2013. http://www.rfc-editor.org/rfc/rfc7020.txt.

[217] Geoff Houston. IPv6 CIDR REPORT for 10 Oct 22. https://www.cidr-report.org/v6/as2.0/. last accessed: March 1, 2022.

[218] Geoff Houston. Address span metrics. https://bgp.potaroo.net/as6447/, 2021. Last accessed: 27th June, 2021.

[219] B. Huffaker, K. Keys, M. Fomenkov, and K. Claffy. As-to-organization dataset. http://www.caida.org/research/topology/, 2022.

[220] Bradley Huffaker, Amogh Dhamdhere, Marina Fomenkov, et al. Toward topology dualism: improving the accuracy of as annotations for routers. In *International Conference on Passive and Active Network Measurement*, pages 101–110. Springer, 2010.

[221] Hurricane Electric. Hurricane Electric Route Filtering Algorithm. https://routing.he.net/algorithm.html. Last access: October 2022.

[222] G. Huston and G. Michaelson. Validation of Route Origination Using the Resource Certificate Public Key Infrastructure (PKI) and Route Origin Authorizations (ROAs). RFC 6483 (Informational), February 2012.

[223] Geoff Huston. As131072 ipv4 cidr report. https://www.cidr-report.org/as2.0/. Last access: October 2022.

[224] Geoff Huston. As131072 ipv6 bgp table data. https://bgp.potaroo.net/v6/as2.0/index.html. last accessed: 19th Sept. 2021.

[225] Geoff Huston. As131072 ipv6 bgp table report: average prefix length. https://bgp.potaroo.net/cgi-bin/plota?file=/var/data/bgp/v6/as2.0/bgp-average-prefix.txt. Last access: October 2022.

[226] Geoff Huston. As131072 ipv6 cidr report: possible aggregation gains. https://www.cidr-report.org/v6/as2.0/#Gains. Last access: October 2022.

[227] Geoff Huston. BGP more specifics: routing vandalism or useful?, 2017.

[228] IANA. Border gateway protocol (bgp) well-known communities. https://www.iana.org/assignments/bgp-well-known-communities/bgp-well-known-communities.xhtml, 2021.

[229] IANA. Autonomous System (AS) Numbers. Available at https://www.iana.org/assignments/as-numbers/as-numbers.xhtml, 2022. last-accessed: Tuesday, 22nd April 2022.

[230] iFog GmbH. Ixp access. https://ifog.ch/en/ip/ixp-access-vm-amsterdam. Last access: October 2022.

[231] IIJ. Internet Health Report. Available at https://ihr.iijlab.net/ihr/en-us/, 2022. last-accessed: Sunday, 8th May 2022.

[232] Investopedia. How to identify a stock under consolidation? https://www.investopedia.com/ask/answers/120414/how-do-i-identify-stock-under-consolidation.asp, 2020. Last accessed: June 3rd, 2020.

[233] IP-AS. Ipv4 resources leasing. http://ip-as.com/, 2020. last-accessed: June 3rd, 2020.

[234] ipinfo.io. The trusted source for IP address data. Available at https://ipinfo.io/, 2022. last-accessed: Sunday, 8th May 2022.

[235] ipinsight.io. Looking glass. https://whois.ipinsight.io/looking-glass/, 2021. last accessed: 21st. June. 2021.

[236] IPRoyal. Pricing. https://iproyal.com/, 2020. last-accessed: June 3rd, 2020.

[237] IPTrading.com. Buy ipv4 addresses. https://iptrading.com/resource/buy-ipv4-addresses/, 2020. last-accessed: June 3rd, 2020.

[238] IPV4Broker. Ip lease. https://ipv4broker.net/en/, 2020. last-accessed: June 3rd, 2020.

[239] IPv4.Global. Ipv4 prior sales. https://auctions.ipv4.global/prior-sales, 2020. last-accessed: June 3rd, 2020.

[240] IPv4Mall. Lease ipv4 addresses. https://ipv4mall.com/lease-ipv4/, 2020. last-accessed: June 3rd, 2020.

[241] IPXO. Ipv4 price history. https://www.ipxo.com/blog/ipv4-price-history/, 2022.

[242] irr.net. List of routing registries, 2021.

[243] Isolario. BGPScanner. Available at https://gitlab.com/Isolario/bgpscanner/-/wikis/Home, 2022. Last-accessed: Monday, 25th April 2022.

[244] isolario.it (via Internet Archive). Isolario Project. Available at https://web.archive.org/web/20220118192649/https://isolario.it/, 2022. last-accessed: Sunday, 8th May 2022.

[245] IX.br. Total traffic. https://ix.br/agregado/, 2021. Last accessed: 27th June, 2021. Archived version available at:https://web.archive.org/web/20210627071318/https://ix.br/agregado/.

[246] Dhananjay Jagtap, Alex Yen, Huanlei Wu, et al. Federated infrastructure: usage, patterns, and insights from" the people's network". In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 22–36, 2021.

[247] Siyuan Jia, Matthew Luckie, Bradley Huffaker, et al. Tracking the deployment of ipv6: Topology, routing and performance. *Computer Networks*, 165:106947, 2019.

[248] Yuchen Jin, Colin Scott, Amogh Dhamdhere, et al. Stable and practical {AS} relationship inference with problink. In *16th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 19)*, pages 581–598, 2019.

[249] Yunchen Jin. Problink. GitHub repository: `https://github.com/YuchenJin/ProbLink`, 2019.

[250] Ziting Jin. Toposcope. GitHub repository: `https://github.com/Zitong-Jin/TopoScope`, 2020.

[251] Zitong Jin, Xingang Shi, Yan Yang, et al. Toposcope: Recover as relationships from fragmentary observations. In *Proceedings of the ACM Internet Measurement Conference*, pages 266–280, 2020.

[252] Juniper. Features supported on MX5 in Junos OS 21.2R3. `https://apps.juniper.net/home/mx5/features`. last accessed: March 1, 2022.

[253] Juniper. Example: Configuring a Routing Policy to Prepend the AS Path, 2019. Available at `https://www.juniper.net/documentation/en_US/junos/topics/example/routing-policy-security-routing-policy-to-prepend-to-as-path-configuring.html` Last accessed: April 4th, 2020.

[254] Juniper. Understanding bgp confederations, 2021.

[255] Costas Kalogiros, Marcelo Bagnulo, and Alexandros Kostopoulos. Understanding incentives for prefix aggregation in bgp. In *Proceedings of the 2009 workshop on Re-architecting the internet*, pages 49–54, 2009.

[256] Thomas Karagiannis, Konstantina Papagiannaki, and Michalis Faloutsos. Blinc: multilevel traffic classification in the dark. In *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 229–240, 2005.

[257] Josh Karlin, Stephanie Forrest, and Jennifer Rexford. Pretty good bgp: Improving bgp by cautiously adopting routes. In *Proceedings of the 2006 IEEE International Conference on Network Protocols*, pages 290–299. IEEE, 2006.

[258] Ethan Katz-Bassett, David R Choffnes, Ítalo Cunha, et al. Machiavellian routing: improving internet availability with bgp poisoning. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, pages 1–6, 2011.

[259] Ethan Katz-Bassett, Harsha V Madhyastha, Vijay Kumar Adhikari, et al. Reverse traceroute. In *NSDI*, volume 10, pages 219–234, 2010.

[260] Ethan Katz-Bassett, Colin Scott, David R Choffnes, et al. Lifeguard: Practical repair of persistent route failures. *ACM SIGCOMM Computer Communication Review*, 42(4):395–406, 2012.

[261] Erin Kenneally and David Dittrich. The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research. *Available at SSRN 2445102*, 2012.

[262] Stephen Kent, Charles Lynn, and Karen Seo. Secure border gateway protocol (s-bgp). *IEEE Journal on Selected areas in Communications*, 18(4):582–592, 2000.

[263] Thomas Kernen. Looking glass. `http://traceroute.org/`, 2021. last accessed: 21st. June. 2021.

[264] T. King, C. Dietzel, J. Snijders, et al. Blackhole community. RFC 7999, RFC Editor, October 2016.

[265] Selira Kotoua, Mustafa Ilkan, and Hasan Kilic. The growing of online education in sub saharan africa: Case study ghana. *Procedia-Social and Behavioral Sciences*, 191:2406–2411, 2015.

[266] A. Krause and D. Golovin. Submodular function maximization. *Tractability: Practical Approaches to Hard Problems*, 3(19):8, 2012.

[267] Thomas Krenc, Robert Beverly, and Georgios Smaragdakis. AS-level BGP community usage classification. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 577–592, 2021.

[268] Thomas Krenc and Anja Feldmann. Bgp prefix delegations: A deep dive. In *Proceedings of the 2016 Internet Measurement Conference*, pages 469–475, 2016.

[269] Ivana Kursan and Mirela Mihić. Business intelligence: The role of the internet in marketing research and business decision-making. *Management: journal of contemporary management issues*, 15(1):69–86, 2010.

[270] Craig Labovitz. Internet traffic 2009-2019. slides at `https://www.lacnic.net/innovaportal/file/4016/1/lacnog-internet-traffic-2009-2019.pdf` and recording at `https://www.youtube.com/watch?v=jGnVcCQUCdk`., 2020.

[271] LACNIC. stats. Available at `https://ftp.lacnic.net/pub/stats/lacnic/` Last accessed: May 31st, 2020.

[272] LACNIC. Policies relating to the exhaustion of ipv4 address space. `https://www.lacnic.net/691/2/lacnic/11-policies-relating-to-the-exhaustion-of-ipv4-address-space`, 2012. last-accessed: June 3rd, 2020.

[273] LACNIC. Ipv4 exhaustion: Lacnic has assigned the last remaining address block. `https://www.lacnic.net/4848/2/lacnic/ipv4-exhaustion:-lacnic-has-assigned-the-last-remaining-address-block`, 2020. Last accessed: October 23rd, 2020.

[274] LACNIC. Latest transfers. `https://ftp.lacnic.net/pub/stats/lacnic/transfers/transfers_latest.json`, 2020. Last accessed: June 25th, 2020.

[275] Lacnic. Membership categories and fees. `https://www.lacnic.net/2399/2/lacnic/membership-categories-and-fees`, 2020. last-accessed: 03/29/2020.

[276] LACNIC. Phases of ipv4 exhaustion. `https://www.lacnic.net/1039/2/lacnic/ipv4-depletion-phases`, 2020. last-accessed: June 3rd, 2020.

[277] LACNIC. Waiting list. https://www.lacnic.net/1039/2/lacnic/phases-of-ipv4-exhaustion, 2020. Last accessed: October 23rd, 2020.

[278] LACNIC. Extended delegations, 20180405. https://ftp.lacnic.net/pub/stats/lacnic/delegated-lacnic-extended-20180405, 2021. Last accessed: 24th April, 2021.

[279] Anukool Lakhina, Konstantina Papagiannaki, Mark Crovella, et al. Structural analysis of network traffic flows. In *Proceedings of the joint international conference on Measurement and modeling of computer systems*, pages 61–72, 2004.

[280] Franck Le, Geoffrey G Xie, and Hui Zhang. On route aggregation. In *Proceedings of the seventh conference on emerging networking experiments and technologies*, pages 1–12, 2011.

[281] M. Lepinski and K. Sriram. Bgpsec protocol specification. RFC 8205, RFC Editor, September 2017.

[282] Kirtus G Leyba, Joshua J Daymude, Jean-Gabriel Young, et al. Cutting through the noise to infer autonomous system topology. *arXiv preprint arXiv:2201.07328*, 2022.

[283] LIR.SERVICES. Pricing. https://www.lir.services/ip-address-lease/, 2020. last-accessed: June 3rd, 2020.

[284] Suqi Liu, Ian Foster, Stefan Savage, et al. Who is. com? learning to parse whois records. In *Proceedings of the 2015 Internet Measurement Conference*, pages 369–380, 2015.

[285] Ioana Livadariu, Ahmed Elmokashfi, and Amogh Dhamdhere. On ipv4 transfer markets: Analyzing reported transfers and inferring transfers in the wild. *Computer Communications*, 111:105–119, 2017.

[286] Ioana Livadariu, Ahmed Elmokashfi, Amogh Dhamdhere, and KC Claffy. A first look at ipv4 transfer markets. In *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, pages 7–12, 2013.

[287] Sharon Loane. The role of the internet in the internationalisation of small and medium sized companies. *Journal of International Entrepreneurship*, 3(4):263–277, 2005.

[288] Aemen Lodhi, Natalie Larson, Amogh Dhamdhere, et al. Using peeringdb to understand the peering ecosystem. *ACM SIGCOMM Computer Communication Review*, 44(2):20–27, 2014.

[289] logicweb. Bulk ipv4 address leasing & ipv6 leasing. https://www.logicweb.com/bulk-ip-address-leasing/, 2020. last-accessed: June 3rd, 2020.

[290] Logosnet. Ipv4 address leasing. https://logosnet.cy.net/ip-leasing/, 2020. last-accessed: June 3rd, 2020.

[291] lookinglass.org. Bgp looking glass services. https://lookinglass.org/, 2021. last accessed: 21st. June. 2021.

[292] Matthew Luckie, Bradley Huffaker, Amogh Dhamdhere, et al. As relationships, customer cones, and validation. In *Proceedings of the 2013 conference on Internet measurement conference*, pages 243–256, 2013.

[293] Matthew Luckie, Bradley Huffaker, Alexander Marder, et al. Learning to extract geographic information from internet router hostnames. In *Proceedings of the 17th International Conference on emerging Networking EXperiments and Technologies*, pages 440–453, 2021.

[294] Andra Lutu, Marcelo Bagnulo, Cristel Pelsser, et al. An analysis of the economic impact of strategic deaggregation. *Computer Networks*, 81:147–163, 2015.

[295] Andra Lutu, Marcelo Bagnulo, and Rade Stanojevic. An economic side-effect for prefix deaggregation. In *2012 Proceedings IEEE INFOCOM Workshops*, pages 190–195. IEEE, 2012.

[296] Andra Lutu, Cristel Pelsser, Marcelo Bagnulo, and Kenjiro Cho. The aftermath of prefix deaggregation. In *Proceedings of the 2013 25th International Teletraffic Congress (ITC)*, pages 1–8. IEEE, 2013.

[297] Andra Lutu, Diego Perino, Marcelo Bagnulo, et al. A characterization of the covid-19 pandemic impact on a mobile network operator traffic. In *Proceedings of the ACM internet measurement conference*, pages 19–33, 2020.

[298] Doug Madory. Excessive BGP AS Path Prepending is a Self-Inflicted Vulnerability, 2019. Available at `https://ripe79.ripe.net/archives/video/187/` Last accessed: May 30th, 2020. We reference the questions after the talk.

[299] Doug Madory. Excessive BGP AS Path Prepending is a Self-Inflicted Vulnerability, 2019. Available at `https://blogs.oracle.com/internetintelligence/excessive-as-path-prepending-is-a-self-inflicted-vulnerability` Last accessed: May 30th, 2020.

[300] MailOnline. Hackers attack stock exchange: Cyber criminals take down website for more than two hours as part of protest against world's banks. `https://www.dailymail.co.uk/news/article-3625656/Hackers-attack-Stock-Exchange-Cyber-criminals-website-two-hours-protest-against-world-s-banks.html`, 2016.

[301] Majestic. The majestic million. `https://majestic.com/reports/majestic-million/`, 2021. last accessed: 21st. June. 2021.

[302] G. Malkin. Traceroute using an ip option. RFC 1393, RFC Editor, January 1993.

[303] Edwin Mallette. Ipv4 broker / service -. `https://mailman.nanog.org/pipermail/nanog/2020-June/207972.html`, 2020. Last accessed: June 25th, 2020.

[304] MANRS. RPKI Week. `https://www.manrs.org/resources/events/rpki-week/`. Last access: October 2022.

[305] MANRS. About MANRS. Available at `https://www.manrs.org/about/`, 2021. last-accessed: Tuesday, 20th April 2021.

[306] MANRS. Prefix filter configuration tools, 2021.

[307] Z Morley Mao, Randy Bush, Timothy G Griffin, and Matthew Roughan. Bgp beacons. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 1–14, 2003.

[308] Data Center Map. Cloud servers. https://www.datacentermap.com/cloud.html, 2020. Last accessed: June 29th, 2020.

[309] Pedro Marcos. Can AS-PATH prepending compromise the security of Internet routing?, 2019. Available at https://blog.apnic.net/2019/06/18/can-as-path-prepending-compromise-the-security-of-internet-routing/ Last accessed: May 30th, 2020.

[310] Pedro Marcos, Marco Chiesa, Christoph Dietzel, et al. A survey on the current internet interconnection practices. *ACM SIGCOMM Computer Communication Review*, 50(1):10–17, 2020.

[311] Pedro Marcos, Marco Chiesa, Lucas Müller, et al. Dynam-ix: A dynamic interconnection exchange. In *Proceedings of the 14th International Conference on emerging Networking EXperiments and Technologies*, pages 228–240, 2018.

[312] Pedro Marcos, Lars Prehn, Lucas Leal, et al. As-path prepending: there is no rose without a thorn. In *Proceedings of the ACM Internet Measurement Conference*, pages 506–520, 2020.

[313] Alexander Marder, Matthew Luckie, Amogh Dhamdhere, et al. Pushing the boundaries with bdrmapit: Mapping router ownership at internet scale. In *Proceedings of the Internet Measurement Conference 2018*, pages 56–69, 2018.

[314] P. Marques and F. Dupont. Use of bgp-4 multiprotocol extensions for ipv6 interdomain routing. RFC 2545, RFC Editor, March 1999.

[315] MAXMIND. Geoip2 city accuracy. https://www.maxmind.com/en/geoip2-city-accuracy-comparison?country=&resolution=250&cellular=excluding, 2021. Last accessed: 27th June, 2021. Archived version available at:https://web.archive.org/web/20210627075223/https://www.maxmind.com/en/geoip2-city-accuracy-comparison?country=&resolution=250&cellular=excluding.

[316] MAXMIND. Geolite2 free geolocation data. https://dev.maxmind.com/geoip/geolite2-free-geolocation-data, 2021. Last accessed: 27th June, 2021.

[317] Guillaume Mazoyer, Martijn Schmidt, Alexandre Jeronimo Correa, and 29 others. Respawner looking glass utility. https://github.com/gmazoyer/looking-glass, 2021. last accessed: 21st. June. 2021.

[318] Fabricio Mazzola, Pedro Marcos, Ignacio Castro, et al. On the latency impact of remote peering. In *International Conference on Passive and Active Network Measurement*, pages 367–392. Springer, 2022.

[319] Tyler McDaniel, Jared M Smith, and Max Schuchard. Flexsealing bgp against route leaks: peerlock active measurement and analysis. *Network and Distributed Systems Security (NDSS) Symposium 2021*, 2021.

[320] Sally J McMillan and Margaret Morrison. Coming of age with the internet: A qualitative exploration of how the internet has become an integral part of young people's lives. *New media & society*, 8(1):73–95, 2006.

[321] D. McPherson and V. Gill. BGP MULTI_EXIT_DISC (MED) Considerations. RFC 4451, RFC Editor, March 2006.

[322] Stephen McQuistin, Sree Priyanka Uppu, and Marcel Flores. Taming anycast in the wild internet. In *Proceedings of the Internet Measurement Conference*, pages 165–178, 2019.

[323] Megaport. Data centre interconnect. https://www.megaport.com/services/datacentre-interconnect/. Last access: October 2022.

[324] Xiaoqiao Meng, Zhiguo Xu, Beichuan Zhang, et al. Ipv4 address allocation and the bgp routing table evolution. *ACM SIGCOMM Computer Communication Review*, 35(1):71–80, 2005.

[325] Merit. Internet Routing Registries. https://www.irr.net/. Last access: October 2022.

[326] Meta. Meta's subsea cable investments expected to contribute over half a trillion dollars to asia-pacific and european economies by 2025. https://tech.facebook.com/engineering/2022/2/economic-impact-subsea-cables/, 2022.

[327] Campaign US Michael Heusner. Marilia Mendonça breaks world record with live-streamed concert, 2020. Available at https://www.campaignlive.com/article/marilia-mendonca-breaks-world-record-live-streamed-concert/1680008 Last accessed: June 1st, 2020.

[328] Mikrotik. Main/Backup link setup, 2010. Available at https://wiki.mikrotik.com/wiki/Manual:Simple_BGP_Multihoming Last accessed: April 14th, 2020.

[329] Christopher Mims. Google, amazon, meta and microsoft weave a fiber-optic web of power. https://www.wsj.com/articles/google-amazon-meta-and-microsoft-weave-a-fiber-optic-web-of-power-11642222824, 2022.

[330] P. Mohapatra, J. Scudder, D. Ward, et al. Bgp prefix origin validation. RFC 6811, RFC Editor, January 2013. http://www.rfc-editor.org/rfc/rfc6811.txt.

[331] Reynaldo Morillo, Justin Furuness, Amir Herzberg, et al. Rov++: Improved deployable defense against bgp hijacking. *Network and Distributed Systems Security (NDSS) Symposium 2021*, 2021.

[332] Sebastian Moss. Exclusive: After meta cancels odense data center expansion, projects are being "rescoped" globally. https://www.datacenterdynamics.com/en/news/exclusive-after-meta-cancels-odense-data-center-expansion-other-projects-are-being-rescoped/, 2022.

[333] MSK-IX. Internet traffic peak hit on March 30. https://www.msk-ix.ru/en/press-center/news/?id=20200331, 2020. https://www.ams-ix.net/ams/news/ams-ix-breaks-through-8-tbps-barrier.

[334] Lucas Müller, Matthew Luckie, Bradley Huffaker, et al. Challenges in inferring spoofed traffic at IXPs. In *Proceedings of the 15th International Conference on Emerging Networking Experiments and Technologies*, pages 96–109, 2019.

[335] Lucas Müller, Matthew Luckie, Bradley Huffaker, et al. Github: spoofer-ix. Available at https://github.com/spoofer-ix/spoofer-ix/blob/master/asrel-customer-cone/asrank.pl, 2022. Last-accessed: Wednesday, 30th November 2022.

[336] Johannes Naab, Patrick Sattler, Jonas Jelten, et al. Prefix top lists: Gaining insights with prefixes from domain-based top lists on dns deployment. In *Proceedings of the Internet Measurement Conference*, pages 351–357, 2019.

[337] NANOG. Future NANOG Meetings. Available at https://www.nanog.org/meetings/future/, 2021. last-accessed: Tuesday, 20th April 2021.

[338] Marcin Nawrocki, Jeremias Blendin, Christoph Dietzel, et al. Down the black hole: dismantling operational practices of bgp blackholing at ixps. In *Proceedings of the Internet Measurement Conference*, pages 435–448, 2019.

[339] RIPE NCC. How does the ipv4 waiting list work? https://www.ripe.net/manage-ips-and-asns/ipv4/how-waiting-list-works, 2019. Last accessed: October 25th, 2020.

[340] RIPE NCC. Ipv4 address allocation and assignment policies for the ripe ncc service region. https://www.ripe.net/publications/docs/ripe-733, 2019. Last accessed: June 28th, 2020.

[341] RIPE NCC. What is ipv4 run out? https://www.ripe.net/manage-ips-and-asns/ipv4/ipv4-run-out, 2019. Last accessed: June 28th, 2020.

[342] RIPE NCC. A first for the ripe ncc: Seizure of the "right to registration of ipv4 addresses" for the recovery of money. https://labs.ripe.net/Members/ciaran_byrne/seizure-of-the-right-to-registration-of-ipv4-addresses, 2020. Last accessed: October 25th, 2020.

[343] RIPE NCC. Routing information service (ris). https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris, 2021. last accessed: 21st. June. 2021.

[344] RIPE NCC. Billing, payment and fees. https://www.ripe.net/participate/member-support/payment, 2022. Last access: October 2022.

[345] RIPE NCC. How to request an ipv6 allocation. https://www.ripe.net/manage-ips-and-asns/ipv6/request-ipv6/how-to-request-an-ipv6-allocation, 2022. Last access: October 2022.

[346] RIPE NCC. Ipv6 enabled networks. https://v6asns.ripe.net/v/6, 2022.

[347] RIPE NCC. RIPE Atlas probes map. https://atlas.ripe.net/results/maps/network-coverage/, 2022. Accessed: 07 May 2022.

[348] RIPE NCC. Ris routing beacons. https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/archive/ris-routing-beacons, 2022.

[349] RIPE NCC. Routing Information Service, 2022. Available at http://www.ripe.net/ris/. Last accessed: May 30th, 2020.

[350] RIPE NCC. libbgpdump. https://github.com/RIPE-NCC/bgpdump, 2023.

[351] Eugenio Nerio Nemmi, Francesco Sassi, Massimo La Morgia, et al. The parallel lives of autonomous systems: Asn allocations vs. bgp. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 593–611, 2021.

[352] Fork Networking. Ip address leasing. `http://www.forked.net/ip-address-leasing/`, 2020. last-accessed: June 3rd, 2020.

[353] A. Newton and S. Hollenbeck. JSON Responses for the Registration Data Access Protocol (RDAP). RFC 7483 (Proposed Standard), March 2015.

[354] Vivek Nigam. Reclaiming unused ipv4 address space. `https://blog.apnic.net/2020/07/06/reclaiming-unused-ipv4-address-space/`, 2018. Last accessed: October 21st, 2020.

[355] NLnetLabs. Routinator. Available at `https://github.com/NLnetLabs/routinator` Last accessed: May 31st, 2020.

[356] NLNOG. Filtering small prefixes, 2021.

[357] NLNOG. Nlnog looking glass—communities. `https://github.com/NLNOG/lg.ring.nlnog.net/tree/main/communities`, 2023.

[358] Leslie Nobile. Arin enters phase four of the ipv4 countdown plan. `https://www.arin.net/vault/announcements/2014/20140423.html`, 2014. last-accessed: June 3rd, 2020.

[359] NOCTION. Bgp prefix filtering, 2021.

[360] George Nomikos, Vasileios Kotronis, Pavlos Sermpezis, et al. O peer, where art thou? uncovering remote peering interconnections at ixps. In *Proceedings of the Internet Measurement Conference 2018*, pages 265–278, 2018.

[361] William Norton. 95th Percentile Internet Billing Method. Available at `https://drpeering.net/white-papers/Ecosystems/95th-percentile-measurement-Internet-Transit.html` Last accessed: June 1st, 2020.

[362] William B Norton. *The Internet peering playbook: connecting to the core of the Internet*. DrPeering Press, 2014.

[363] NRO. Internet number resource status report. `https://www.nro.net/wp-content/uploads/NRO-Statistics-2022-Q3-FINAL.pdf`, 2022.

[364] University of Oregon. University of oregon route views project. `http://www.routeviews.org/routeviews/`, 2022.

[365] Ricardo Oliveira, Dan Pei, Walter Willinger, et al. The (In)Completeness of the Observed Internet AS-level Structure. *IEEE/ACM Transactions on Networking*, 18(1):109–122, 2009.

[366] Ricardo Oliveira, Walter Willinger, Beichuan Zhang, et al. Quantifying the completeness of the observed internet as-level structure. *work*, 11(15):13–17, 2008.

[367] Ricardo V Oliveira, Dan Pei, Walter Willinger, et al. In search of the elusive ground truth: the internet's as-level connectivity structure. *ACM SIGMETRICS Performance Evaluation Review*, 36(1):217–228, 2008.

[368] Ricardo V Oliveira, Beichuan Zhang, and Lixia Zhang. Observing the evolution of internet as topology. In *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 313–324, 2007.

[369] Cathy O'Neil. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown, New York, first edition edition, 2016.

[370] Number Resource Organization. Rir comparative policy overview 2020-q1. `https://www.nro.net/wp-content/uploads/comp-pol-2020-q1.pdf`, 2020. Last accessed: October 25th, 2020.

[371] Chiara Orsini, Alistair King, Danilo Giordano, et al. Bgpstream: a software framework for live and historical bgp data analysis. In *Proceedings of the 2016 Internet Measurement Conference*, pages 429–444, 2016.

[372] Chiara Orsini, Alistair King, Danilo Giordano, et al. Bgpstream: a software framework for live and historical bgp data analysis. In *Proceedings of the 2016 Internet Measurement Conference*, pages 429–444, 2016.

[373] Eric Osterweil, Shane Amante, Dan Massey, and Danny McPherson. The great ipv4 land grab: resource certification for the ipv4 grey market. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, pages 1–6, 2011.

[374] Our World in Data. Human Height. Available at `https://ourworldindata.org/human-height`, 2022. Last-accessed: 15 May 2022.

[375] PacketPioneer. Intro to quic - more than just udp! `https://packetpioneer.com/intro-to-quic-more-than-just-udp/`, 2021.

[376] Ramakrishna Padmanabhan, John P Rula, Philipp Richter, et al. Dynamips: Analyzing address assignment practices in ipv4 and ipv6. In *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies*, pages 55–70, 2020.

[377] Craig Partridge and Mark Allman. Ethical considerations in network measurement papers. *Communications of the ACM*, 59(10):58–64, 2016.

[378] PCH. Internet exchange directory. `https://www.pch.net/ixp/dir`, 2022. Last accessed: March 29th, 2022.

[379] PCH. Packet Clearing House. Available at `https://www.pch.net/`, 2022. last-accessed: Sunday, 8th May 2022.

[380] PeeringDB. Hivelocity inc. `https://web.archive.org/web/20220130161818/https://www.peeringdb.com/net/2159`, 2021.

[381] PeeringDB. The Interconnection Database. Available at `https://www.peeringdb.com/`, 2022. last-accessed: Monday, 25th April 2022.

[382] Cristel Pelsser, Luca Cittadini, Stefano Vissicchio, and Randy Bush. From Paris to Tokyo: On the Suitability of ping to Measure Latency. In *Proceedings of the 2013 Internet Measurement Conference*, pages 427–432, 2013.

[383] Cristel Pelsser, Olaf Maennel, Pradosh Mohapatra, et al. Route flap damping made usable. In *International Conference on Passive and Active Network Measurement*, pages 143–152. Springer, 2011.

[384] Colin Petrie. BGP Even-More Specifics in 2017, 2017. Available at https://labs.ripe.net/Members/stephen_strowes/bgp-even-more-specifics-in-2017 Last accessed: April 14th, 2020.

[385] Alice B Popejoy and Stephanie M Fullerton. Genomics is failing on diversity. *Nature News*, 538(7624):161, 2016.

[386] Alin C. Popescu, Brian J. Premore, and Todd Underwood. Anatomy of a leak: As9121. https://www.youtube.com/watch?v=l_UCneZm6dE, 2004.

[387] Angela Pratesi, Wendy Miller, and Elizabeth Sutton. Democratizing knowledge: using wikipedia for inclusive teaching and research in four undergraduate classes. *Radical Teacher*, 114:22–33, 2019.

[388] PREFIXIPv4BROKER. Lease ipv4 addresses. https://www.prefixbroker.com/lease-ipv4/, 2020. last-accessed: June 3rd, 2020.

[389] Lars Prehn and Anja Feldmann. How biased is our validation (data) for as relationships? In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 612–620, 2021.

[390] Lars Prehn, Franziska Lichtblau, Christoph Dietzel, and Anja Feldmann. Peering only? analyzing the reachability benefits of joining large ixps today. In *International Conference on Passive and Active Network Measurement*, pages 338–366. Springer, 2022.

[391] Lars Prehn, Franziska Lichtblau, and Anja Feldmann. When wells run dry: the 2020 ipv4 address market. In *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies*, pages 46–54, 2020.

[392] PROSTOHOST. Rent ipv4 address. https://prostohosting.com/en/services/ipv4lease.html, 2020. last-accessed: June 3rd, 2020.

[393] Enric Pujol, Ingmar Poese, Johannes Zerwas, et al. Steering hyper-giants' traffic at scale. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, pages 82–95, 2019.

[394] pulp documentation team. Optimization with pulp. https://coin-or.github.io/pulp/, 2022. Last access: September 2022.

[395] Bruno Quoitin, Cristel Pelsser, Olivier Bonaventure, and Steve Uhlig. A performance evaluation of BGP-based traffic engineering. *International journal of network management*, 15(3):177–191, 2005.

[396] Bruno Quoitin, Cristel Pelsser, Louis Swinnen, et al. Interdomain traffic engineering with bgp. *IEEE Communications magazine*, 41(5):122–128, 2003.

[397] RADb. aut-num: As213045, 2021.

[398] RADB. Query for AS3356. Available at https://www.radb.net/query?keywords=AS3356, 2021. last-accessed: Tuesday, 22nd April 2021, Internet Archive snapshot: https://web.archive.org/web/20210422074619/https://www.radb.net/query?keywords=AS3356.

[399] Audrey Randall, Enze Liu, Gautam Akiwate, et al. Trufflehunter: Cache snooping rare domains at large public dns resolvers. In *Proceedings of the ACM Internet Measurement Conference*, pages 50–64, 2020.

[400] Rapid 7. Http get responses. Available at `https://opendata.rapid7.com/sonar.http/`.

[401] Rapid7. Rapid7 Open Data. Available at `ttps://opendata.rapid7.com/`, May 2021.

[402] RapidDedi. Rapiddedi ipv4 plans. https://rapiddedi.com/ipv4-space-rent.html, 2020. last-accessed: June 3rd, 2020.

[403] RapidSeedbox. Ipv6 address for rental. `https://www.rapidseedbox.com/ipv4-rental#pricing`, 2022. Last access: October 2022.

[404] IX Reach. Remote peering. `https://www.ixreach.com/services/remote-peering/`. Last access: October 2022.

[405] Y. Rekhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). RFC 4271, RFC Editor, January 2006. `http://www.rfc-editor.org/rfc/rfc4271.txt`.

[406] RentIPv4. Current pricing. `https://rentipv4.com/?service=current-pricing`, 2020. last-accessed: June 3rd, 2020.

[407] A. Retana, R. White, V. Fuller, and D. McPherson. Using 31-Bit Prefixes on IPv4 Point-to-Point Links. RFC 3021 (Proposed Standard), December 2000.

[408] RETN. Remote ix. `https://retn.net/products/remote-ix`. Last access: October 2022.

[409] Andreas Reuter, Randy Bush, Italo Cunha, et al. Towards a Rigorous Methodology for Measuring Adoption of RPKI Route Validation and Filtering. *ACM SIGCOMM Computer Communication Review*, 48(1):19–27, 2018.

[410] Philipp Richter, Mark Allman, Randy Bush, and Vern Paxson. A primer on ipv4 scarcity. *ACM SIGCOMM Computer Communication Review*, 45(2):21–31, 2015.

[411] Philipp Richter, Oliver Gasser, and Arthur Berger. Illuminating large-scale ipv6 scanning in the internet. In *Proceedings of the 22nd ACM Internet Measurement Conference*, pages 410–418, 2022.

[412] Philipp Richter, Georgios Smaragdakis, Anja Feldmann, et al. Peering at peerings: On the role of ixp route servers. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 31–44, 2014.

[413] Philipp Richter, Florian Wohlfart, Narseo Vallina-Rodriguez, et al. A multi-perspective analysis of carrier-grade nat deployment. In *Proceedings of the 2016 Internet Measurement Conference*, pages 215–229, 2016.

[414] NLNOG RING. Participants. Available at `https://ring.nlnog.net/participants/`, 2022. last-accessed: Sunday, 8th May 2022.

[415] RIPE. stats. Available at `https://ftp.ripe.net/pub/stats/ripencc/` Last accessed: May 31st, 2020.

[416] RIPE. Inter-rir transfers. https://www.ripe.net/manage-ips-and-asns/resource-transfers-and-mergers/inter-rir-transfers, 2015. last-accessed: June 3rd, 2020.

[417] RIPE. 5.1 Allocations made by the RIPE NCC to LIRs, 2019. Available at https://www.ripe.net/publications/docs/ripe-733#51 Last accessed: April 14th, 2020.

[418] RIPE. The ripe ncc has run out of ipv4 addresses. https://www.ripe.net/publications/news/about-ripe-ncc-and-ripe/the-ripe-ncc-has-run-out-of-ipv4-addresses, 2019. last-accessed: June 3rd, 2020.

[419] RIPE. The RIPE NCC has run out of IPv4 Addresses, 2019. Available at https://www.ripe.net/publications/news/about-ripe-ncc-and-ripe/the-ripe-ncc-has-run-out-of-ipv4-addresses Last accessed: June 1st, 2020.

[420] RIPE. https://www.ripe.net/manage-ips-and-asns/ipv4/ipv4-pool, 2020. last-accessed: June 3rd, 2020.

[421] RIPE. Ipv4 transfer statistics. https://www.ripe.net/manage-ips-and-asns/resource-transfers-and-mergers/transfer-statistics/within-ripe-ncc-service-region/ipv4-transfer-statistics, 2020. Last accessed: June 25th, 2020.

[422] RIPE. Recognised ipv4 transfer brokers. https://www.ripe.net/manage-ips-and-asns/resource-transfers-and-mergers/brokers, 2020. Last accessed: June 29th, 2020.

[423] RIPE. Ripe ncc charging scheme 2020. https://www.ripe.net/publications/docs/ripe-722, 2020. last-accessed: 03/29/2020.

[424] RIPE. ripe.db.inetnum. https://ftp.ripe.net/ripe/dbase/split/, 2020. Last accessed: June 28th, 2020.

[425] RIPE. Meetings and Events. Available at https://www.ripe.net/participate/meetings, 2021. last-accessed: Tuesday, 20th April 2021.

[426] RIPE. Ripe atlas measurement dumps - [2021/05/17-23], 2021.

[427] RIPE. Routing Information service (RIS). Available at https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris, 2022. last-accessed: Monday, 25th April 2022.

[428] RIPE. Ipv4 waiting list. https://www.ripe.net/manage-ips-and-asns/ipv4/ipv4-waiting-list, 2023. last-accessed: January 3rd, 2020.

[429] RIPE. Reducing ixp ipv4 assignment default size to a /26. https://www.ripe.net/participate/policies/proposals/2023-01, 2023.

[430] RIPE NCC. Ripe ncc begins to allocate ipv4 address space from the last /8. https://www.ripe.net/publications/news/announcements/ripe-ncc-begins-to-allocate-ipv4-address-space-from-the-last-8, 2012. last-accessed: June 3rd, 2020.

[431] RIPE NCC. Extended delegations, 20180405. https://ftp.apnic.net/stats/ripe-ncc/2018/delegatedâĂŤripencc-extended-20180405.bz2, 2021. Last accessed: 24th April, 2021.

[432] RIPE NCC. RIPE RPKI Snapshots, May 2021.

[433] RIPE NCC. RIPE Atlas measurement platform. https://atlas.ripe.net/, 2022.

[434] Nils Rodday, Lukas Kaltenbach, Italo Cunha, et al. On the deployment of default routes in inter-domain routing. In *Proceedings of the ACM SIGCOMM 2021 Workshop on Technologies, Applications, and Uses of a Responsible Internet*, pages 14–20, 2021.

[435] Matthew Roughan, Simon Jonathan Tuke, and Olaf Maennel. Bigfoot, sasquatch, the yeti and other missing links: what we don't know about the as graph. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, pages 325–330, 2008.

[436] Matthew Roughan, Walter Willinger, Olaf Maennel, et al. 10 lessons from 10 years of measuring and modeling the internet's autonomous systems. *IEEE Journal on Selected Areas in Communications*, 29(9):1810–1821, 2011.

[437] Routing-Bits blog. Low memory handling. https://routing-bits.com/2011/08/21/low-memory-handling/. last accessed: March 1, 2022.

[438] Erik Rye, Robert Beverly, and Kimberly C Claffy. Follow the scent: defeating ipv6 prefix rotation privacy. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 739–752, 2021.

[439] Erik C Rye and Robert Beverly. Discovering the ipv6 network periphery. In *International Conference on Passive and Active Network Measurement*, pages 3–18. Springer, 2020.

[440] Dimitris Sacharidis, Kyriakos Mouratidis, and Dimitris Kleftogiannis. A common approach for consumer and provider fairness in recommendations. In *Proc. ACM RecSys (Late-breaking Results,)*, 2019.

[441] Said Jawad Saidi, Oliver Gasser, and Georgios Smaragdakis. One bad apple can spoil your ipv6 privacy. *ACM SIGCOMM Computer Communication Review*, 52, June 2022.

[442] Adam Satariano and Scott Reinhard. How russia took over ukraine's internet in occupied territories. https://www.nytimes.com/interactive/2022/08/09/technology/ukraine-internet-russia-censorship.html, 2022.

[443] Lauren S Schlesselman. Perspective from a teaching and learning center during emergency remote teaching. *American Journal of Pharmaceutical Education*, 84(8), 2020.

[444] Brandon Schlinker, Todd Arnold, Italo Cunha, and Ethan Katz-Bassett. Peering: Virtualizing bgp at the edge for research. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, pages 51–67, 2019.

[445] Brandon Schlinker, Todd Arnold, Italo Cunha, and Ethan Katz-Bassett. PEERING: Virtualizing BGP at the Edge for Research. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, Orlando, FL, December 2019.

[446] Brandon Schlinker, Ítalo Cunha, Yi-Ching Chiu, et al. Internet performance from facebook's edge. In *Proceedings of the Internet Measurement Conference*, pages 179–194, 2019.

[447] Brandon Schlinker, Hyojeong Kim, Timothy Cui, et al. Engineering egress with edge fabric: Steering oceans of content to the world. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, pages 418–431, 2017.

[448] Brandon Schlinker, Kyriakos Zarifis, Italo Cunha, et al. Peering: An as for us. In *Proceedings of the 13th ACM Workshop on Hot Topics in Networks*, pages 1–7, 2014.

[449] Kyle Schomp, Onkar Bhardwaj, Eymen Kurdoglu, et al. Akamai dns: Providing authoritative answers to the world's queries. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pages 465–478, 2020.

[450] Max Schuchard, Christopher Thompson, Nicholas Hopper, and Yongdae Kim. Peer pressure: Exerting malicious influence on routers at a distance. In *2013 IEEE 33rd International Conference on Distributed Computing Systems*, pages 571–580. IEEE, 2013.

[451] Security Boulevard. Notorious 'Hijack Factory' Shunned from Web. `https://securityboulevard.com/2018/07/notorious-hijack-factory-shunned-from-web/`. Last access: October 2022.

[452] Khwaja Zubair Sediqi, Lars Prehn, and Oliver Gasser. Hyper-specific prefixes: gotta enjoy the little things in interdomain routing. *ACM SIGCOMM Computer Communication Review*, 52(2):20–34, 2022.

[453] Semaphore. 95th percentile bandwidth metering explained and analyzed. Available at `https://www.semaphore.com/95th-percentile-bandwidth-metering-explained-and-analyzed/` Last accessed: June 1st, 2020.

[454] Pavlos Sermpezis and Vasileios Kotronis. Inferring catchment in internet routing. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(2):1–31, 2019.

[455] Pavlos Sermpezis, Vasileios Kotronis, Konstantinos Arakadakis, and Athena Vakali. Estimating the impact of bgp prefix hijacking. In *2021 IFIP Networking Conference (IFIP Networking)*, pages 1–10. IEEE, 2021.

[456] Pavlos Sermpezis, Vasileios Kotronis, Petros Gigis, et al. Artemis: Neutralizing bgp hijacking within a minute. *IEEE/ACM Transactions on Networking (TON)*, 26(6):2471–2486, 2018.

[457] M Zubair Shafiq, Lusheng Ji, Alex X Liu, and Jia Wang. Characterizing and modeling internet traffic dynamics of cellular devices. *ACM SIGMETRICS Performance Evaluation Review*, 39(1):265–276, 2011.

[458] Anant Shah, Romain Fontugne, Emile Aben, et al. Disco: Fast, good, and cheap outage detection. In *2017 Network Traffic Measurement and Analysis Conference (TMA)*, pages 1–9. IEEE, 2017.

[459] Benjamin Shantz. Determining ownership and control of ipv4 addresses. *Wash. UL Rev.*, 94:739, 2016.

[460] Xingang Shi, Yang Xiang, Zhiliang Wang, et al. Detecting Prefix Hijackings in the Internet with Argus. In *Proceedings of the 2012 Internet Measurement Conference*, pages 15–28, 2012.

[461] Dmitry Shin, Reginaldo Giovane Guaitanele, and Bretton Vine. Hsdn php looking glass. `https://github.com/hsdn/lg`, 2021. last accessed: 21st. June. 2021.

[462] Georgos Siganos and Michalis Faloutsos. Analyzing bgp policies: Methodology and tool. In *IEEE INFOCOM 2004*, volume 3, pages 1640–1651. IEEE, 2004.

[463] Jared M Smith and Max Schuchard. Routing around congestion: Defeating ddos attacks and adverse network conditions via reactive bgp routing. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 599–617. IEEE, 2018.

[464] P. Smith and R. Evans. RIPE-532 - RIPE Routing Working Group Recommendations on IPv6 Route Aggregation, November 2011.

[465] P. Smith, R. Evans, and M. Hughes. RIPE-399 - RIPE Routing Working Group Recommendations on Route Aggregation, December 2006.

[466] Job Snijders. Rpki's 2022 year in review — growth and innovation. `https://blog.apnic.net/2023/01/18/rpkis-2022-year-in-review-growth-and-innovation/`. Last access: October 2022.

[467] Job Snijders. Peeringdb accuracy: Is blind faith reasonable? `https://archive.nanog.org/sites/default/files/wed.general.peeringdb.accuracy.snijders.14.pdf`, 2013. last accessed: 24th. June. 2021.

[468] Job Snijders. Calgary internet exchange (yycix) deploys world's first aspa-filtering route servers. `https://seclists.org/nanog/2023/Feb/6`, 2023.

[469] Internet Society. Routing security for policymakers. `https://www.internetsociety.org/resources/doc/2018/routing-security-for-policymakers/`, 2018.

[470] Internet Society. State of ipv6 deployment 2018. `https://www.internetsociety.org/resources/2018/state-of-ipv6-deployment-2018/`, 2018. Last accessed: June 29th, 2020.

[471] Hostio Solutions. Get your clean ipv4 and ipv6 ranges from hostio solutions. `https://hostio.solutions/ip-rental/`, 2020. last-accessed: June 3rd, 2020.

[472] SPAMHAUS. How do security researchers use whois data? `https://www.spamhaus.org/whitepapers/dnsbl_function/`, 2020. Last accessed: June 29th, 2020.

[473] SPAMHAUS. How has gdpr affected spam? `https://www.spamhaus.org/news/article/775/how-has-gdpr-affected-spam`, 2020. Last accessed: June 29th, 2020.

[474] SPAMHAUS. The spamhaus don't route or peer lists. `https://www.spamhaus.org/drop/`, 2022. Last accessed: March 29th, 2022.

[475] K. Sriram, D. Montgomery, D. McPherson, et al. Problem Definition and Classification of BGP Route Leaks. RFC 7908, RFC Editor, June 2016.

[476] Ross Stapleton-Gray and William Woodcock. National internet defense—small states on the skirmish line. *Communications of the ACM*, 54(3):50–55, 2011.

[477] Harald Steck. Calibrated recommendations. In *Proc. ACM RecSys*, 2018.

[478] Florian Streibelt, Franziska Lichtblau, Robert Beverly, et al. Bgp communities: Even more worms in the routing can. In *Proceedings of the Internet Measurement Conference 2018*, pages 279–292, 2018.

[479] Tom Strickx. Technical Debt: an Anycast Story, 2018. Available at `https://ripe77.ripe.net/archives/video/2222/` Last accessed: May 30th, 2020.

[480] Stephen Strowes. Visibility of IPv4 and IPv6 Prefix Lengths in 2019. `https://labs.ripe.net/author/stephen_strowes/visibility-of-ipv4-and-ipv6-prefix-lengths-in-2019/`, 2019. Last access: October 2022.

[481] Stephen Strowes and Colin Petrie. BGP Even-More Specifics in 2017. `https://labs.ripe.net/Members/stephen_strowes/bgp-even-more-specifics-in-2017`, 2017.

[482] subnets.ru. Looking glass list. `http://subnets.ru/wrapper.php?p=1`, 2021. last accessed: 21st. June. 2021.

[483] Lakshminarayanan Subramanian, Sharad Agarwal, Jennifer Rexford, and Randy H Katz. Characterizing the internet hierarchy from multiple vantage points. In *Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 618–627. IEEE, 2002.

[484] Yixin Sun, Anne Edmundson, Nick Feamster, et al. Counter-raptor: Safeguarding tor against active routing attacks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 977–992. IEEE, 2017.

[485] Louis Swinnen, Sébastien Tandel, S Uhlig, et al. An Evaluation of BGP-based Traffic Engineering Techniques. Technical report, INFONET, October 2002.

[486] Team Cymru. The Bogon Reference. `https://team-cymru.com/community-services/bogon-reference/`, 2020.

[487] Cecilia Testart, Philipp Richter, Alistair King, et al. Profiling bgp serial hijackers: capturing persistent misbehavior in the global routing table. In *Proceedings of the Internet Measurement Conference*, pages 420–434, 2019.

[488] Cecilia Testart, Philipp Richter, Alistair King, et al. To Filter or not to Filter: Measuring the Benefits of Registering in the RPKI Today. In *International Conference on Passive and Active Network Measurement*, pages 71–87. Springer, 2020.

[489] The Broadband Forum. TR-069: CPE WAN Management Protocol, March 2018.

[490] The New York Times. Hackers hit dozens of countries exploiting stolen n.s.a. tool. `https://www.nytimes.com/2017/05/12/world/europe/uk-national-health-service-cyberattack.html`, 2017.

[491] Andree Toonk. Aws and their billions in ipv4 addresses. `https://toonk.io/aws-and-their-billions-in-ipv4-addresses/`, 2020. Last accessed: October 24rd, 2020.

[492] Martino Trevisan, Danilo Giordano, Idilio Drago, et al. Five years at the edge: Watching internet from the isp network. *IEEE/ACM Transactions on Networking*, 28(2):561–574, 2020.

[493] CAIDA UCSD. The ipv4 routed /24 topology dataset - [2021/05/17-23], 2021.

[494] CAIDA UCSD. The ipv6 routed /48 topology dataset - [2021/05/17-23], 2021.

[495] Leo Van Audenhove, Julia Pohle, Milton Mueller, et al. Dimensioning the elephant: An empirical analysis of the ipv4 number market. *info*, 2013.

[496] Verizon. Seeing the World with RIPE Atlas. `https://labs.ripe.net/Members/verizon_digital/seeing-the-world-with-ripe-atlas`, 2017.

[497] Kevin Vermeulen, Ege Gurmericliler, Ítalo Cunha, et al. Internet scale reverse traceroute. In *Proceedings of the 22nd ACM Internet Measurement Conference*, pages 694–715, 2022.

[498] C. Villamizar, R. Chandra, and R. Govindan. Bgp route flap damping. RFC 2439, RFC Editor, November 1998.

[499] Virtua.Cloud. Our network. `https://web.archive.org/web/20220130154537/https://www.virtua.cloud/our-infrastructure/our-network`, 2022.

[500] Vultr. Announce your ip space. `https://www.vultr.com/features/bgp/`. Last access: October 2022.

[501] W3Techs. Usage statistics of ipv6 for websites. `https://w3techs.com/technologies/details/ce-ipv6`. last accessed: 19th Sept. 2021.

[502] Daniel Wagner, Daniel Kopp, Matthias Wichtlhuber, et al. United we stand: Collaborative detection and mitigation of amplification ddos attacks at scale. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 970–987, 2021.

[503] Hui Wang, Rocky KC Chang, Dah Ming Chiu, and John CS Lui. Characterizing the Performance and Stability Issues of the AS Path Prepending Method: Taxonomy, Measurement Study and Analysis. In *Proceedings of ACM SIGCOMM Asia Workshop*. Citeseer, 2005.

[504] WhatIsMyIPAddress. Regional internet registries, 2023. website: `https://whatismyipaddress.com/rir`, imagelink: `https://whatismyipaddress.com/wp-content/uploads/rir-map-small.png`.

[505] Wikipedia. List of Internet exchange points. Available at `https://en.wikipedia.org/wiki/List_of_Internet_exchange_points` Last accessed: June 1st, 2020.

[506] Wikipedia. 2016 bitfinex hack. `https://en.wikipedia.org/wiki/2016_Bitfinex_hack`, 2022.

[507] Wikipedia. List of tier 1 networks, 2022. Available at `https://en.wikipedia.org/wiki/Tier_1_network` Last accessed: March 30th, 2022.

[508] Wikipedia. National research and education network. Available at https://en.wikipedia.org/wiki/National_research_and_education_network, 2022. last-accessed: Sunday, 8th May 2022.

[509] Wikipedia. Internet service provider. https://en.wikipedia.org/wiki/Internet_service_provider, 2023.

[510] Walter Willinger, David Alderson, and John C Doyle. Mathematics and the internet: A source of enormous confusion and great potential. *Notices of the American Mathematical Society*, 56(5):586–599, 2009.

[511] Philipp Winter, Ramakrishna Padmanabhan, Alistair King, and Alberto Dainotti. Geo-locating bgp prefixes. In *In 2019 Network Traffic Measurement and Analysis Conference (TMA)*, 2019.

[512] Florian Wohlfart, Nikolaos Chatzis, Caglar Dabanoglu, et al. Leveraging interconnections for performance: the serving infrastructure of a large cdn. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, pages 206–220, 2018.

[513] World Health Organization. Coronavirus disease (COVID-19) advice for the public. Available at https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public Last accessed: June 1st, 2020.

[514] World Health Organization. Situation report - 132, 2020. Available at https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200531-covid-19-sitrep-132.pdf?sfvrsn=d9c2eaef_2 Last accessed: June 1st, 2020.

[515] Xantaro. Juniper networks ptx10001 tested as a peering / edge router. https://www.xantaro.net/en/tech-blogs/juniper-ptx10001-test-peering-edge-router/, 2022. Last access: October 2022.

[516] Jianhong Xia and Lixin Gao. On the evaluation of as relationship inferences [internet reachability/traffic flow applications]. In *IEEE Global Telecommunications Conference, 2004. GLOBECOM'04.*, volume 3, pages 1373–1377. IEEE, 2004.

[517] Yan Yang, Xia Yin, Xingang Shi, et al. Inter-domain routing bottlenecks and their aggravation. *Computer Networks*, 162:106839, 2019.

[518] Kok-Kiong Yap, Murtaza Motiwala, Jeremy Rahe, et al. Taking the Edge off with Espresso: Scale, Reliability and Programmability for Global Internet Peering. In *Proceedings of the 2017 Conference of the ACM Special Interest Group on Data Communication*, pages 432–445. ACM, 2017.

[519] Bahador Yeganeh, Ramakrishnan Durairajan, Reza Rejaie, and Walter Willinger. How cloud traffic goes hiding: A study of amazon's peering fabric. In *Proceedings of the Internet Measurement Conference*, pages 202–216, 2019.

[520] YuchenYin. asrank.pl. Available at https://github.com/YuchenJin/ProbLink/blob/master/asrank.pl, 2022. last-accessed: Monday, 25th April 2022.

[521] Beichuan Zhang, Raymond Liu, Daniel Massey, and Lixia Zhang. Collecting the internet as-level topology. *ACM SIGCOMM Computer Communication Review*, 35(1):53–61, 2005.

[522] Xin Zhang, Hsu-Chun Hsiao, Geoffrey Hasker, et al. Scion: Scalability, control, and isolation on next-generation networks. In *2011 IEEE Symposium on Security and Privacy*, pages 212–227. IEEE, 2011.

[523] Ying Zhang. Method and System for Effective BGP AS-Path Pre-pending, May 2013. US Patent App. 13/300,372.

[524] Ying Zhang and Makan Pourzandi. Studying Impacts of Prefix Interception Attack by Exploring BGP AS-PATH Prepending. In *2012 IEEE 32nd International Conference on Distributed Computing Systems*, pages 667–677. IEEE, 2012.

[525] Ying Zhang and Mallik Tatipamula. Characterization and Design of Effective BGP AS-PATH Prepending. In *2011 19th IEEE International Conference on Network Protocols*, pages 59–68. IEEE, 2011.

[526] Zheng Zhang, Ying Zhang, Y Charlie Hu, and Z Morley Mao. Practical defenses against bgp prefix hijacking. In *Proceedings of the 2007 ACM CoNEXT conference*, pages 1–12, 2007.

[527] Ran Zhuo, Shane Greenstein, Bradley Huffaker, and KC Claffy. Gdpr and internet interconnection. `https://cepr.org/voxeu/columns/gdpr-and-internet-interconnection`, 2020.

[528] Maya Ziv, Liz Izhikevich, Kimberly Ruth, et al. Asdb: a system for classifying owners of autonomous systems. In *Proceedings of the 21st ACM Internet Measurement Conference*, pages 703–719, 2021.

[529] Earl Zmijewski. Longer is not always better, 2009. Available at `https://dyn.com/blog/longer-is-not-better/` Last accessed: June 1st, 2020.

[530] Earl Zmijewski. Reckless Driving on the Internet, 2009. Available at `https://dyn.com/blog/the-flap-heard-around-the-world/` Last accessed: June 1st, 2020.