On Mapping the Interconnections in Today's Internet

Reza Motamedi, University of Oregon, motamedi@cs.uoregon.edu
Bahador Yeganeh, University of Oregon, byeganeh@cs.uoregon.edu
Balakrishnan Chandrasekaran, Max-Planck-Institüt für Informatik, balac@mpi-inf.mpg.de
Reza Rejaie, University of Oregon, reza@cs.uoregon.edu
Bruce M. Maggs, Duke University/Akamai Technologies, bmm@cs.duke.edu
Walter Willinger, NIKSUN, Inc., wwillinger@niksun.com

***Abstract*—Internet interconnections are the means by which networks exchange traffic between one another. These interconnections are typically established in facilities that have known geographic locations, and are owned and operated by so-called colocation and interconnection services providers (e.g., Equinix, CoreSite, and EdgeConneX). These previously under-studied colocation facilities and the critical role they play in solving the notoriously difficult problem of obtaining a comprehensive view of the structure and evolution of the interconnections in today's Internet are the focus of this paper.**

We present `mi`^{2}, a new approach for mapping Internet interconnections inside a given colocation facility. [1] We infer the existence of interconnections from localized traceroutes and use the Belief Propagation algorithm on a specially defined Markov Random Field graphical model to geolocate them to a target colocation facility. We evaluate `mi`^{2} by applying it initially to a small set of US-based colocation facilities. In the process, we compare our results against those obtained by two recently developed related techniques and discuss observed discrepancies that derive from how the different techniques determine the ownership of border routers. As part of our validation approach, we also identify drastic changes in today's Internet interconnection ecosystem (e.g., new infrastructures in the form of "cloud exchanges" that offer new types of interconnections called "virtual private interconnections"), and discuss their wide-ranging implications for obtaining an accurate and comprehensive map of the Internet's interconnection fabric.

***Index Terms*—Internet Topology, Colocation Facility, Interconnection Services, Geography**

I. Introduction

Interconnections are the "glue" that ensures that the Internet can function as a network of networks or autonomous systems (AS). Interconnections refer to the physical connectivity between border routers of different networks that allow these networks to connect with each other and exchange traffic. Two networks, for instance, might establish a "private" interconnection (or *private peering*) by connecting two of their border routers, one from each network, through a *dedicated* physical link (or *cross-connect*, also known as *private network interconnection* or *PNI*). To facilitate such private peerings, a *colocation facility (or datacenter) and interconnection services provider* operates colocation facilities wherein networks rent space to deploy their routers. The facility[2] provider may then sell cross-connects to allow any two of its customers to privately exchange traffic. Alternatively, networks may also establish *public peerings* by deploying border routers in a colocation facility that either houses an *Internet exchange point (IXP)* or is part of a geographically distributed IXP and connecting them (for a price) to ports on a switch managed by the IXP. In contrast to cross-connects, network traffic in public peerings is exchanged over a *shared* switching fabric. The notion of "mapping" these interconnections—both private and public—refers to first *inferring* their existence and type, and then *geolocating* (or *pinning*) them to the target colocation facility.

Identifying the number, types and locations of interconnections between pairs of ASes is a critical first step in understanding the structure and evolving nature of the Internet's connectivity fabric. For one, important changes in the way networks interconnect with one another in today's Internet can, in general, be best observed and identified in individual colocation facilities where these interconnections are established and utilized. One recent example of such an observation has been the emergence of new switching infrastructure (termed "cloud exchanges") that a number of the large colo providers (e.g., Equinix and CoreSite) started to operate in some of their main metro locations [1], [2]. Moreover, visible signs of this new Internet infrastructure are new interconnection service offerings in the form of so-called "virtual private interconnections" (VPIs) [2], AWS Direct Connect [3], and Google Cloud Interconnect [4]. VPIs enrich the existing options for networks to interconnect with one another and are specially designed to help enterprises reap the benefits of the growing number of offered cloud services. They also make the task of determining how a given network connects to the rest of the Internet more difficult as they enable networks to deploy "hybrid" connectivity options; that is, restrict cloud-related traffic to VPIs, utilize PNIs for other "value-added" traffic, and send all other traffic through an IXP via public peerings.

Importantly, the emergence of such hybrid connectivity significantly complicates the study of inter-domain networking problems such as critical Internet infrastructure protection [5] Internet service failures [6], routing issues [7], colo-centric

[1] An open-source prototype of `mi`^{2} is available at our project website located at https://onrg.gitlab.io/projects/mii

[2] In this paper, we use the terms *colocation facility*, *facility*, and *colo* interchangeably.

selection of traffic relays for network overlays [8], detecting peering infrastructure outages [9], or Internet inter-domain congestion problems (e.g., see [10] and references therein).

Systematically mapping the interconnections in today's Internet is, however, a well-known difficult problem, and despite a flurry of different efforts reported in the recent literature on this topic (e.g., see [11], [12], [13], [14]), a complete (or comprehensive) solution to this mapping problem has eluded researchers to date. Among the many challenges that have remained largely unsolved and require new ideas are *(i)* the design of reliable techniques for inferring existing interconnections as well as mapping or pinning them to the colocation facility where they are utilized, *(ii)* the development of methods that are not network-specific but are applicable across the wide spectrum of networks that make up today's Internet, *(iii)* a basic understanding of what constitutes a "good" collection of traceroute data for mapping a targeted set of interconnections (e.g., between two ASes, or in a given city or colo), and (iv) a viable approach for validation for this problem space that is notorious for its dearth of ground truth.

In this paper, we present `mi`^{2}, a new methodology and tool for systematically mapping (both inferring and pinning) all (private and public) interconnections in a target colo. `mi`^{2} relies on the information derived from carefully designed traceroute-based measurement campaigns to first infer the likely interconnections at a facility, and then pin them to the inside or outside of that facility. `mi`^{2} is a bottom-up technique: It maps the Internet's interconnections one colo at a time. Such an approach to mapping interconnections has a number of practical benefits. For one, it clearly defines the required probing campaign, and it limits the scope of all possible interconnections to the networks native to the target facility. Moreover, the problem of pinning a discovered interconnection reduces to one of mapping it to the inside or outside of that facility. Finally, by enabling the application of our technique to any of the thousands of colos around the world, `mi`^{2} is well-suited to discover major changes in the Internet's interconnection fabric in the very locations where these interconnections are offered and established. For the purpose of this paper, we illustrate `mi`^{2}'s capabilities by applying it to three CoreSite colos in Los Angeles, Chicago, and Miami, respectively, and evaluate and discuss its accuracy and coverage in comparison to two existing related techniques [13], [15] that consider relevant sub-problems of our overall mapping problem.

In the process of designing, implementing, and evaluating `mi`^{2}, we contribute to the existing literature on mapping interconnections in the following four ways.

• ***Describing a reliable method for inferring interconnections.*** We begin with an in-depth analysis of the router-level view derived from purposefully designed traceroute campaigns (i.e., collection of different traceroutes), and propose a combination of well-established and new heuristics for identifying the owner AS of individual routers to reliably infer all observed interconnections in a given campaign. In the process, we also account for the inaccuracies and ambiguities inherent in counting interconnections from traceroute-based measurements.

• ***Leveraging a probabilistic approach for pinning inferred interconnections that limits the impact of erroneous geolocation information.*** The main reason for utilizing a probabilistic method for pinning inferred interconnections to the inside (or outside) of a given colo facility is to avoid the propagation of any potential geolocation errors during the pinning process. To this end, we first identify a set of observed interfaces, called *anchors*, whose locations (i.e., either inside or outside the facility) can be reliably determined. Leveraging domain knowledge, we establish a few heuristics to assess the likelihood that two (or more) observed interfaces in our campaigns are "co-present" in the same facility. We then leverage a purposefully constructed Markov Random Field (MRF) graphical model that encodes the "co-presence" relationship information among different sets of interfaces. The resulting graphical model along with the locations of anchors are used as input to a probabilistic inference technique known as Belief Propagation. This technique determines the probability that individual interfaces can be pinned to the inside (or outside) of the target facility.

• ***Demonstrating the benefits of extensive validation efforts for a problem that is notorious for a dearth of ground truth.*** We report on our efforts to validate the output of `mi`^{2} despite a scarcity of ground truth that stems from a general unwillingness of the colo operators to share interconnectivity details at their facilities. We use a number of publicly available and diverse sources of information to perform "spot checks", which reveal that our results are very accurate. With respect to evaluating `mi`^{2}'s coverage (i.e., its ability to infer and pin all interconnections at a target facility), we combed through previously ignored sources of information. In particular, we looked at reports that publicly traded colo companies such as CoreSite are required to file with the SEC in the US to inform their shareholders and financial analysts.[3]

• ***Advancing a community-wide effort to support reproducible research in the area of mapping today's interconnections.*** An attractive feature of our open-source `mi`^{2} prototype is that it can be readily enhanced by third-party researchers. For example, in the case of the described probabilistic pinning approach, if better co-presence relationship information becomes available, either by means of new heuristics or an improved understanding of existing heuristics, it can be readily leveraged to improve an existing encoding of co-presence relationships among different sets of interfaces. Furthermore, `mi`^{2} enables, by being open-source, reproducible networking research and encourages comparisons with alternative tools as a way to advance our understanding of mapping the Internet's interconnections. Given the problem space's scarcity of ground truth, it is mainly through direct comparisons with other applicable tools that it will be possible to uncover the elusive ground truth.

A tool like `mi`^{2} might also help a service provider such as a CDN to decide in which colo to install its servers and from which in-colo networks to buy transit from. For example, the CDN can estimate in advance how much traffic it expects to serve to the clients of each of the networks present in the colo.

[3]For a prior use of similar data (i.e., U.S. SEC Form 10-K filings) in the context of mapping ASes to their organizations, see [16].

Since no single network may host more than a small fraction of the clients, it may not be feasible or cost effective for the CDN to peer directly with all of the networks at the colo. Instead, the CDN can use mi^2 to identify a smaller set of networks to peer with at the colo, with the identified networks containing the bulk of the clients. Note that the selected networks may or may not be the same networks that host the largest number of clients. Because these mi^2-inferred "one-hop-away" peerings occur between networks within the same colo, the CDN can expect that the performance will be comparable with using direct peering with these networks.

Roadmap. We review related work in Section II and focus, in particular, on the key differences between our approach and previously considered techniques. While Section III provides an overview of our approach, Section IV and V describe our interconnections inference and pinning techniques, respectively. Our evaluation and validation efforts are presented in Sections VI and VII, respectively. Section VIII concludes with a discussion of remaining challenges.

II. Related Work

There has been a large body of research on the Internet's AS-level topology, using a graph-theoretic model where nodes are ASes and edges between nodes are logical entities indicating that the involved ASes interconnect with one another in one or more locations around the world (see [17], [18], [19] and references therein). The task, however, of systematically mapping the physical realizations of AS interconnections to specific locations has remained a challenging problem. The reasons are partly the scale, heterogeneity, and distributed nature of the Internet [20], [21], partly the absence of adequate mapping tools and techniques [22], and partly the significant difficulties that detailed validation efforts face in view of a general lack of publicly available ground truth data [23], [24].

Recent work has addressed only certain aspects of the overall problem. One such aspect concerns the Internet's colocation facilities themselves [8]. Another aspect is how individual networks, especially the large content providers or CDNs connect to the rest of the Internet [25], [26], [27]. Yet another aspect deals with the special case of mapping IXP-related public interconnections and has motivated past efforts such as [28], [29], [30], [31], [11], [32]. However, due to their IXP-specific nature, these methods and related tools such as *traIXroute* [33] are not applicable to infer and exhaustively enumerate the *private* interconnections at a colo.

A number of recent studies [12], [13], [14] have focused exclusively on the task of inferring interconnections from traceroute data where the key challenge is to reliably determine the owner AS for the IP addresses at each hop of a traceroute and identify the relevant interconnection (i.e., inter-AS IP-level segment). In addressing this challenge, one of these studies (i.e., *bdrmap* [12]) recognizes the unreliability of earlier IP-to-AS mapping techniques [34], [35], [36], [37], [38] and leverages various IP alias resolution methods [39], [40], [41], [42], [43], [44], [45] to design a tool for inferring all interconnections between a given AS and its neighboring ASes. *bdrmap* relies on carefully crafted, targeted traceroute measurements—launched from various vantage points inside the AS towards its neighbors—and employs a set of structured heuristics to identify network boundaries at the router-level (i.e., border routers). Another of these studies [13] presents *MAP-IT*, a tool that attempts to exhaustively enumerate, from a given set of traceroute measurements, exact interface addresses on both sides of any interconnection (together with the specific pair of ASes involved) that is traversed by any of the traceroute probes. The latest effort on this inference problem is described in [14], where the authors synthesize *bdrmap* and *MAP-IT* to design *bdrmap-IT*, a new tool that infers both the AS owner of all routers and all the inter-domain links in a given traceroute dataset.

Neither of the studies [12], [13], [14] have been designed for the purpose of mapping interconnections in the sense of mi^2, e.g., they do not address the problem of pinning. As tools for inferring interconnections from traceroute data, however, they have been evaluated and shown to perform well in certain settings. Nevertheless, real-world complications (e.g., unresponsive routers and lack of proper vantage points) will necessitate more substantial evaluation efforts to provide the sort of completeness, accuracy, and correctness properties that are required before the tools' results can be trusted and used in practice. For example, while *bdrmap* offers an attractive method for identifying an AS's border routers, its broader applicability remains questionable without further information about the extent of the required traceroute campaign and the quality of *bdrmap*'s output as a function of the number and locations of available vantage points inside the target AS. Specifically, to apply *bdrmap* to infer all interconnections at a given colo (i.e., the inference-only part of mi^2's inference and pinning task) would require one or more vantage points in each of the colo's tenant ASes. This is clearly an unreasonable requirement, especially because a majority of colo tenants are typically small networks and provide no vantage points for launching the required traceroutes. Similarly, although *MAP-IT*'s ability to infer interconnections in an arbitrary set of traceroute measurements is appealing, a feature of real-world traceroutes is that some are more useful with respect to inferring interconnections than others. Thus, without a means to assign any level of confidence to its output (i.e., inferred interconnections) as a function of certain properties of the input (i.e., requirements on the considered set of traceroutes), *MAP-IT*'s largely data-driven approach is of limited practical value. Also, since the initial version of *MAP-IT* does not leverage a router-level view of the observed interfaces (as, for example, mi^2 and *bdrmap* do), the tool's output suffers from readily observable inconsistencies (see, for example, Section VII).

Prior work has also relied on commonly used methods—reverse DNS lookup, IP geolocation, and delay-based techniques—for geolocating or pinning inferred interconnections to specific colos (e.g., [46], [47]). The inherent limitations of these methods, however, are well-known, especially when used for geolocating infrastructure-related entities such as interface IP addresses assigned to router ports [48], [49], [50], [45], [51]. A recent study [15] casts pinning as a *constrained facility search (CFS)* problem and leverages various data

sources (e.g., publicly available information about the tenant ASes at different colos, opportunistic traceroute measurements, and targeted traceroutes) to ultimately create enough constraints to pin an inferred interconnection to a single facility. Although *CFS* [15] significantly outperforms heuristics based on DNS naming schemes or IP geolocation, in practice, the approach suffers from an inability to judge the quality of its input (i.e., set of inferred interconnections). As a result, *CFS* is prone to errors: either working with inferred interconnections that are incorrect or pinning connections to facilities that they are not actually located in. Besides, CFS's approach of exploiting opportunistic traceroute measurements is not a reliable recipe for mapping *all* the interconnections at a target facility.

There are also instances where colo providers establish fiber-optic connections ("tethers") between their datacenters to give customers in their newer datacenter(s), where there is typically plenty of space, the option to interconnect with existing customers located in the older datacenter(s), where space has become scarce [15]. Since "tethering" is typically invisible to traceroute, mapping interconnections in such settings simply means identifying the campus or "logical" facility of tethered locations where the given datacenter provider operates the colos and offers interconnection services. An interconnection option known as *remote peering* [52] can similarly be viewed as a type of "tethering". Here, remote peering refers to peering without a physical presence at the IXP and is typically realized by a third-party such as an IXP reseller that operates a Layer-2 infrastructure to connect the remote peer's router to one or more ports at the IXP's switch. While inferring the details of remote peering, especially the differentiation between remote and local peers, is an interesting problem in its own right [53], as far as our mapping effort described in this paper is concerned, our highly localized traceroute campaigns targeting our three CoreSite locations minimize the likelihood of encountering remote peers and wrongly mapping the remote router to target facility that houses the IXP. We leave the detailed treatment of remote peering as outlined in [53] for future work.

In terms of the use of the Belief Propagation (BP) algorithm as our technique-of-choice for performing inference on data with probabilistic interdependencies [54] (i.e., pinning inferred interconnections), we are only aware of some isolated applications of BP to Internet measurement. One such example is for adaptive diagnosis in distributed systems (e.g., see [55]). Some other problems where this technique has been applied include fraud detection [56], fake reviews [57], [58], and collective classification of web pages [59].

Finally, our work provides a complementary view of Durairajan et al.'s effort [60] on mapping the US long-haul fiber-optic infrastructure. In particular, by zooming in on the nodes (i.e., cities) of that map and focusing on the colocation facilities in those cities where the different long-haul optic-fiber routes terminate or originate, our effort aims to reveal the intra-facility connectivity fabric responsible for "handing over" traffic from one tenant to another. As such, while our work does not attempt to connect the US long-haul fiber-optic connectivity fabric with the connectivity fabrics in the thousands of colocation facilities across the US, it does contribute a key piece to the *routing puzzle*: determining the physical routes over which a service provider's traffic flows in the Internet (within the US). Knowing these physical routes helps shed light on a number of inter-domain networking problems such as protecting critical Internet infrastructure, debugging routing problems, detecting Internet service failures, and reducing Internet inter-domain congestion. It also benefits previous efforts on mapping the Internet's infrastructure and its "material geography" [61], [62], [63] and on studying the role of public policy in critical Internet infrastructure protection [5].

TABLE I: Information about the target CoreSite facilities.

Facility	Address	CS	PDB	ASNs	VPs	Target IPs
LAX	One Willshire/ 900 N Alameda St	290	217	444	142	3637
CHI	427 S La Salle St	46	13	44	47	315
MIA	2115 NW 22nd St	27	10	27	34	188

III. Our Approach in a Nutshell

Our proposed methodology `mi`^{2} maps the interconnections inside a target colocation facility. The per-colo casting of the mapping problem naturally motivates purposefully localized traceroute probes for the target colo. These probes are more likely to cross different interfaces of border routers at the target facility, and thus promise to offer a more complete and accurate view of the router-level topology at that facility. Our methodology for mapping interconnections at a colo consists of the following four main steps.

1) Selecting target facilities. For the purpose of this paper and to illustrate the applicability of `mi`^{2} to different environments, we select three of the eight different US-based locations—Los Angeles, Chicago, and Miami—where the colocation and interconnection solution provider CoreSite owns and operates datacenters. With Los Angeles being its largest market, CoreSite operates two colos in LA that are connected or "tethered" to form a large-sized single virtual facility or campus (LAX).[4] In addition to hosting the largest number of tenants among all CoreSite facilities, the LAX facility is also where CoreSite operates the largest IXP on the West Coast of the US. In addition, as we became aware during our later validation efforts (see Section VII for details), starting in 2013, this facility also houses CoreSite's Open Cloud Exchange where networks and enterprises can establish virtual private interconnections to connect directly to their favorite cloud providers. We view the combination of the number of tenants in a colo and the different types of interconnection options offered at a colo as an adequate indicator of the role a colo plays in the geographic area it serves and is located in. In this sense, when compared to the LAX facility, the CoreSite Chicago (CHI) and CoreSite Miami (MIA) facilities with some 30+ and 15+ tenants in a single building, respectively, represent medium- and small-sized colos that also operate no IXP and/or cloud exchange and offer only standard interconnections in the form of cross connections. Table I (columns 1-5) summarizes the basic information about the three selected target facilities at the time when we performed our mapping exercise (i.e., early

[4]We leave the problem of accurately mapping interconnections in an IXP that spans multiple geo-dispersed facilities [53] for future work.

2016). Columns "CS" and "PDB" list the number of tenants as provided by CoreSite [64] and PeeringDB [65], respectively. The column "ASNs" shows how many AS numbers belong to the tenants provided by CS or PDB.[5]

2) Performing localized measurements. Given a target facility, the next step of our approach consists of performing traceroute measurements in a "localized" manner. To this end, we use carefully designed traceroutes where both the vantage points (VPs), for launching the traceroutes, and the traceroute targets are chosen so as to increase the likelihood that the resulting traceroute probes will traverse the interconnections that are utilized by the different tenants in this target facility. Running such a colocation-specific traceroute campaign requires *(i)* obtaining publicly available information about the target facility, e.g., list of tenants, and co-located IXP(s), *(ii)* selecting appropriate VPs, and *(iii)* properly identifying traceroute targets. Table I indicates, in the last two columns, the number of VPs and target IP addresses used for mapping the interconnections at the different colos.

Vantage point selection. In terms of VP selection, we leverage both automated *looking glasses (LGs)* from traceroute repositories (e.g., [65], [66]) and RIPE Atlas probes [67]. For LGs, we prefer those residing within a tenant AS to those that are at the shortest AS-hop distance within a tenant AS's customer cone, and among these qualified LGs, we select those that are located within the city of the target facility or, in case there are none, those that are geographically closest to the target facility. For RIPE Atlas probes, after applying the same shortest AS-hop criteria as for LGs, we only considered qualified RIPE Atlas probes that are within a 100-mile radius of the target facility. The first two rows of Table II provide the details on the LG and RIPE Atlas vantage points, i.e., the number of unique VP IP addresses and the corresponding counts of unique ASes that were used for the different measurement campaigns (MIA, CHI, and LAX).

Target selection. With respect to traceroute targets, we select *local IP addresses* (i.e., IPs of tenant ASes that are derived from a large pool of geolocated IP addresses collected from major P2P applications and are, on average, some 75 miles from the target facility) and *local web servers* (i.e., servers hosted by tenant ASes, located in the same city as the target facility and geolocated within a 50-mile radius of the target facility). To geolocate IP addresses, we used three different databases—MaxMind [68], IP2location [69], and EdgeScape [70]—and employed majority voting to decide on the final answer. If, for some tenant in a target facility, neither of these selections yields an IP, we identify all the /24 prefixes advertised by this tenant, geolocate the first IP in each prefix, and select all IP addresses that are within a 50-mile radius of our target facility as traceroute destinations for such an *unseen tenant*. The last three rows of Table II present the breakdown of target IP addresses and their ASes, for each of the three campaigns, by the different selection methods.

3) Inferring interconnections. With these localized traceroutes from a single campaign, the third step tackles the problem of inferring the interconnections that are traversed by these traceroute probes. Our goal here is to identify border routers rather than border interfaces. This strategy allows us to corroborate different pieces of information to ensure robustness to potential error in individual pieces. The main challenge is to accurately and reliably identify the pairs of border routers associated with each traversed interconnection and determine the owner ASes of those routers. To address this challenge, we develop a set of heuristics that exploit the colocation-centric nature of the available traceroutes. They are designed to produce a router-level view of all interfaces encountered by these traceroutes that is self-consistent (in terms of assigning interfaces to routers and routers to ASes and identifying border routers) and self-evident (in terms of being supported by multiple pieces of evidence). This approach allows us to cope with unreliable IP-to-AS mapping results as well as other inaccuracies such as *address sharing*, i.e., one AS loaning an IP address to another so that both ends of a link have addresses from the same prefix.

[5]Some effort is required to map the listed tenants to the corresponding ASes and identify the associated ASNs.

TABLE II: Characteristics of vantage points' and destination IP addresses' selection for the different measurement campaigns.

	MIA		CHI		LAX	
	IP	AS	IP	AS	IP	AS
LG	24	21	23	22	95	77
RIPE	10	6	24	16	47	21
Local IP	24	19	23	29	95	179
Local Web	86	15	272	17	1,049	64
Unseen AS	78	1	20	8	2,493	68

4) Pinning interconnections. Given such a set of inferred interconnections, `mi`^{2}'s final step consists of geolocating or "pinning" them to the inside or outside of the target facility. The main challenge here is to deal with incomplete or partially incorrect information about the location of some of the observed interfaces. To this end, we formulate the pinning task as a statistical inference problem for a specially defined *Markov Random Field (MRF) graphical model*, and we use the *Belief Propagation (BP)* algorithm to solve it. The benefits of this approach to pinning interconnections are that the MRF model can robustly cope with the inherent "noise" in traceroute-based inferences and BP expresses the pinning results as "beliefs" (i.e., inferred interconnections are mapped to the inside or outside of the target facility with certain probabilities). We defer the discussion of these steps to Sections IV and V.

IV. Inferring Interconnections

A. Problem Formulation

A commonly used approach for inferring interconnections between pairs of ASes from traceroute data involves mapping of the IP address at each hop of a traceroute to its corresponding AS, and searching the resulting AS-augmented view of traceroutes for adjacent IP hops with different ASes. A change in ASN of adjacent hops presents an inter-AS IP segment which indicates the presence of an interconnection. In practice, however, accurately identifying inter-AS IP segments from traceroutes and properly counting the corresponding unique interconnections are challenging problems. We use the intuitive

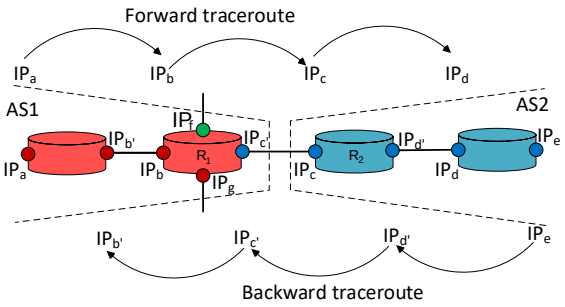


Fig. 1: An example router-level topology depicting an inter-AS IP-level segment (i.e., interconnection).

notion of *near-side* and *far-side* IP addresses for an inter-AS segment that is detected in a traceroute to indicate the order in which these two hops are observed in that traceroute.

Among the reasons for why it is inherently difficult to reliably determine an inter-AS segment from a traceroute are the error-prone nature of all existing IP-to-AS mapping techniques [71] and the practice of subnet sharing between the two interfaces (i.e., using a $/30$ or a $/31$ for addressing) on either side of an interconnection. Figure 1 illustrates the problem using a simple linear topology with four routers. The router interfaces in the figure are colored based on the (owner) AS allocating the corresponding IP address. The interconnection (link $IP_{c'} - IP_c$) is between the two border routers (R_1 and R_2) where R_1 is owned by $AS1$ and R_2 is owned by $AS2$. Per Figure 1, the interfaces on the two ends of this interconnection are allocated by AS2. Figure 1 also shows a (forward) traceroute from AS1 to AS2 that traverses the hops $IP_a \rightarrow IP_b \rightarrow IP_c \rightarrow IP_d$ and correctly identifies the adjacent hops $IP_b - IP_c$ as the inter-AS IP segment. The (backward) traceroute from AS2 to AS1 that traverses this same segment in the opposite direction, however, incorrectly identifies $IP_{c'} - IP_{b'}$ as the inter-AS IP segment for the same interconnection.

Another challenge is posed by routers that do not respond using the ingress interface but instead use a default interface (e.g., IP_f in Figure 1). For example, assuming that the left border router in Figure 1 responds to the forward traceroute with the default interface IP_f that is not even along the path and is mapped to another AS (say $AS3$), then the result would be an incorrectly inferred inter-AS segment (between $AS1$ and $AS3$) along this path. Finally, there is also the problem of determining whether different inter-AS IP segments are associated with the same physical interconnection. For instance, the two inter-AS IP segments $IP_b - IP_c$ and $IP_g - IP_c$ in Figure 1 are both associated with the same physical link ($IP_{c'} - IP_c$). This example shows how the commonly used method of simply counting all the inter-AS IP segments *inflates* the actual number of interconnections and an alternative method is needed to address this issue.

To address these challenges associated with identifying the correct IP-level segment with the physical AS-level interconnection, we advance the existing literature on IP-to-AS mapping by exploiting the localized nature of our traceroute measurements. In particular, our strategy of executing highly localized probing campaigns, unlike general-purpose or opportunistically launched traceroutes, can be expected to reveal multiple interfaces of the different tenant ASes' routers (i.e., *alias sets*, where an *alias set* is defined as a subset of interfaces of a router) inside or in close vicinity of these target facilities. Our key intuition is that by leveraging such an alias-set-based aggregate view produced by our traceroutes, inferring the owner ASes of encountered routers can be performed more accurately compared to relying on isolated interfaces to determine border routers and, hence, identifying the interconnections between them will be less error prone.

Building on this intuition, our methodology for inferring interconnections comprises the following steps: *(i)* identifying alias sets of individual routers, *(ii)* determining the owner AS of each identified router (including border routers), and *(iii)* properly accounting for the interconnections between identified border routers. In the remainder of this section, we elucidate the different techniques used in the above steps.

B. Identifying Individual Routers

To obtain the aggregate view of the interface addresses encountered in our set of localized measurements, we rely on the alias resolution technique of MIDAR [45] (referred to as the *Alias* heuristic). Commonly used alias resolution techniques such as MIDAR, however, are known to result in *false negatives* when routers are unresponsive to probe requests, do not use monotonic IP ID counters[6], or do not share an IP ID counter across interfaces. In fact, any IP-ID-based alias resolution technique is unable to identify an alias associated with such routers [41, p. 3]. Inferring these alias sets[7] missed by MIDAR requires further efforts. Here, instead of exploring the feasibility of more generic alternative IP alias resolution techniques such as Palmtree [72], we rely on two new hand-crafted heuristics that are motivated by our settings and aim at creating a consistent alignment of the inter-AS IP segments encountered in the different localized traceroutes.

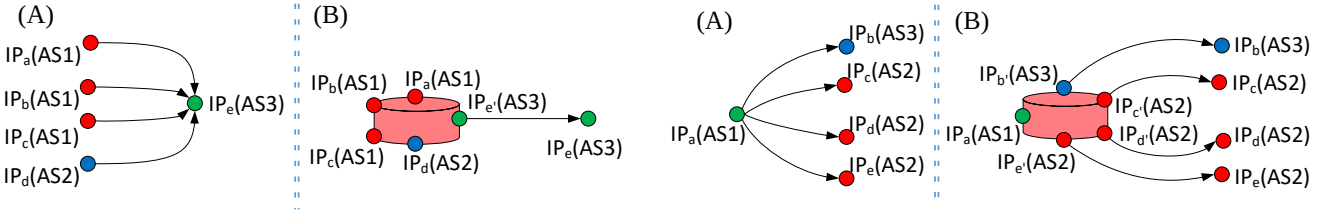
Fan In & Fan Out heuristics. To explain these two new heuristics, consider a collection of inter-AS IP segments inferred from different traceroutes that all share either the second address of a segment, a *Fan-in* scenario as shown in Figure 2I, or the first address of a segment, a *Fan-out* scenario as in Figure 2II. The color of a router or interface, in Figure 2, indicates the corresponding owner AS.[8] For both the Fan-in and Fan-out cases, assuming that the underlying traceroutes do not encounter layer-2 switches along the way and that routers respond with their incoming interface IP[9], Figure 2 also depicts the only plausible router-level topology that is consistent with all the observed inter-AS IP segments. More specifically, in the Fan-in scenario, the first hops of all inter-AS IP segments, e.g., IP_a through IP_d in Figure 2I, must form an alias set. In the case of Fan-out, Figure 2II, for each of the four second hops, there must be an IP address from the same subnet (thus

[6]The IP identifier (IP ID) is a 16 (32) bits field in the IPv4 (v6) header used for aiding reassembly of fragmented packets. Many TCP/IP stacks use a simple counter to set the value of the IP ID field.

[7]Since alias set is a subset of interfaces belonging to the same router, different *alias sets* refer to different routers.

[8]Figure 2 indicates owner ASes only for illustration; knowledge of owner ASes is *not required* for determining alias sets.

[9]In most countries, the fraction of routers that respond with their incoming interface is above 50%. The fraction is even higher in the U.S. [73].



(I) Fan-in (II) Fan-out

Fig. 2: **(I)** *Fan-in*, and **(II)** *Fan-out* structures illustrating *(A)* inter-AS IP segments from traceroutes, and *(B)* the only plausible router-level topology that is consistent with the observations.

owned by the same AS) that is a member of an alias set with the first-hop IP address.

C. Determining Owner AS of Routers

To determine the AS owner of each inferred router (or alias set), we first apply a commonly used IP-to-AS mapping heuristic [74] to assign the default (i.e., BGP-based) AS owner to each encountered interface in our campaign. A key element of `mi`^{2} is to change this default AS owner assignment for each interface for which it has convincing evidence that a change is in order. We next discuss several heuristics that `mi`^{2} applies (in the presented order) to determine an interface's AS owner, either by leveraging information in the form of alias sets that we identified using the Alias, Fan-in, or Fan-out techniques, or by relying on domain-specific heuristics that can, where applicable, link interfaces with their corresponding routers in a more direct manner.

a. Conservative voting heuristic. First, we employ a more conservative version of the commonly used majority voting heuristic [49], [75]. Instead of a simple majority voting scheme to determine the AS owner of a router, we use a *conservative voting heuristic* and declare a router to be owned by AS X *if* the number of interfaces that have been mapped to AS X is more than *two-times plus one* of that mapped to any other AS. This heuristic ensures a degree of robustness to errors that are prevalent in traditional (i.e., BGP-based) approaches. In particular, when using this conservative voting heuristic, no AS owner is assigned for a router whose alias set is either small or does not result in a clear majority.

b. IXP-assigned IPs heuristic. Next, to ensure that our approach is also capable of inferring public IXP-provided interconnections among ASes, we also include in our toolkit the previously proposed *IXP-assigned IPs heuristic* that was specially designed to infer public interconnections from traceroutes [28], [33]. In short, ASes that are members at an IXP use addresses from the IXP's IP prefix on their router interfaces attached to the IXP switch. To identify the member AS to which an observed IP_{ixp} was assigned, we consider all the next hops of that IP_{ixp} across different traceroutes. We conclude that IP_{ixp} is assigned to (and the corresponding router is owned by) AS X if the next hop IPs across all traceroutes are owned by X as shown in Figure 3.

c. Sink IP heuristic. Our third heuristic is motivated by the observation that many of our localized traceroutes that are destined toward small regional tenant ASes terminate at a specific set of IP addresses, referred to as *sink IPs*, before they reach the destination AS[76]. Moreover, these sink IP addresses are never observed in traceroutes toward other ASes. The strong association between small regional ASes and sink IPs suggests that these IPs reside on a router that is owned by the regional AS (perhaps these IPs reside on the first router on the path to the regional ASes that implements the policy of blocking traceroutes). The validity of the heuristic is apparent for sink IPs with informative DNS names, e.g., the sink IP for AS30188 (Televergence Solutions Inc.) is advertised by AS3257 (GTT), but its DNS name is *televergence-gw.ip4.gtt.net*.

d. Subnet matching heuristic. In instances where an interconnection is traversed by traceroutes in both directions, the *subnet matching* heuristic leverages the use of the same subnet between interfaces on either ends of the interconnection to accurately determine the inter-AS segment (see also [77] that describes a similar idea in the context of a newly proposed IP alias inference method). To illustrate, consider the two traceroutes over the router-level topology shown in Figure 4: $IP_a(AS1) \rightarrow IP_b(AS1) \rightarrow IP_c(AS2)$ and $IP_d(AS2) \rightarrow IP_{b'}(AS1)$, where $IP_x(ASi)$ denotes that IP_x was mapped to ASi. Suppose IP_b and $IP_{b'}$ share a subnet and the (border) routers respond to traceroute probes using the ingress interface's IP [78]. The two traceroutes, hence, must pass through the same subnet, i.e., $IP_b(AS1) - IP_{b'}(AS1)$, in both directions in which case the subnet matching heuristic condition holds. Moreover, since the next hop IPs (from IP_b to IP_c in the forward direction and from IP_d to $IP_{b'}$ in the reverse direction) belong to different ASes, the router on the right must be owned by $AS2$ and therefore the

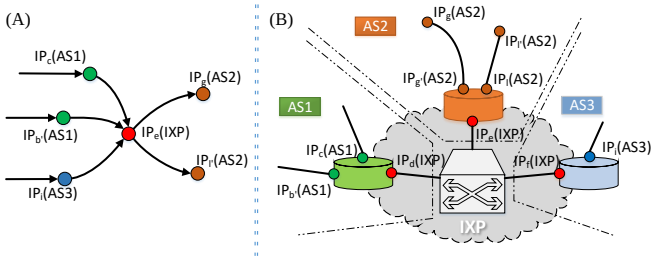


Fig. 3: (A) The traceroute view of traces that hit the IXP, and (B) the inferred physical router-level connections corresponding with the IXP.

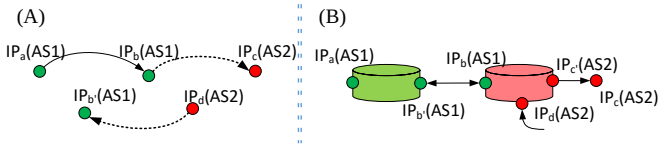


Fig. 4: (A) The "traceroute view" of an interconnection observed from opposite directions, and (B) the corresponding physical router-level topology.

link $IP_b(AS1) - IP_{b'}(AS1)$ is an interconnection between $AS1$ and $AS2$. It also implies that $IP_d(AS2)$ and $IP_b(AS1)$ are part of an alias set, but failed to be identified as such by MIDAR [45]. In practice, we use XNET [79] to identify all IP address pairs that are on the same subnet. Among the IP-segments from $AS1$ to $AS2$ and the reverse direction, we examine whether the far-side of one inter-AS IP segment (e.g., $IP_{b'}(AS1)$) is in the same subnet with the near-side of another IP-segment (e.g., $IP_b(AS1)$) in the reverse direction. If these conditions are satisfied, we re-map the near-side IP (i.e., IP_b) to $AS2$ instead of $AS1$.

e. Valley-free heuristic. If none of the above heuristics determine the AS owner of an identified router, we label the owner as "ambiguous" and apply the *Valley-free* heuristic that leverages control plane information, i.e., inferred AS relationships [80]. Specifically, we consider all traceroutes that pass through any of the interfaces of an ambiguous router and focus on the AS-level view of the three-hop segment—the interface, and the hops before and after that interface. Iterating through the list of potential owner ASes of each interface, we check each time whether the resulting AS-level path segment is *valley free*. Any candidate AS that satisfies this condition in all traceroutes is considered to be a viable owner for this particular ambiguous router. In the case of multiple candidate ASes, we select an owner AS at random.

Apart from the IXP-assigned IPs and Valley-free heuristics, all the aforementioned heuristics are improvements over existing techniques and have been tailored to leverage the localized nature of our traceroute measurements. Note that a successful assignment of an AS owner to a router by any of these heuristics results in all interfaces of the router being mapped to that same AS owner. Thus, for any router, assuming more than one heuristic is found to be applicable, observing consistent outcomes (i.e., owner AS assignment to that router) across the different heuristics will at once increase our confidence in the assignment. Finally, if all of our heuristics fail to identify the AS owner, `mi`^{2} refrains from making any changes and honors the original BGP-based AS ownership assignment for the concerned interfaces.

D. Accurate Interconnection Accounting

A key implication of accurately inferring the owner AS of border routers is that the resulting router-level view avoids overcounting the actual number of interconnections encountered in traceroutes. In particular, our *Fan-in* and *Subnet matching* heuristics offer concrete guidelines for aggregating a group of inter-AS IP segments that are associated with a single interconnection. For example, the inferred physical router-level view of the Fan-in scenario in Figure 2I clearly shows that there is a single interconnection between the two ASes. Therefore, simply counting the number of inter-AS IP segments in different traceroutes as in [15] will typically result in inflating the actual number of interconnections, potentially by significant amounts, depending on the observed traceroute view.

V. Pinning Interconnections

Given a set of inferred interconnections from the previous step, the goal of "pinning" is to determine whether the interfaces on either side of a given interconnection (and hence the interconnection itself) are located inside or outside of our target colo. There are two challenges in this pinning process: *(i)* how to determine whether two directly connected interfaces are colocated in the same facility and *(ii)* how to prevent erroneous information about an interface from propagating to other interfaces and negatively impacting our ability to accurately pin them. To cope with these challenges, we adopt a *probabilistic technique* to pin the location of all interfaces associated with the inferred interconnections. To this end, we first identify a set of "anchor" interfaces for which we have strong evidence that they should be pinned to either the inside or outside of the target facility. Second, using domain knowledge, we establish a few heuristic rules for assessing the *likelihood of co-presence* for pairs of interfaces, i.e., the likelihood that two interfaces are located in the same facility. Leveraging a graphical model in the form of a Markov Random Field (MRF), we capture these likelihoods by encoding them as edge weights between related interfaces. Third, we use majority voting among the output of the three IP2Geo databases (i.e., Edgescape, IP2Location, and Maxmind) to infer the rough location of each observed interface. Then we consider a subgraph of our original graphical model that only contains all nodes (interfaces) that were mapped by the above method to the same metro area where our target colo facility is located, and we use this subgraph as input to the Belief Propagation algorithm. This algorithm implements a probabilistic inference technique that determines the probabilities with which the different interfaces are pinned to the inside (or outside) of the target facility. Since this technique considers for each interface the weighted effect of all other related interfaces, it can effectively deal with "noise" in the input (e.g., erroneous information on anchor or co-located interfaces). Leveraging this subgraph of the original MRF graphical model further reduces noise and computational overhead for running the Belief Propagation algorithm. We next describe these different steps for pinning in more detail.

A. Identifying Anchor Interfaces

Anchor interfaces that can, to the best of our knowledge, be reliably pinned to the inside (or outside) of the target facility are key to bootstrapping the pinning process. The information sources that we utilize to determine such anchor interfaces are *(i)* facility information embedded in DNS names [39], *(ii)* IXP-provided information about colocated IP addresses [29], [81], and *(iii)* the presence of small regional ASes among the facility's tenants.

TABLE III: Counts of IN-anchors and OUT-anchors identified by different techniques for each colo.

	Miami		Chicago		Los Angeles	
	IN	OUT	IN	OUT	IN	OUT
DNS hints	0	21	0	40	3	43
IXP	0	7	3	27	238	64
Small Regional AS	11	0	126	0	336	0

Identification and inclusion of the third information source to determine anchor interfaces is yet another unique trait of our facility-oriented approach to mapping interconnections. This source exploits the presence of small regional tenant ASes in our target facility towards identifying anchor interfaces. More specifically, because small regional tenant ASes typically deal with limited traffic volumes, economic arguments suggest that they establish their interconnections at a single facility in the city or region where their customers reside. It is therefore reasonable to assume that all the observed border IP addresses associated with such small regional tenant ASes are located in one and the same facility. We consider all observed IP addresses (including any LG) associated with such small regional ASes to be located in the same colo. More formally, we first use CAIDA's AS rank data [82] and consider a tenant AS to be a potential small regional AS if it has less than 5 ASes in its customer cone and advertises less than 50 /24-prefixes. We then cross-check any of these potential regional ASes against PeeringDB and only consider those as our regional ASes that according to PeeringDB are present at a single colo in the metro area of our target facility.

Table III presents the number of in/out anchors that we identified using each of these three techniques in each one of the target cities. The number and relative fraction of in/out anchors in each campaign depends on various factors including the number of colos in the target city and the number of tenant ASes in the target colo.

B. Encoding Co-presence Rules

For identifying any instances of co-presence of two or more interfaces observed in a set of traceroutes from our campaign, we devised four co-presence heuristics or rules. These rules are listed below in the order of decreasing confidence that we have in them. The first two rules are applicable to two or more interfaces while the last two are only relevant to two adjacent hops of a traceroute.

1) Alias sets. All interfaces in an alias set belong to the same router and must therefore be pinned to the same facility.

2) Common or Different PoP-tags. Many ASes embed a combination of geo-location code with trailing digits (e.g., `pao1` and `pao3`) to denote their points of presence (PoPs) at different colocation facilities in the same city. We call these alphanumeric strings *PoP-tags*[10]. All interfaces of a tenant AS that are associated with the same PoP-tag must be pinned to the same facility. Alternatively, interfaces (of the same tenant AS) associated with different PoP-tags (i.e., implying different PoPs) should be pinned to different facilities.

3) Inter-domain links. The interfaces of two adjacent hops of a traceroute that belong to different ASes (i.e., inter-AS IP segment) should be pinned to the same facility if their difference in RTT delays is relatively small (e.g., less than a few milliseconds).

4) Intra-domain links. The interfaces of two adjacent hops of a traceroute that belong to the same AS (i.e., intra-AS IP segment) should be pinned to the same facility with a probability that is inversely proportional to their RTT difference.

Regarding the last two rules, we note that the highly-localized nature of our probes coupled with commonly-used hot-potato routing suggests that the (forward and reverse) routes from a local vantage to both ends of an inter-AS interconnection are similar and will therefore result in comparable RTT values.

C. MRF Model Construction

To construct our Markov Random Field (MRF) graphical model, we represent each observed interface in a measurement campaign as a node in a graph, and encode a node's co-presence relationships with other interfaces as edges. Each edge is annotated with a weight that indicates our relative level of confidence in the corresponding co-presence relationship. This encoding entails augmenting the final graph with additional "logical" nodes for the *Alias sets* and *Common PoP-tags* rules. For these rules, if nodes a, b, c, and d are the interfaces of an alias set (or PoP-tag) A, instead of encoding this co-presence relationship as a clique among the four nodes, we add a new logical node A to the graph and encode the alias set (or PoP-tag) relationships as a star-shaped graphlet, containing edges between the center node A and each of the nodes $a - d$. The rationale for substituting densely-connected substructures like cliques with more sparsely-connected graphlets such as stars is to reduce the number of cycles in the resulting graphical model. Reducing cycles is generally recommended when performing certain inference algorithms (e.g., Belief Propagation) on MRF graphical models [54].

Note that to facilitate reproducibility efforts, we provide details about the parameterization of our MRF graphical model and our choice of model parameters in the Supplementary Material (Part A).

D. Probabilistic Inference for Pinning

Belief Propagation (BP) is an algorithm for performing inference on data with probabilistic inter-dependencies [54]. The BP algorithm uses our MRF graphical model as input to infer the posterior state probabilities of all nodes in the generated graph given the observed states for anchor nodes (i.e., interfaces). The algorithm proceeds by iteratively passing messages between nodes based on the previous beliefs and the pairwise joint probabilities. The algorithm updates the state of each node in each iteration until a significant fraction (say 90%) of the nodes reach a steady (i.e., non-oscillating) state. While there is no theoretical guarantee of BP's convergence, in practice BP is known to work well, typically converging quickly to a stable and accurate solution [54], [83].

[10]For identifying PoP-tags, we used simple parsing rules that look for 3-letter segments (matching an airport code) immediately followed by numeric values.

TABLE IV: The breakdown of pinned interconnections by mi² in the LAX campaign based on the pinning status—"hit" (h), "miss" (m), and "close-call (c)—of the interfaces at both ends. Each cell is further divided into "private + public" interconnections.

	h-h	h-c	h-m	c-c	c-m	m-m
MIA	22+0	23+0	8+26	5+0	3+0	161+1
CHI	48+0	7+0	34+0	13+0	28+0	130+0
LAX	1228+1343	109+1	83+25	76+0	28+0	94+77

The BP algorithm emits for each node the probability with which it is pinned to the inside of the target facility. Nodes that reach a steady state are pinned to the inside of the target facility by the algorithm if their associated probability is 0.9 or higher. These nodes are termed *hits*, i.e., they are considered to be inside of the facility. In contrast, nodes that reach a steady state with an associated probability of 0.1 or less are referred to as *misses* and are considered to be outside of the target facility. The rest of the nodes, including the unstable ones, are *close calls* and are *not mapped* by our approach. With this definition of "hits" and "misses" for individual interfaces, to determine which inferred interconnections are located inside our target facility, we simply have to look at either ends of a given interconnection and check whether our algorithm mapped both the interfaces to the inside of our target facility.

An illustration of the insensitivity of the BP-based pinning inference results to our parameterization of the underlying MRF graphical model and empirical evidence in support of the accuracy of our pinning results are provided in the Supplementary Material (Part B).

VI. Results and Comparisons

In an effort to illustrate and evaluate the key elements of mi², we selected three CoreSite facilities—LAX, CHI, and MIA—as targets. Table I presents some basic information about these facilities and the corresponding localized traceroute campaigns that we ran to obtain the input for mi². With our particular selection of traceroute sources (i.e., vantage points) and destinations, we had to launch only a total of 170K, 8K and 2.5K traceroutes for the LAX, CHI, and MIA campaigns, respectively, and we gathered all of these measurements in a single day.

Mapping interfaces to ASes. mi²'s ability to infer interconnections rests largely on its effective use of alias-based aggregate information for mapping interfaces to ASes. In this regard, Table V shows the results of applying our heuristics to the interfaces that we observed in the three measurement campaigns. Each row corresponds to a distinct measurement campaign (e.g., MIA). While the "Total" column provides the counts of unique interfaces observed in all collected traceroutes, the column labelled "AS-Inferred" shows the subset of interfaces for which our heuristics for inferring an AS owner (refer Section IV-C) were applicable. The remaining interfaces (i.e., "Total" – "AS-Inferred") for each campaign (or row) are mostly associated with intra-AS links or simply do not trigger any of our heuristics due to their infrequent appearances in our traceroutes.

The remaining seven columns of Table V (from "Alias" through "Valley Free") show the breakdown of all the inferred interfaces (under "Inferred") across the different heuristics used for inferring their AS owners. The three columns under "Majority Voting" show the technique that was used to infer the association of a group of interfaces that are part of an alias set. Whenever multiple heuristics were applicable to a given interface and inferred the same owner AS, we counted the interface only towards the first applicable heuristic (i.e., left-most column). If both *Fan-in* and *Fan-out*, for instance, identify AS X as the owner of a given interface, we only increment the count for *Fan-in*. Across all three campaigns, as Table V shows, our conservative majority voting heuristic infers the owner AS for roughly 70% of IPs that are associated with an *Alias* set. This percentage increases to over 90% when we add the IPs whose owner AS is inferred by the majority voting heuristic on alias sets that are determined by *Fan-in*, *Fan-out*, and *Sink IP* heuristic. Furthermore, the AS owners for a small fraction ($\leq 5\%$) of interfaces are determined by the *Valley-free* heuristic.

Inferring interconnections. Given all the interfaces from each of the three campaigns for which our heuristics were applicable and inferred a unique AS owner, Table VI summarizes our findings about the resulting inferred interconnections. The first two columns of Table VI list the total number of unique inferred interconnections at the AS-level and IP-level for each campaign, respectively. The third column shows the number of inferred IP-level interconnections after applying our aggregation method for counting interconnections described in Section IV-D. The fourth and fifth columns shows the subset of AS-level and aggregated IP-level interconnections (and their percentage) that are between tenant ASes in the respective target facility. The results in Table VI highlight two important points. *First, the number of inferred IP-level interconnections drops by more than 50% as a result of our aggregation method which illustrates the importance of this aggregation step in preventing the incorrect overcounting of interconnections. Second, some 60%-80% of inferred (aggregate IP-level) interconnections are between known tenants in the different target facilities which demonstrates our success in "localizing" the traceroutes of our measurement campaigns.*

From inferred to pinned interconnections. The inferred (aggregated) IP-level interconnections between tenant ASes in a target facility (i.e., "Agg. IP-level" in Table VI) are the result of running mi²'s inference component on the data produced by the corresponding localized measurement campaigns. Now, we apply mi²'s pinning algorithm to map these inferred interconnections to the inside (or outside) of the corresponding target facility.

For each colocation facility and the associated localized measurements, mi²'s probabilistic pinning algorithm labels each observed IP address as *(i)* a "hit" (h), when the address is pinned to the inside of the facility, or *(ii)* a "miss" (m), when it is pinned to the outside of the facility, or *(iii)* a "close call" (c), when the IP address is not pinned. Consequently, the inferred (aggregated) IP-level interconnections can be subdivided into six groups based on the labels assigned to either

TABLE V: mi²: Results for heuristic-based inference of owner AS for observed interface IPs.

			Majority Voting			Heuristics			
	Total	**AS-Inferred**	**Alias**	**Fan-in**	**Fan-out**	**IXP**	**Sink IP**	**Subnet**	**Valley Free**
MIA	2,569	1,810	78%	6%	3%	1%	8%	0.0%	3%
CHI	2,993	2,038	70%	6%	3%	2%	14%	0.1%	5%
LAX	22,324	16,444	68%	7%	5%	2%	14%	0.3%	5%

TABLE VI: mi²: Interconnections inferred at the AS and IP level in the different campaigns.

	Inferred Interconnections				
	AS-level	**IP-level**	**Agg. IP-level**	**AS-level (b/w tenants)**	**Agg. IP-level (b/w tenants)**
MIA	316	1,634	644	164	396 (62%)
CHI	390	1,541	703	181	397 (56%)
LAX	3,518	20,712	8,039	2,662	6,602 (82%)

ends of each interconnection. Table IV presents the number of interconnections in each one of these six pinned groups of interconnections in the LAX campaign. The interconnections in each cell of Table IV are further divided into "private+public" based on whether an interconnection is established at an IXP (public) or not (private). These results show that while a large percentage of the inferred interconnections in the LAX facility are pinned to that facility, this percentage is lower for other campaigns. The main reason for this discrepancy between the different campaigns is the much larger number of in- and out-anchors that we utilized for the LAX campaign (see Table III). We defer validation of the inferred interconnections to Section VII.

Comparisons with MAP-IT and CFS. Although a number of recent studies such as [12], [13], [14] have dealt with the problem of inferring interconnections from traceroute data, none of them have been designed for the purpose of mapping interconnections in the sense of mi². Nevertheless, to the extent possible and where appropriate, we focused on the CoreSite LAX campus location to perform a comparative study and report below on the main findings of our efforts. In the Supplementary Material (Part C), we present a detailed account of our comparative study between mi² and *MAP-IT* as well as *CFS*.

We first note that a direct comparison of these recent efforts shows a number of design principles that the resulting tools have in common. For one, being either explicitly (*bdrmap*, *CFS*) or implicitly (*MAP-IT*) based on traceroute measurements, their success depends critically on the availability of suitable vantage points for launching the traceroutes, including publicly available Looking Glasses, RIPE Atlas probes, and general-purpose traceroute servers. Furthermore, being traceroute-based, the different tools all depend on some form of IP-to-AS mapping and are therefore restricted to inferring or geo-locating interconnections between entities that own an AS number (ASN) and participate in inter-domain routing.

Next, a direct and fair comparison between mi² and *bdrmap* is unfortunately not feasible because *bdrmap*'s focus on a single network makes the tool not applicable to our colo-centric setting. On the other hand, *MAP-IT* allows for an informative and more direct head-to-head comparison with mi² but clearly shows the shortcomings of a general-purpose tool for inferring interconnections. In particular, we observe that the IP-to-AS mapping produced by *MAP-IT* can be very unreliable and inconsistent because of its limited use of relevant information (e.g., alias sets). Finally, special care is needed when comparing the results of mi² and *CFS*. For one, *CFS* was designed for a different purpose and setting, does not claim to exhaustively map the interconnections in a given colocation facility, and has no publicly available code. Despite these restrictions, we were able to shed light on some of the observed differences. In particular, by simply relying on BGP-inferred AS ownership of routers/interfaces, *CFS* results in incorrect or inaccurate pinning results by virtue of incorrectly inferred interconnections.

VII. Validation Efforts

Colocation facility providers are in general averse to disclosing data on the interconnections established in their facilities. Unsurprisingly, our attempts to obtain such information from the providers of our target colos were futile. The tenants of a colo are also reluctant to share interconnectivity details, unless such details are sufficiently coarse so as to not reveal the type, precise location, and name of the peer of an interconnection (e.g., [75]). This general paucity of reliable ground truth makes the validation of any approach for mapping interconnections (including ours) challenging. In light of these difficulties, we take a more nuanced approach to validation and discuss its implications.

A. On Accuracy of mi²

In lieu of a full-fledged validation of mi², we report on a number of limited validation efforts that use publicly available sources of information or hard-to-come-by control-plane data, or leverage our ongoing collaboration with a large CDN.

IXP-assigned IPs. The IP addresses that IXPs assign to their members (i.e., the IXP-assigned IP-to-member-AS mappings) are publicly available [65] and reveal the AS owners of those IPs. mi² identified the correct owner AS in more than 90% of 600 such IPs.

IPs with informative DNS names. There are often hints embedded in the DNS names (i.e., PTR records of IP addresses) that suggest IP address sharing between two ASes.

An IP address with the DNS name `ae-0.teliasonera.chcgil09.us.bb.gin.ntt.net`, for instance, should be inferred to be owned by AS1299 (i.e., TeliaSonera). For more than 91% of around 400 IP addresses observed in our traceroute campaigns that have a DNS name from which we can infer an AS, mi^2's inference matches the AS owners inferred from the hints.

Ground truth for a tenant AS. We ran `show bgp summary` on a router owned by a large CDN that is also a tenant in the CoreSite LA campus. The BGP summary revealed that the CDN connected with five other tenants in this facility via five different interconnections. mi^2 correctly inferred and pinned these five interconnections to this target facility. This effort leveraged our ongoing relationship with this large CDN.

An opportunistic control-plane constellation. We identified a LG-enabled router belonging to a tenant AS in CoreSite LA, i.e., Hurricane Electric (HE), that was tagged to be located in that facility. Using `show bgp summary` on this LG, we obtained the IP address of the next BGP hop (the peer router, i.e., the neighbor AS) for this tenant and used this information for validation.. This opportunistic effort produced 160 different IP-level interconnections between HE and the other tenants in this facility. mi^2 correctly inferred and pinned 124 of these 160 interconnections. Manual inspection of the missing 36 revealed that mi^2 could not infer them because they served as backup routes; routes through Equinix served as the *preferred* option. We used another LG-enabled router owned by HE, one located in an Equinix facility in LA, to confirm these path preferences.

Note that some of the errors observed in these reported validation efforts could be due to a number of reasons including stale DNS names, incorrect or missing PeeringDB entries, or missing prefix information for the related IXPs. While these "spot checks" provide only limited evidence for the general validity of mi^2, they do show that, within the context of the reliable data we could obtain, our methodology is effective and has a high degree of accuracy.

B. On Completeness of mi^2

One goal of mi^2 is to map all interconnections at a target colocation facility. To ascertain if mi^2 satisfies this goal, we searched for relevant, but rarely utilized information that the operators of these facilities make publicly available. In particular, when combing through various CoreSite-provided online materials, we came across quarterly investor presentations and earnings call transcripts that provide up-to-date and presumably reliable aggregate interconnection-related information on its LAX facility. Specifically, investor presentation [84]'s time frame matches that of our LAX measurement campaign and includes on page 12 the statement **"Our entertainment and gaming ecosystem in Los Angeles is interconnected using more than** 2000 **cross connections [i.e., private interconnections]."** That mi^2 only maps a total of roughly 1300 private interconnections to the LAX facility is clear evidence that mi^2 is, unfortunately, falling short of its goal.

To identify the root cause of this shortcoming, we carefully combed through a wide range of publicly available CoreSite material that concerns its LAX facility. Our examinations revealed that even though the Internet interconnection landscape has recently undergone rapid changes, there has been little or no published work that details some of these changes. These changes can be best seen and analyzed when focusing on individual colos such as the CoreSite LAX facility and can be broadly divided into the following three categories.

A new type of tenant. A closer look at the tenant lists published by colo providers such as CoreSite (e.g., [64], [85]) shows a mix of tenants that consists of a growing number of enterprises (e.g., digital content providers, multimedia, system integrators, managed services, etc.) that typically do not own an AS number (and, hence, do not participate in BGP), and are deployed in a given facility using IP addresses assigned by either an upstream provider or by the colo provider. Moreover, according to published CoreSite statistics [86], since 2013, the number of such enterprise customers deployed in its LAX facility alone has grown by almost 50% per year, from some 50+ in 2013 to over 150 in late 2016. They constitute the fastest growing segment of deployed tenants.

A new type of infrastructure. A few years ago, CoreSite announced the launch of the company's *(Open) Cloud Exchange*, a switching fabric specifically designed to facilitate interconnectivity among networks, cloud providers, and enterprises in ways that provide the scalability and elasticity essential for cloud-based services and applications [2]. Similar platforms have been launched by other major colo and datacenter providers such as Equinix [1] and EdgeConneX [87]. The attraction of a cloud exchange for enterprises is that it facilitates establishing service-to-service interconnectivity among all involved parties (i.e., enterprises, cloud service providers, customers and clients of new services provided by enterprises). Such easy-to-establish interconnectivity typically helps in making new cloud service offerings successful. However, due to the way they are operated, cloud exchanges remain by and large invisible to traditional (i.e., traceroute-based or BGP-based) measurement campaigns.

A new type of interconnection service. To satisfy the increasing demand for this service-to-service interconnectivity, the colo providers that operate cloud exchanges have introduced a new interconnection option called virtual private interconnection (VPI). By purchasing a single port from such a cloud exchange operator, enterprises with or without an AS number can circumvent the public Internet by establishing VPIs to any number of cloud service providers that are present at that cloud exchange. These cloud exchanges also provide a programmable, real-time cloud management portal that supports the varied needs of enterprise customers by enabling them to establish VPIs in a highly-flexible, on-demand, and near real-time manner (e.g., see [88]).

The impact that these observed changes have had on the Internet interconnection landscape has been profound. Not only do they turn a large pool of entities (i.e., enterprises without an ASN) that have so far been largely absent from the interconnection marketplace into active participants and buyers, but they also enrich the existing Internet infrastructure with new entities (i.e., cloud exchanges) that are specifically designed to support VPIs. For instance, with LA being one of CoreSite's biggest markets, the decision of some of the

cloud providers (e.g., AWS and Azure) to deploy in its LAX campus has served as a "magnet" for new enterprise customers looking to establish hybrid or multi-cloud (e.g., public and private clouds) architectures for their businesses or IT needs. In turn, the presence of these major cloud providers in the CoreSite LAX facility has fueled a growth in the number of VPIs at its cloud exchange. To our knowledge, there are two main reasons why none of these increasingly popular VPIs are visible to any existing mapping techniques (including ours). First, all existing techniques for inferring and/or pinning Internet interconnections, due to their traceroute-based nature and their reliance on conventional measurement platforms, lack cloud-centric vantage points (e.g., VMs running in AWS). Second, current mapping techniques are also unable to deal with entities that cannot be identified with AS numbers, and are, in general, incapable of revealing the connectivity at cloud exchanges due to their reliance on Layer-2 connectivity. As a result, future progress on accurately and exhaustively mapping the interconnections in today's Internet will require novel ideas for designing and developing suitable new measurement tools and inference techniques that can cope with these issues (e.g., see [89].

VIII. Summary and Outlook

When we started this work some three years ago, our objective was to devise a methodology for tackling the yet-unsolved problem of systematically mapping the Internet's interconnections, one colocation facility at a time. On the one hand, by applying the developed methodology, `mi`^{2}, to three colos in the continental U.S., performing some necessarily limited evaluations, and comparing, to the extent possible, `mi`^{2}'s results against those obtained by recently proposed related efforts, we have made significant progress towards the stated goal. On the other hand, even though we mapped only a few colos, by focusing on a large colo (i.e., the CoreSite's LAX campus), we have discovered that existing interconnection options are more complicated than previously thought and are evolving rapidly. Indeed, by tapping into previously ignored data sources, our validation efforts have revealed drastic changes in today's Internet interconnection marketplace. In particular, we report on the emergence of new types of players (e.g., enterprises operating without an ASN) utilizing new types of interconnections (e.g., VPI) at newly emerging infrastructures (e.g., cloud exchanges), mention some of the technological and economic drivers responsible for this shifting interconnection landscape, and explain why these findings negatively affect *all currently known* mapping efforts.

Our study thus reaffirms the need for re-examining existing methods that claim to infer and map all types of interconnections that are established and utilized at a given colo that may or may not operate an IXP and/or a cloud exchange. Our observations also emphasize that new methods are needed to track and study the type of "hybrid" connectivity that are in use today at the Internet's edge. This hybrid connectivity describes an emerging strategy whereby one part of an Internet player's traffic bypasses the public Internet (i.e., cloud service-related traffic traversing cloud exchange-provided VPIs), another part is handled by its upstream ISP (i.e., traversing colo-provided private interconnections), and yet another portion of its traffic is exchanged over the colo-owned and colo-operated IXP. As the number of businesses investing in cloud services is expected to continue to increase rapidly, multi-cloud strategies are predicted to become mainstream, and the majority of future workload-related traffic is anticipated to be handled by cloud-enabled colos [90], tracking and studying this hybrid connectivity will significantly shape and define the research efforts of the networking community. Knowing the structure of this hybrid connectivity, for instance, is a prerequisite for studying which types of interconnections will handle the bulk of tomorrow's Internet traffic and how much of that traffic will bypass the public Internet. A better understanding of these and related problems will shed light on the role that traditional players such as Internet transit providers and emerging players such as cloud-centric datacenter providers may play in the future Internet.

IX. Acknowledgment

This material is based upon work supported by the National Science Foundation under Awards CNS-1320977, CNS-1717187 and CNS-1719165. We would like to thank Akamai Technologies as well as the authors of CFS [15] and MAP-IT [13] for generously sharing their tools and datasets. We would also like to thank the ToN reviewers for their constructive feedback on earlier versions of this paper.

References

[1] Equinix, "Cloud Exchange," http://www.equinix.com/services/interconnection-connectivity/cloud-exchange/, 2017.
[2] CoreSite, "THE CORESITE OPEN CLOUD EXCHANGE - One Connection. Countless Cloud Options." https://www.coresite.com/solutions/cloud-services/open-cloud-exchange.
[3] Amazon, "AWS Direct Connect," https://aws.amazon.com/directconnect/.
[4] Google, "Cloud Interconnect," https://cloud.google.com/interconnect/docs/.
[5] S. P. Gorman, "Networks, complexity, and security: The role of public policy in critical infrastructure protection," Ph.D. dissertation, George Mason University, 2004, AAI3123118.
[6] D. Oppenheimer, A. Ganapathi, and D. A. Patterson, "Why do Internet services fail, and what can be done about it?" in *USENIX USITS*, 2003.
[7] M. S. Kang and V. D. Gligor, "Routing bottlenecks in the Internet: Causes, exploits, and countermeasures," in *ACM SIGSAC*, 2014.
[8] V. Kotronis, G. Nomikos, L. Manassakis, D. Mavrommatis, and X. Dimitropoulos, "Shortcuts through colocation facilities," in *ACM IMC*, 2017.
[9] V. Giotsas, C. Dietzel, G. Smaragdakis, A. Feldmann, A. Berger, and E. Aben, "Detecting peering infrastructure outages in the wild," in *ACM SIGCOMM*, 2017.
[10] A. Dhamdhere, D. D. Clark, A. Gamero-Garrido, M. Luckie, R. K. Mok, G. Akiwate, K. Gogia, V. Bajpai, A. C. Snoeren, and K. Claffy, "Inferring persistent interdomain congestion," in *ACM SIGCOMM*, 2018.
[11] V. Giotsas, S. Zhou, M. Luckie *et al.*, "Inferring multilateral peering," in *ACM CoNEXT*, 2013.
[12] M. Luckie, A. Dhamdhere, B. Huffaker, D. Clark *et al.*, "bdrmap: Inference of borders between IP networks," in *ACM IMC*, 2016.
[13] A. Marder and J. M. Smith, "MAP-IT: Multipass accurate passive inferences from traceroute," in *ACM IMC*, 2016.
[14] A. Marder, M. Luckie, A. Dhamdhere, B. Huffaker, J. M. Smith *et al.*, "Pushing the boundaries with bdrmapit: Mapping router ownership at Internet scale," in *ACM IMC*, 2018.
[15] V. Giotsas, G. Smaragdakis, B. Huffaker, M. Luckie, and k. claffy, "Mapping peering interconnections to a facility," in *ACM CoNEXT*, 2015.

[16] X. Cai, J. Heidemann, B. Krishnamurthy, and W. Willinger, "An organization-level view of the Internet and its implications (extended)," *USC/Information Sciences Institute, Tech. Rep. ISI-TR-2009-679*, 2012.

[17] A. Dhamdhere and C. Dovrolis, "Twelve years in the evolution of the Internet ecosystem," *IEEE/ACM Transactions on Networking (ToN)*, 2011.

[18] M. Roughan, W. Willinger, O. Maennel, D. Perouli, and R. Bush, "10 lessons from 10 years of measuring and modeling the Internet's autonomous systems," *Selected Areas in Communications*, 2011.

[19] W. Willinger and M. Roughan, "Internet topology research redux," *SIGCOMM eBook: Recent Advances in Networking*, 2013.

[20] B. Donnet and T. Friedman, "Internet topology discovery: a survey," *IEEE Communications Surveys & Tutorials*, 2007.

[21] H. Haddadi, M. Rio, G. Iannaccone, A. Moore, and R. Mortier, "Network topologies: Inference, modeling, and generation," *IEEE Communications Surveys & Tutorials*, 2008.

[22] R. Motamedi, R. Rejaie, and W. Willinger, "A survey of techniques for Internet topology discovery," *IEEE Communications Surveys & Tutorials*, 2015.

[23] R. Oliveira, W. Willinger, and B. Zhang, "Quantifying the completeness of the observed Internet AS-level structure," UCLA, 080026, Tech. Rep., 2008.

[24] R. Oliveira, D. Pei, W. Willinger, B. Zhang, and L. Zhang, "The (in)completeness of the observed Internet AS-level structure," *IEEE/ACM Transactions on Networking (ToN)*, 2010.

[25] K.-K. Yap, M. Motiwala, J. Rahe, S. Padgett, M. Holliman, G. Baldus, M. Hines, T. Kim, A. Narayanan, A. Jain *et al.*, "Taking the edge off with Espresso: Scale, reliability and programmability for global Internet peering," in *ACM SIGCOMM*, 2017.

[26] B. Schlinker, H. Kim, T. Cui, E. Katz-Bassett, H. V. Madhyastha, I. Cunha, J. Quinn, S. Hasan, P. Lapukhov, and H. Zeng, "Engineering egress with edge fabric: Steering oceans of content to the world," in *ACM SIGCOMM*, 2017.

[27] F. Wohlfart, N. Chatzis, C. Dabanoglu, G. Carle, and W. Willinger, "Leveraging interconnections for performance: the serving infrastructure of a large CDN," in *ACM SIGCOMM*, 2018.

[28] K. Xu, Z. Duan, Z.-L. Zhang, and J. Chandrashekar, "On properties of Internet exchange points and their impact on AS topology and relationship," in *Networking*, 2004.

[29] B. Augustin, B. Krishnamurthy, and W. Willinger, "IXPs: Mapped?" in *ACM IMC*, 2009.

[30] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger, "Anatomy of a large European IXP," in *ACM SIGCOMM*, 2012.

[31] M. A. Sánchez, J. S. Otto, Z. S. Bischof, D. R. Choffnes, F. E. Bustamante, B. Krishnamurthy, and W. Willinger, "Dasu: Pushing experiments to the Internet's edge," in *USENIX NSDI*, 2013.

[32] N. Chatzis, G. Smaragdakis, J. Böttger, T. Krenc, and A. Feldmann, "On the benefits of using a large IXP as an Internet vantage point," in *ACM IMC*, 2013.

[33] G. Nomikos and X. Dimitropoulos, "traixroute: Detecting IXPs in traceroute paths," in *PAM*, 2016.

[34] Z. M. Mao, R. Bush, T. G. Griffin, and M. Roughan, "BGP beacons," in *ACM IMC*, 2003.

[35] Z. M. Mao, D. Johnson, J. Rexford, J. Wang, and R. Katz, "Scalable and accurate identification of AS-level forwarding paths," in *IEEE INFOCOM*, 2004.

[36] K. Chen, D. R. Choffnes, R. Potharaju, Y. Chen, F. E. Bustamante, D. Pei, and Y. Zhao, "Where the sidewalk ends: Extending the Internet AS graph using traceroutes from P2P users," in *ACM CoNEXT*, 2009.

[37] B. Huffaker, A. Dhamdhere, M. Fomenkov *et al.*, "Toward topology dualism: Improving the accuracy of AS annotations for routers," in *PAM*, 2010.

[38] B. Chandrasekaran, G. Smaragdakis, A. Berger, M. Luckie, and K. Ng, "A server-to-server view of the Internet," in *ACM CoNEXT*, 2015.

[39] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with Rocketfuel," *ACM SIGCOMM CCR*, 2002.

[40] M. H. Gunes and K. Sarac, "Analytical IP alias resolution," in *IEEE International Conference on Communications*, 2006.

[41] A. Bender, R. Sherwood, and N. Spring, "Fixing Ally's growing pains with velocity modeling," in *ACM IMC*, 2008.

[42] R. Sherwood, A. Bender, and N. Spring, "Discarte: A disjunctive Internet cartographer," in *ACM SIGCOMM CCR*, 2008.

[43] K. Keys, "Internet-scale IP alias resolution techniques," *ACM SIGCOMM CCR*, 2010.

[44] J. Sherry, E. Katz-Bassett, M. Pimenova, H. V. Madhyastha, T. Anderson, and A. Krishnamurthy, "Resolving IP aliases with prespecified timestamps," in *ACM IMC*, 2010.

[45] K. Keys, Y. Hyun, M. Luckie, and K. Claffy, "Internet-scale IPv4 alias resolution with MIDAR," *IEEE/ACM Transactions on Networking (ToN)*, 2013.

[46] V. Giotsas, M. Luckie, B. Huffaker *et al.*, "Inferring complex AS relationships," in *ACM IMC*, 2014.

[47] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan, "Mapping the expansion of Google's serving infrastructure," in *ACM IMC*, 2013.

[48] M. J. Freedman, M. Vutukuru, N. Feamster, and H. Balakrishnan, "Geographic locality of IP prefixes," in *ACM IMC*, 2005.

[49] M. Zhang, Y. Ruan, V. S. Pai, and J. Rexford, "How DNS misnaming distorts Internet topology mapping," in *USENIX ATC, General Track*, 2006.

[50] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-based geolocation of Internet hosts," *IEEE/ACM Transactions On Networking (ToN)*, 2006.

[51] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye, "IP geolocation databases: Unreliable?" *SIGCOMM CCR*, 2011.

[52] I. Castro, J. C. Cardona, S. Gorinsky, and P. Francois, "Remote peering: More peering without Internet flattening," in *ACM CoNEXT*, 2014.

[53] G. Nomikos, V. Kotronis, P. Sermpezis, P. Gigis, L. Manassakis, C. Dietzel, S. Konstantaras, X. Dimitropoulos, and V. Giotsas, "O peer, where art thou?: Uncovering remote peering interconnections at IXPs," in *ACM IMC*, 2018.

[54] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propagation and its generalizations," *Exploring artificial intelligence in the new millennium*, 2003.

[55] I. Rish, M. Brodie, S. Ma, N. Odintsova, A. Beygelzimer, G. Grabarnik, and K. Hernandez, "Adaptive diagnosis in distributed systems," *IEEE Transactions on Neural Networks (ToN)*, 2005.

[56] S. Pandit, D. H. Chau, S. Wang, and C. Faloutsos, "Netprobe: a fast and scalable system for fraud detection in online auction networks," in *WWW*, 2007.

[57] L. Akoglu, R. Chandy, and C. Faloutsos, "Opinion fraud detection in online reviews by network effects." in *ICWSM*, 2013.

[58] L. Akoglu and C. Faloutsos, "Graph-based spam/fraud detection algorithms and apps," https://www.andrew.cmu.edu/user/lakoglu/icdm12/ICDM12-Tutorial%20-%20PartIII.pdf, 2012.

[59] B. Taskar, P. Abbeel, and D. Koller, "Discriminative probabilistic models for relational data," in *Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2002.

[60] R. Durairajan, J. Sommers, W. Willinger, and P. Barford, "Intertubes: A study of the US long-haul fiber-optic infrastructure," in *ACM SIGCOMM*, 2015.

[61] S. Graham, "Excavating the material geographies of cybercities," *The Cybercities Reader. London: Routledge*, 2004.

[62] E. Malecki, "The economic geography of the Internet's infrastructure," *Economic geography*, 2002.

[63] ——, "11 Internet networks of world cities: agglomeration and dispersion," *International Handbook of Globalization and World Cities*, 2012.

[64] CoreSite, "Carrier List," http://www.coresite.com/resources/carrier-list, 2015.

[65] PeeringDB, "Exchange Points List," https://peeringdb.com/, 2016.

[66] "BGP Looking Glass Database," http://www.bgplookingglass.com/.

[67] "RIPE RIS," https://goo.gl/jFAsIA.

[68] MaxMind-LLC, "GeoIP, 2016," http://www.maxmind.com, 2016.

[69] IP2Location, "IP2Location DB9, 2015," http://www.ip2location.com/.

[70] Akamai, "EdgeScape Service Description," https://www.akamai.com/us/en/multimedia/documents/akamai/akamai-services.pdf.

[71] Z. M. Mao, J. Rexford, J. Wang, and R. H. Katz, "Towards an Accurate AS-Level Traceroute Tool," in *Applications, Technologies, Architectures, and Protocols for Computer Communications*, 2003.

[72] M. E. Tozal and K. Sarac, "Palmtree: An IP alias resolution algorithm with linear probing complexity," *Elsevier Computer Communications*, 2011.

[73] M. Luckie *et al.*, "A second look at detecting third-party addresses in traceroute traces with the IP timestamp option," in *PAM*, 2014.

[74] Team Cymru, "IP to ASN Mapping," https://www.team-cymru.org/IP-ASN-mapping.html, 2016.

[75] N. Feamster, "Revealing Utilization at Internet Interconnection Points," *Available at SSRN 2756888*, 2016.

[76] CAIDA, "AS relationships – with geographic annotations," https://www.caida.org/data/as-relationships-geo/, 2018.

[77] M. Gunes and K. Sarac, "Resolving IP aliases in building traceroute-based Internet maps," *IEEE/ACM Transactions on Networking (ToN)*, 2009.

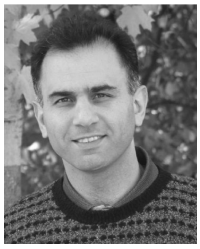
[78] F. Baker, "Requirements for IP version 4 routers," *RFC1812*, 1995.

[79] M. Tozal and K. Sarac, "Tracenet: An Internet topology data collector," in *ACM IMC*, 2010.
[80] CAIDA, "AS Relationships," http://www.caida.org/data/as-relationships/, 2016.
[81] N. Chatzis, G. Smaragdakis, A. Feldmann, and W. Willinger, "There is more to IXPs than meets the eye," *SIGCOMM CCR*, vol. 43, no. 5, Nov 2013.
[82] CAIDA, "AS Rank: AS Ranking," http://as-rank.caida.org/, 2016.
[83] D. H. Chau, S. Pandit, and C. Faloutsos, "Detecting fraudulent personalities in networks of online auctioneers," in *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 2006.
[84] Coresite, "Investor Presentation," http://www.coresite.com/investors/coresite-information/investor-presentations, 2015.
[85] CoreSite, "Additional Carrier List," http://www.coresite.com/resources/resource-library/additional/carrier-list, 2015.
[86] Seeking-Alpha, "CoreSite 3Q 2016 Results - Earnings Call Transcripts," http://seekingalpha.com/article/4016025-coresite-realtys-corceo-paul-szurek-q3-2016-results-earnings-call-transcript, 2016.
[87] EdgeConnex, "Edge Cloud Connect," http://www.edgeconnex.com/services/edge-cloud-connect/, 2017.
[88] Equinix, "Equinix Cloud Exchange," http://www.equinix.com/resources/data-sheets/equinix-cloud-exchange/, 2017.
[89] B. Yeganeh, R. Durairajan, R. Rejaie, and W. Willinger, "How cloud traffic goes hiding: A study of Amazon's peering fabric," in *ACM IMC*, 2019.
[90] Gartner, "Cloud Adoption Trends Favor Public Cloud With a Hybrid Twist," https://www.gartner.com/doc/3401517/report-highlight-market-trends-cloud, 2016.
[91] A. Marder, "MAP-IT," https://github.com/alexmarder/MAP-IT, 2017.

Reza Motamedi is currently a Software Engineer at Twitter Inc, where his interests are cloud infrastructures and warehouse computing. Before joining the technology industry, Reza worked as a graduate research fellow at the University of Oregon. During this time, his reseach interest was networked systems, with a focus on measurement and analysis. Reza's publications cover topics including characterizing user behavior in social networks, uncovering topology of the Internet, and scalable algorithms for analyzing large graphs. Reza received his Ph.D. from the University of Oregon in 2016, and his M.S. from Sharif University of Technology, Tehran, Iran in 2010.

Bahador Yeganeh is currently a PhD student at University of Oregon where he is a graduate research assistant. His research is focused on uncovering and characterizing the emergent peering ecosystem of cloud providers. He received is B.S. degree from Isfahan University of Technology in 2013.

Balakrishnan Chandrasekaran is a Senior Researcher at the Max-Planck-Insitut für Informatik in Saarbrücken, Germany. He received his PhD from Duke University, and his research interests focus broadly on networked systems.



Reza Rejaie is currently a Professor at the University of Oregon. From 1999 to 2002, he was a Senior Technical Staff member at AT&T Labs—Research in Menlo Park, California. He received a NSF CAREER Award for his work on Peer-to-Peer streaming in 2005 and a European Union Marie Curie Fellowship in 2009. Reza has been a visiting professor at IMDEA Networks Institute, the Politecnico di Torino, and Sorbonne University. He received his Ph.D. degree from the University of Southern California in 1999, and his B.S. degree from the Sharif University of Technology in 1991. Reza is a Fellow of IEEE (2017) and a Senior member of the ACM (2006).

Bruce Maggs received the S.B., S.M., and Ph.D. degrees in computer science from the Massachusetts Institute of Technology in 1985, 1986, and 1989, respectively. After spending one year as a Postdoctoral Associate at MIT, he worked as a Research Scientist at NEC Research Institute in Princeton from 1990 to 1993. In 1994, he moved to Carnegie Mellon University, where he stayed until joining Duke University in 2009. While on a two-year leave-of-absence from Carnegie Mellon, Maggs helped to launch Akamai Technologies, serving as its first Vice President for Research and Development. He retains a part-time role at Akamai as Vice President for Research. In 2018 he was part of a large team that received the inaugural SIGCOMM Networking Systems Award for the Akamai CDN, and was named an ACM Fellow.

Walter Willinger is Chief Scientist at NIKSUN, Inc., the world leader in real-time monitoring and cyber forensics solutions. Before joining NIKSUN, he worked at AT&T Labs-Research in Florham Park, NJ from 1996 to 2013 and at Bellcore Applied Research from 1986 to 1996. Dr. Willinger received his Dipl. Math. from the ETH Zurich and his M.S. and Ph.D. in ORIE from Cornell University. He is a Fellow of ACM (2005), Fellow of IEEE (2005), AT&T Fellow (2007), and Fellow of SIAM (2009), co-recipient of the 1995 IEEE Communications Society W.R. Bennett Prize Paper Award and the 1996 IEEE W.R.G. Baker Prize Award, and co-recipient of the 2005 and 2016 ACM/SIGCOMM Test-of-Time Paper Awards. His paper "On the Self-Similar Nature of Ethernet Traffic" is featured in "The Best of the Best - Fifty Years of Communications and Networking Research," a 2007 IEEE Communications Society book compiling the most outstanding papers published in the communications and networking field in the last half century.

APPENDIX A
DETAILS OF THE MRF GRAPHICAL MODEL

Parameterization. In our MRF model, we assign a binary random variable X_v to each node v, i.e., each node v can be in exactly one of two "states" — $X_v = IN$ (1) denotes that v is pinned to the inside of the target facility, and $X_v = OUT$ (0) if it is pinned to the outside, with $P[X_v = IN] = 1 - P[X_v = OUT]$. In particular, the state of each node that represents an inside (outside) anchor interface is set to IN (OUT). A desirable feature of an MRF is its ability to effectively encode whether the likelihood that two connected nodes have similar or opposite states is high or low (i.e., four possible cases). In particular, the fully parameterized model for the joint probabilities for each one of the four possible states for a pair of connected nodes, v_1 and v_2, listed in Table A.1 encodes the essence of two of our co-presence rules. In this model, the probabilities depend nominally on two parameters ϵ and ϕ, with ϕ being the main parameter and ϵ often set to a small value (e.g., $\epsilon = 0.05$). As such, Table A.1(a) explicitly accounts for the *Alias sets* co-presence rule where, for two interfaces v_1 and v_2 in an alias set, high probabilities are assigned to similar states (e.g., $(v_1 = IN) \wedge (v_2 = IN)$) and low probabilities to opposite states (e.g., $(v_1 = IN) \wedge (v_2 = OUT)$). Table A.1(b), in contrast, encodes the *Common/Different PoP-tags* co-presence rule that expects a more differentiated assignment of probabilities (i.e., low probability for pinning two interfaces belonging to different PoP-tags to the inside of a colocation facility, and high probabilities for the other three possible states). Note that the values of the probability fractions for the four possible states of each rule are simply determined as follows: first, the numerator is set to (ϵ+ϕ) and ϵ for states with high and low probability, respectively. Then, the summation of all four numerators is used as the denominator of all fractions so that the sum of all probabilities is equal to one.

TABLE A.1: Joint probabilities for two co-presence rules: (a) *Alias sets*, and (b) *Different PoP-tags*, $\epsilon \ll \phi$

(a) Alias sets

X_{v_1}	X_{v_2}	P
OUT	*OUT*	$\frac{\epsilon+\phi}{4\epsilon+2\phi}$
OUT	*IN*	$\frac{\epsilon}{4\epsilon+2\phi}$
IN	*OUT*	$\frac{\epsilon}{4\epsilon+2\phi}$
IN	*IN*	$\frac{\epsilon+\phi}{4\epsilon+2\phi}$

(b) Different PoP-tags

X_{v_1}	X_{v_2}	P
OUT	*OUT*	$\frac{\epsilon+\phi}{4\epsilon+3\phi}$
OUT	*IN*	$\frac{\epsilon+\phi}{4\epsilon+3\phi}$
IN	*OUT*	$\frac{\epsilon+\phi}{4\epsilon+3\phi}$
IN	*IN*	$\frac{\epsilon}{4\epsilon+3\phi}$

Choice of ϕ. To complete the MRF construction, a large training set is typically used to "learn" the proper value of ϕ. For our problem, however, this approach is not feasible because of a general lack of appropriately labelled training data. Instead, we take a more pragmatic view and argue that *as long as the joint probabilities assigned for the various co-presence rules are aligned with the relative level of confidence we have in them (i.e., in the order indicated above), a probabilistic inference technique properly pins individual interfaces without being too sensitive to the specific value of ϕ.* In fact, being largely insensitive to the choice of the value of ϕ implies that the constructed model is inherently robust and that the model's output is not an artifact of a specific parameter setting. To enforce the desired alignment, we simply consider a linear ordering of rules by setting $\phi = (5 - k)c$ where k denotes the order (or rank) of a co-presence rule, i.e., $k = 1, 2, 3, 4$ for the *Alias sets*, *Common or Different PoP-tags*, *Inter-domain links*, and *Intra-domain links* rules, respectively. The parameter c, hence, simply defines the relative gap between the value of ϕ for consecutive rules.

APPENDIX B
SENSITIVITY AND ACCURACY OF BP-BASED PINNING

Sensitivity results for BP-based pinning. Interpreting quantitative results about `mi`^{2}'s ability to pin inferred interconnections to a given target facility requires a basic understanding of the (in)sensitivity of the BP-based pinning method to the parameterization of the underlying MRF graphical model (see Section V-C). To perform this basic (in)sensitivity analysis, we consider the case of our LAX campaign and show in Figure B.1 the inferred distribution of "beliefs" (i.e., probabilities) for all nodes encountered in this campaign as a function of the parameter c. Figure B.1 illustrates that the probability distributions tend to become more bimodal as we increase c. This behavior implies that the inferred probabilities represent a clear pattern in the data and are not an artifact of our choice of the value of the parameter c.

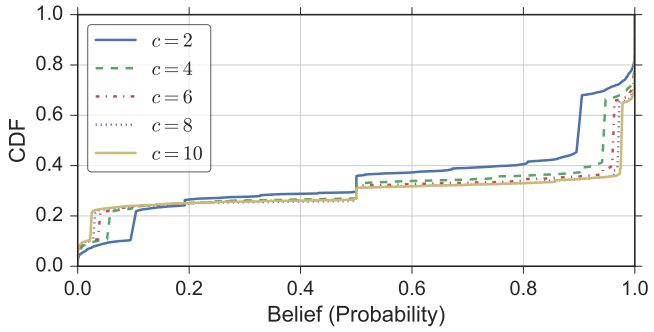


Fig. B.1: The effect of the parameter c on the distribution of beliefs for all nodes in the LAX campaign.

Assessing the accuracy of BP-based pinning. We apply a commonly used test technique to assess the sensitivity of BP's outcome to the choice of the parameter c. In particular, we remove $10\% - 60\%$ of randomly selected anchors for testing, and run the BP algorithm with the remaining anchors. We repeat each test 10 times using different random sets of anchors. As an example, consider the case where the goal is to maximize the number of correctly inferred inside/outside anchors for our measurement in LA using 40% of anchors for testing.

The left-side plots in Figure B.2 (from top to bottom) show the summary distribution of the fraction of test anchors inside the target facility in LA that are mapped as *hit*, *miss* or *close-call*, as a function of c. The right-side plots show the same information for test anchors outside the target facility in LA. These results demonstrate that once the value of c exceeds 3 or 4, more than 90% of inside anchors are correctly mapped and the variations across different runs are very small ($< 2\%$).

The mapping accuracy for outside anchors is around 80% and exhibits a somewhat larger variability. Note that the relatively lower accuracy in mapping outside interfaces is caused by the significantly smaller fraction of outside anchors, which are likely to be located across a geographically diverse set of metro areas. (Note that that our main interest is in confirming that an anchor is inside or outside of our target colo and not in determining exactly where an outside anchor is located. This empirical analysis suggests that accuracy is highest for the BP algorithm for c-values between 4 and 9.

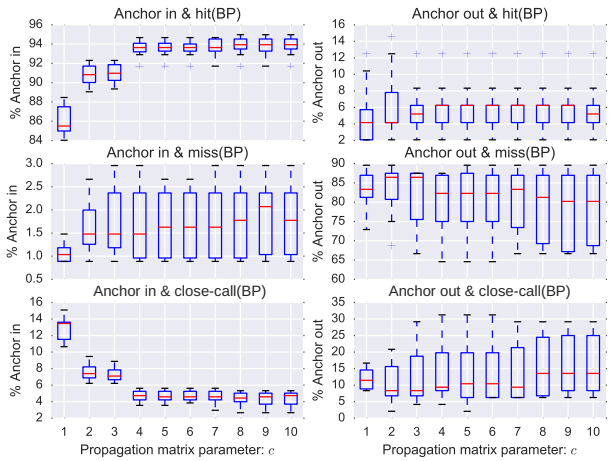


Fig. B.2: The effect of parameter $\boldsymbol{c}$ on the accuracy of BP for pinning anchors that are not used for pinning in the CoreSite-LA campaign.

APPENDIX C
COMPARISON OF mi² WITH OTHER APPROACHES

Comparison with the MAP-IT method. We report here on a head-to-head comparison of mi² with *MAP-IT* [13], a recently developed tool specifically designed for inferring interconnections from a given set of traceroute measurements. For this comparison, we consider only CoreSite's LAX campus and used all the traceroutes from our LAX campaign to derive the IP (or interface) adjacency graph that the *MAP-IT* tool requires as input [91]. Given this input, *MAP-IT* identified a total of 2615 unique IP-level interconnections that further reduced to 2209 (between 1139 pairs of ASes) after aggregation. The 2209 interconnections inferred by *MAP-IT* exclude 31 duplicates that we discovered in *MAP-IT*'s output. Furthermore, aggregation for *MAP-IT*-inferred interconnections refers to the removal of any instances of interconnection that represents an already existing interconnection but in the opposite direction. *Between the* 8093 mi²*-inferred and* 2209 *MAP-IT-inferred (aggregated) IP-level interconnections, we find* 2156 *interconnections that are common, and for* 1565 *(i.e., 73%) of them, both approaches infer exactly the same IP-level segment as the interconnection between the same pair of tenant ASes.*

The most striking difference between the two methods is the large gap in the number of mi²- and *MAP-IT*-inferred (aggregated IP-level) interconnections. To explain this gap, we checked for how many of the applicable interfaces observed in our LAX campaign did these two methods infer different AS owners. We found that for the 22,324 observed interface IPs in the LAX campaign, mi² and *MAP-IT* inferred different AS owners for only 2683 (12%) of them. We refer to this subset of interfaces with inconsistently inferred AS owners as *IIAS interfaces*. The IIAS interfaces require further examination as they are the reason for the large difference observed in inferred interconnections between mi² and *MAP-IT*.

Upon closer examination of these IIAS interfaces, we noticed that the *MAP-IT*-inferred AS owner agrees with the (default) BGP-inferred AS owner for 80% of the IIAS interfaces; the mi²-inferred AS owner, in stark contrast, matches with the BGP-inferred owner for only 18% of the interfaces. For the rest (i.e., 2%), both techniques deviate from the BGP-derived AS ownership. These numbers illustrate that mi² *is changing the default BGP-derived AS owner more often than MAP-IT (i.e., about 4-out-5 times vs. 1-out-of-5 times for the case of these IIAS interfaces)*. More importantly, we next show that mi² only changes the default BGP-derived AS owner of individual IIAS interfaces based on compelling evidence. In contrast, in the infrequent cases when *MAP-IT* changes the BGP-inferred AS owner of an IIAS interface, it sometimes does so for the wrong reason. Next, we detail how and why *MAP-IT* incorrectly changes the BGP-derived AS owner of some interfaces.

We begin by examining the IIAS interfaces that were identified as members of an alias set using our *Alias* heuristic. For the LAX campaign, there are some 600 such sets, and, for each of them, mi² leveraged its conservative majority voting heuristic to infer the unique AS owner for all interfaces in each set. In contrast, when examining the consistency of the *MAP-IT*-inferred AS owner for interfaces in any of these 600 sets, we observed 466 sets (~75%) where the *MAP-IT*-inferred AS ownership is internally inconsistent, i.e., *MAP-IT* declares that different interfaces of the same router belong to different ASes.

Even if MAP-IT were to incorporate alias (router-level) information in conjunction with the traditional or our conservative majority voting heuristic (see Section IV-C), it would remain less trustworthy than mi² *simply because of its limited use of such information.* Recall that in the process of applying the *Subnet matching* heuristic, mi² expands the set of observed interfaces from a given campaign by considering also the interfaces that were discovered by running XNET [79] using /29 expansion (see Section IV-C). For our LAX campaign, this additional effort produces a more *expanded view* of each of the 600 alias sets. This expanded view typically enables mi² to make a more informed decision about changing or not changing the BGP-inferred AS ownership of such expanded alias sets/routers than the *MAP-IT* method with its partial view of those alias sets. Indeed, we encountered several instances among the LAX campaign's 600 alias sets when *MAP-IT* with its partial view incorrectly changed the BGP-inferred AS owner of the grouping's interfaces. In contrast, mi² with its expanded view of these sets coupled with its use of the conservative voting heuristic relied on strong evidence for not changing the BGP-inferred AS owners.

Comparison with the CFS method. As a reference point

for our pinning results, we checked the interconnections that mi^2 pinned to the inside of a facility against those obtained by using the recently proposed Constrained Facility Search (*CFS*) method [15]. Note, however, that *CFS* was designed for a different purpose and setting, it does not claim to exhaustively map the interconnections inside any particular facility, and it has no publicly available code. For these reasons, a direct comparison between the two methods is, unfortunately, impossible. Nevertheless, it is possible to examine the set of interconnections that *CFS* mapped to, for example, CoreSite LAX, one of our target colo facilities, and check how *CFS*'s results compare to mi^2's findings about the interconnections in that colo.

To this end, we obtained from the authors of [15] a set of traceroutes that their *CFS* method relied on to map 317 private IP-level interconnections to the same facility that we targeted with our LAX campaign. Each of these *CFS*-provided interconnections is represented by the near-side IP and its associated AS as well as the far-side AS (note that *CFS* does not provide the far-side IP for an inferred interconnection). Since mi^2 intends to exhaustively map the interconnections inside a given target facility, for all practical purposes (and not accounting for possible churn due to the misaligned time periods of the two studies or for a possible lack of suitably-located VPs), mi^2 should be able to detect all these 317 *CFS*-provided interconnections and pin them to the CoreSite LAX campus.

To examine what mi^2 has to say about these 317 *CFS*-mapped interconnections, we first noticed that these interconnections are associated with 89 unique near-side IP addresses that can be divided into two groups. The first group consists of 43 near-side IP addresses (associated with 167 of the 317 interconnections) that both *CFS* and mi^2 observed. 37 of them (associated with 137 out of the 167 interconnections) are also near-side IP addresses of an mi^2-inferred interconnection. mi^2's pinning algorithm marks 36 of those 37 IP addresses as "hit" and the remaining one as "close call". Moreover, mi^2 agrees with the *CFS*-provided AS owner for all of these 37 near-side IP addresses. The two methods by and large, however, disagree about the far-side AS owner (for 130 out of 137). We note that there is another group of IPs that are mutually visible to both *CFS* and mi^2 but are identified as near-side IPs only by mi^2 since their AS owners are determined based on our heuristics (not BGP data). Considering this group of IP addresses further decreases the alignment of inferred near-side IPs for mutually visible IPs.

The second group consists of the remaining 46 IP addresses that are only observed by *CFS*. Given that these 46 near-side IP addresses were not seen in any of mi^2's traceroutes, we leveraged the *MAP-IT* tool [91] to try and gain further insight. Observe that the *MAP-IT* and *CFS*-provided interconnections, however, are not comparable at the IP level since *CFS* only provides the near-side IP while *MAP-IT* only identifies the far-side IP of an interconnection. The inferred AS owners of the near-side IP addresses of interconnections, however, are comparable since that data is provided by both methods. Interestingly, when we executed *MAP-IT* with the adjacency matrix resulting from the *CFS*-provided traceroutes, we observed that *MAP-IT* and *CFS* agreed on the AS owner for *only one* of these 46 near-side IP addresses. *This empirical finding suggests that, by simply relying on BGP-inferred AS owners, CFS incorrectly infers these near-side IP addresses as part of the interconnections it mapped. In contrast, by leveraging different types of additional interface-related information, MAP-IT- and* mi^2*-based IP-to-AS mapping efforts are capable of changing the default BGP-derived AS ownership of interfaces, and, as discussed in Section VI, they do so occasionally (MAP-IT) or frequently (*mi^2*).* This example highlights the diligence that is *necessary* for accurately inferring and pinning interconnections.