



THE IBY AND ALADAR FLEISCHMAN FACULTY OF ENGINEERING

The Zandman-Slaner Graduate School of Engineering

# **The Internet PoP Level Graph**

**By**

**Noa Zilberman**

THESIS SUBMITTED TO THE SENATE OF TEL-AVIV UNIVERSITY

in partial fulfillment of the requirements for the degree of

"DOCTOR OF PHILOSOPHY"

October, 2013





THE IBY AND ALADAR FLEISCHMAN FACULTY OF ENGINEERING

The Zandman-Slaner Graduate School of Engineering

# **The Internet PoP Level Graph**

**By**

**Noa Zilberman**

THESIS SUBMITTED TO THE SENATE OF TEL-AVIV UNIVERSITY

in partial fulfillment of the requirements for the degree of

"DOCTOR OF PHILOSOPHY"

Under The Supervision of Prof. Yuval Shavitt

October, 2013



This work was carried out under the supervision  
of Prof. Yuval Shavitt



## Acknowledgments

I would like to thank all the people who supported me along the way, professionally and personally.

I had the privilege of having Prof. Yuval Shavitt as my MSc. and PhD. thesis advisor. Beyond the technical and research skills that I have learned from him, he had opened to me a window to a world of new research areas and topics. This thesis covers only a fraction of the studies I have done in parallel to this work, for the sheer academic interest.

I would like to pay special thanks to the entire DIMES team, specifically to Ido and Boaz, who fight every day to make the servers work and the agents measure. I would also like to thank Lior Neudorfer and Dima Feldan who contributed to different parts of this thesis as part of their MSc thesis. I would like to acknowledge all the other people in the DIMES and Networks lab who supported this work: Udi, Noam, Danny, Galina, Eyal, Dror, Elad, Anat, Yaron, Ohad, and all those who I have forgotten to mention. A special place is saved for Tomer Tankel, may he rest in peace.

During my PhD studies I had the pleasure to be the teaching assistant of Prof. Boaz Patt-Shamir. He had taught me many lecturing practices as well as details of networking essentials and for which I would always thank him.

I would like to thank the team in Dune Networks (Broadcom) who allowed me to complete this thesis in parallel to my work, who taught me and supported me: Meir, Meir, Ofer, Eyal and the rest of the team.

Last, I would like to thank my family. I would keep your names confidential, but my appreciation is endless. I hope this thesis makes you proud.





## Abstract

Service providers tend to place multiple routers in a single location called a Point of Presence (PoP), which serves a certain area. Inferring PoP level maps is gaining interest due to its importance to many areas, e.g., for tracking the Internet evolution and studying its properties. In this thesis I introduce a novel structural approach to automatically generate large scale PoP level maps using traceroute measurements from multiple locations. The PoPs are first identified based on their structure, and are then assigned a location using information from several geo-location databases. I discuss the tradeoffs in this approach and provide extensive validation details. The generated maps can be widely used for research, and I provide some possible directions.

The geographical location of Internet IP addresses is important for academic research, commercial and homeland security applications. Thus, both commercial and academic databases and tools are available for mapping IP addresses to geographic locations. Evaluating the accuracy of these mapping services is complex since obtaining diverse large scale ground truth is very hard. As an example for the usage of PoPs, I show how they can be used to test the accuracy of IP Geolocation services. I am able to group close to 100,000 IP addresses worldwide into groups that are known to share a geo-location with high confidence. I provide an insight into the strength and weaknesses of IP geolocation databases, and discuss their accuracy and encountered anomalies. I show that while commercial databases claim to have a very high level of accuracy, the correctness of their databases is questionable. Academic tools, based on delay measurements, were shown to have a large range of error as well. In the third part of the thesis, I present a novel algorithm that crawls the Internet PoP level graph to improve the accuracy of geolocation, combining information from both geolocation databases and delay measurements. The algorithm uses PoPs with high level of confidence (as defined in the first part of the thesis) to improve the location of PoPs with lower confidence, iteratively, and then geolocate IP addresses. I show that the results provided by the algorithm are more accurate than geolocation databases information while avoiding the pitfalls of delay measurements' usage.

Considerable research is done in order to infer the undisclosed commercial relationships between ASes. These relationships, which have been commonly classified to four distinct Type of Relationships (ToRs), dictate the routing policies between ASes. The next part of this work leverages PoP level maps to improve AS ToR inference. It proposes a method which uses PoP level maps to find complex AS relationships and detect anomalies on the AS relationship level. I present experimental results of using the method on ToR reported by CAIDA and report several types of anomalies and errors. The results demonstrate the benefits of using PoP level maps for ToR inference, requiring considerably less resources than other methods theoretically capable of detecting similar phenomena.

The last part of this work sets the foundations to the development of an evolution model of the Internet based on the PoP level. The PoP topologies of the Internet are annotated with geographical, economical and demographical information to achieve an understanding of the dynamics of the Internet's structure at different time periods, in order to identify the constitutive laws of Internet evolution. These can be used to develop a realistic topology generator and a reliable forecast framework that can be used to predict the size and growth of the Internet as economies grow, demographics change, and as-yet unattached parts of the world connect.



## Table of Contents

<b>Acknowledgments</b>	<b>vii</b>
<b>List of Figures</b>	<b>1</b>
<b>List of Tables</b>	<b>4</b>
<b>1: Introduction</b>	<b>5</b>
<b>2: Related Work</b>	<b>8</b>
2.1 Classification of PoPs . . . . .	8
2.2 Geolocation of PoPs . . . . .	11
2.3 PoP Maps Validation . . . . .	13
2.4 Applications of PoP Maps . . . . .	15
2.5 AS Relationship Inference . . . . .	16
2.6 Internet Evolution Models . . . . .	17
<b>3: Classification of PoPs</b>	<b>19</b>
3.1 PoP Extraction Algorithm . . . . .	19
3.2 PoP Extraction Validation . . . . .	25
<b>4: Assigning a Geographical Location to PoPs</b>	<b>30</b>
4.1 Naive Geolocation Algorithm . . . . .	30
4.2 Validation of the Naive Geolocation Algorithm . . . . .	31
4.3 A Survey of Geolocation Databases . . . . .	34

4.3.1	Background . . . . .	34
4.3.2	Geolocation Services . . . . .	36
4.3.2.1	Databases Accuracy . . . . .	38
4.3.3	The Evaluation Model . . . . .	39
4.3.3.1	Data Evaluation Method . . . . .	39
4.3.3.2	Dataset . . . . .	40
4.3.4	Results . . . . .	41
4.3.4.1	Basic Tests . . . . .	41
4.3.4.2	Ground Truth Location . . . . .	45
4.3.4.3	Comparison Between Databases . . . . .	47
4.3.4.4	Database Anomalies . . . . .	51
4.3.4.5	Database Changes . . . . .	54
4.3.5	Discussion . . . . .	54
4.3.5.1	Active Measurement Accuracy . . . . .	56
4.3.6	Conclusion . . . . .	57
4.4	Crawling Geolocation Algorithm . . . . .	57
4.5	Crawling Geolocation - Datasets . . . . .	62
4.6	Validation of the Crawling Geolocation Algorithm . . . . .	63
4.6.1	Algorithm's Parameters . . . . .	64
4.6.2	Validation of Location Assignment . . . . .	66
4.6.3	Comparison to Geolocation Databases . . . . .	69
4.6.4	IP Geolocation . . . . .	71
<b>5:</b>	<b>Connectivity in PoP Level Maps</b>	<b>73</b>
5.1	Defining PoP Level Connectivity . . . . .	73
5.2	Exploring PoP Level Connectivity . . . . .	74

<b>6: Improving AS Relationship Inference Using PoPs</b>	<b>78</b>
6.1 Background . . . . .	78
6.2 Analysis Process . . . . .	78
6.2.1 Preprocessing . . . . .	79
6.2.2 ToR augmentation . . . . .	79
6.2.3 Complex ToRs and anomaly detection . . . . .	80
6.3 Datasets . . . . .	81
6.4 Results . . . . .	82
6.4.1 Preprocessing Results and ToR augmentation . . . . .	82
6.4.2 Sensitivity analysis . . . . .	83
6.4.3 Anomaly detection . . . . .	84
6.4.3.1 AS Prefix Resolution Errors . . . . .	84
6.4.3.2 IXP and sibling detection . . . . .	85
6.4.3.3 ToR inference errors . . . . .	86
6.4.3.4 Complex AS relationships . . . . .	86
6.4.3.5 Odd Academic ToRs . . . . .	87
6.4.3.6 Traceroute errors . . . . .	87
6.4.3.7 Unresolved Anomalies . . . . .	88
6.4.3.8 ToR Datasets . . . . .	88
6.4.3.9 ToR inference at different layers of aggregation . . . . .	89
6.4.3.10 Dataset Size Dependence . . . . .	89
6.4.3.11 Validation . . . . .	90
<b>7: Setting the Foundations for PoP-Based Internet Evolution Models</b>	<b>91</b>
7.1 Background . . . . .	91
7.2 Datasets and Datasets Limitations . . . . .	92

7.2.1	Datasets Limitations . . . . .	93
7.3	Analysis . . . . .	94
7.3.1	The Relation between PoPs and Population . . . . .	94
7.3.2	The Relation between PoPs and GDP . . . . .	98
7.3.3	The Relation between PoPs and Internet Users . . . . .	100
7.3.4	A Study of the United States . . . . .	101
7.3.4.1	The Relation between PoPs and Population in the US . . . . .	102
7.3.4.2	The Relation between PoPs and GDP in the US . . . . .	103
7.3.4.3	The Relation between PoPs and Transportation in the US . . . . .	104
7.3.4.4	The Relation between PoPs and Demographic Factors in the US . . . . .	106
7.3.4.5	The Relation between PoPs and TV Market in the US . . . . .	108
7.3.4.6	The Relation between PoPs and Sports Teams in the US . . . . .	109
7.4	Discussion . . . . .	110
<b>8:</b>	<b>Conclusion</b>	<b>112</b>
	<b>References</b>	<b>114</b>

## List of Figures

1.1	The Internet's Levels of Aggregation . . . . .	6
3.1	Typical Network Connection . . . . .	20
3.2	Parent-Child classification . . . . .	22
3.3	Bipartite graph example . . . . .	23
3.4	Number of Discovered PoPs vs. Number of measured IP Edges . . . . .	27
3.5	Number of IPs in PoPs vs. Number of measured IP Edges . . . . .	27
3.6	Number of PoPs vs. Maximal Delay . . . . .	28
3.7	Number of IPs in PoPs vs. Maximal Delay . . . . .	28
3.8	Number of PoPs vs. Minimal Number of Measurements . . . . .	28
3.9	Number of IPs in PoPs vs. Minimal Number of Measurements . . . . .	28
4.1	PoPs World Map . . . . .	32
4.2	QWEST US PoP Map . . . . .	32
4.3	Map Of DIMES Agents, March-2010 . . . . .	40
4.4	Map Of Discovered PoPs, March-2010 . . . . .	41
4.5	Range of Convergence Within Databases . . . . .	43
4.6	CDF of Location Votes Percentage Within 100km From PoP Center . . . . .	44
4.7	CDF of Location Votes Percentage Within 500km From PoP Center . . . . .	45
4.8	RMS Distance[km] Between Databases - Heatmap . . . . .	48
4.9	Median Distance[km] Between Databases - Heatmap . . . . .	48
4.10	CDF of Distances[km] Between Databases . . . . .	49

4.11	CDF of Database Location Deviation From PoP Median - 500km Range . . . . .	49
4.12	Breakdown of deviation from PoP majority CDF By region - 500km Range . . . . .	49
4.13	CDF of Database location deviation from PoP majority . . . . .	49
4.14	Database Location Deviation From PoP Median vs. Range of Convergence . . . . .	51
4.15	Mismatch Between Databases - UUNET . . . . .	53
4.16	Mismatch Between Databases - Global Crossing . . . . .	54
4.17	Breakdown of location votes percentage CDF for Spotter by region. . . . .	56
4.18	Breakdown of convergence range CDF for Spotter by region. . . . .	56
4.19	Crawling Algorithm . . . . .	61
4.20	Number of High Level Confidence PoPs vs. Thresholds Values . . . . .	64
4.21	Telefonica PoPs Location Map . . . . .	70
4.22	A CDF of a Large ISP Relocation Range of Error . . . . .	70
4.23	Heatmap of Median Distance Between Geolocation Services . . . . .	71
5.1	An Internet PoP Level Connectivity Map - A Partial DIMES Map of Week 42, 2010 .	74
5.2	Number of Edges within a Link vs. Number of PoP Level Links in DIMES Dataset .	75
5.3	Number of Links per PoP vs. Number of PoPs in DIMES Dataset . . . . .	76
5.4	Links Delay vs. Number of Links in DIMES Dataset . . . . .	77
5.5	Link Delay vs. Edge Delay in DIMES Dataset . . . . .	77
6.1	Anomaly Detection vs. Thresholds Values. . . . .	84
6.2	AS Prefix Resolution Error - Example . . . . .	85
6.3	Complex AS Relationship - Example . . . . .	85
6.4	Academic ToR Anomaly - First Example . . . . .	88
6.5	Academic ToR Anomaly - Second Example . . . . .	88
7.1	Country Level Population vs. Number of PoPs . . . . .	95



7.2 Country Level Population vs. Growth in Number of PoPs . . . . . 96

7.3 City Level Population vs. Number of PoPs . . . . . 97

7.4 Country Level GDP vs. Number of PoPs . . . . . 99

7.5 Country Level GDP (PPP) per Capita vs. Number of PoPs . . . . . 99

7.6 Country Level GDP x Country Level Population vs. Number of PoPs . . . . . 100

7.7 United States City Level Population vs. Number of PoPs . . . . . 102

7.8 United States Metropolitan Level GDP vs. Number of PoPs . . . . . 103

7.9 United States Metropolitan Level Ranking of GDP vs. Ranking of PoPs . . . . . 104

7.10 United States Metropolitan Level Ranking of Highway Congestion vs. Ranking of PoPs 106

7.11 United States Number of TV Homes vs. Number of PoPs . . . . . 108

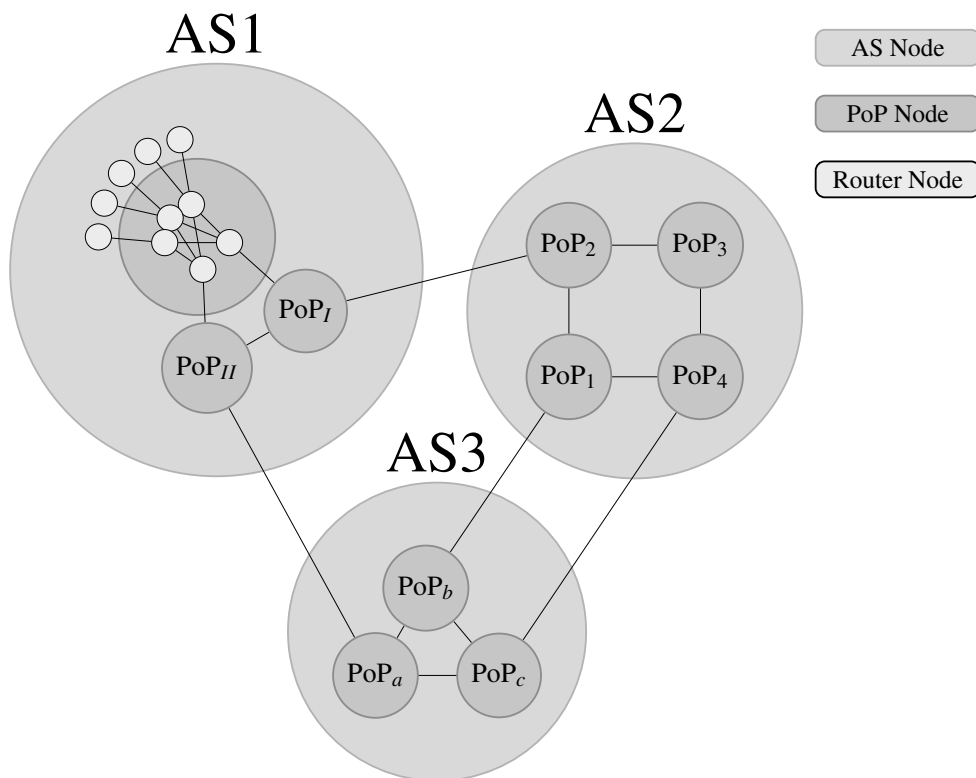
## List of Tables

3.1	Common network motifs in IP interconnections networks of three ASes. . . . .	20
3.2	Changes in PoP maps between different time frames . . . . .	26
4.1	Geolocation Database Accuracy As Reported By Vendor . . . . .	39
4.2	Null IP Address Information . . . . .	42
4.3	Comparison with Caida’s Ground Truth Database . . . . .	46
4.4	PoP Relocation By Algorithm’s Stage . . . . .	63
4.5	PoP Relocation Failures Breakdown . . . . .	64
4.6	Number of Co-Located PoPs vs. Threshold Value . . . . .	65
4.7	Known PoPs Relocation Accuracy . . . . .	68
4.8	IP Geolocation By Algorithm’s Stage . . . . .	71
7.1	Population vs. Number of PoPs on Country Level (Selected Countries) . . . . .	95
7.2	Population vs. Number of PoPs on City Level (Leading Cities in Selected Countries)	96
7.3	Correlation Coefficient Between Size of Different Age Groups and The Number of PoPs . . . . .	107
7.4	Correlation Coefficient Between Size of Different Race Groups and The Number of PoPs . . . . .	108

# I Introduction

The study of the Internet topology attracted a great deal of work over the years. A good survey of these efforts can be found in [27, 28, 125]. Internet topology maps are used for a vast number of applications, such as building models of the Internet [83], studying the robustness of the network [29], network management [102] and improving routing protocols design [88]. There are several levels Internet maps are presented at, each level of abstraction is suitable for studying different aspects of the network. The most detailed level is the IP level, which represents separately each and every IP interface connected to the network. Many projects map the Internet at the IP level, such as Skitter [55], RIPE NCC's Test Traffic Measurement [38], iPlane [72], DIMES [25], Ark [58], and more. This level is far too detailed to comprehend the network map as well as understand tradeoffs between connectivity and redundancy [125], and the large number of entities makes it very hard to handle. One level above the IP level is the router level, aggregating multiple IP interface addresses to a router, using alias resolution, as done by projects such as Mercator [44], MIDAR [64], Ally [105], and RadarGun [15]. While being less detailed than the IP level, this level of aggregation is still highly detailed and difficult to handle. The most coarse level is the Autonomous System (AS) level. It is most commonly used to draw Internet maps, as it is relatively small (tens of thousands of ASes) and therefore relatively easy to handle: there is only one node for every AS, and a link between two nodes is drawn if the corresponding ASes have a direct peering relationship. There are different methods to discover the Internet's AS-level topology, from using traceroutes, as done in Ark, iPlane, and DIMES, through BGP announcements, as done by Routeviews [117] to Internet Routing Registries (IRR) [79]. One limitation of using AS information for Internet mapping is that AS sizes may differ by orders of magnitude. While a large AS can span an entire continent, and a small one can serve a small community, yet both seem identical at the AS level map.

An interim level between the AS and the router graphs is the PoP level. Service providers tend to place multiple routers in a single location called a Point of Presence (PoP), which serves a certain geographical area. A PoP is defined as a group of routers, which belong to the same AS and are physically located at the same building or campus.



**Figure 1.1: The Internet's Levels of Aggregation**

Figure 1.1 demonstrates the Internet aggregation levels. The figure presents for clarity only the AS, PoP, and router levels. Every AS, marked by a large circle, is made of a network of routers, marked by small light gray circles. The routers may be part of a PoP (colored dark gray), or reside outside of a PoP. A router which is not part of a PoP will still be connected to other routers, eventually connecting to a PoP. The points of presence are connected to other PoPs within the same AS as well as to PoPs outside their AS, thus creating AS level connectivity.

The technological nature of PoPs varies between service providers as well as within the same network. Some PoPs operate entirely on the IP level, while other PoPs employ MPLS and VPLS switching. In many cases, service provider mix switching and routing within the same PoP, combining both MPLS and IP. In more rare cases, in Optical Transport Networks, the PoP may only serve as a channel based cross connect. A good example of this mix is shown in CenturyLink's network [17]: In some cities, such as Atlanta, Los Angeles, and New York City, both IP and MPLS/VPLS are used. In other cities, such as Sacramento, Duluth, and Cambridge, MA, there is an IP PoP, while in cities such as New Orleans, San Antonio, and San Diego only MPLS/VPLS is used. Additional examples can be found in the TeliaSonera network map [111] and XO network map [127] <sup>1</sup>. Service provider also tend to distinguish between different types of PoPs, often referring to the hierarchy in the network, e.g.,

<sup>1</sup>This information was also confirmed with a large networking equipment provider

access or backbone PoP [17] or to the area it covers, e.g., a metro PoP [127]. A declining trend is to refer to PoPs by their capacity, such as GigaPoP [61] or TeraPoP<sup>2</sup> [87].

When studying the entire network, and not only specific ISPs, PoP maps give a better level of aggregation than router level maps with a minimal loss of information. PoP level graphs provide the ability to examine the size of each AS network by the number of physical co-locations and their connectivity instead of by the number of its routers and IP links. Points of presence can be annotated with geographical location, as well as information about the size of the PoP. PoP maps can also preserve routing information by annotating links connecting PoPs that belong to different ASes with the type of relationship (ToR). Thus, using PoP level graphs it is possible to detect important nodes of the network and understand network dynamics as well as many more applications.

This chapter surveys the study of Internet PoP level maps, providing an overview of all related works done so far in this field. The chapter is organized as follows: Section 2.1 discusses classification of IP addresses into PoPs and surveys existing works in this field. Section 2.2 describes some methods for assigning a location to points of presence. The validation efforts of PoP level maps is surveyed in Section 2.3. In Section 2.4 I discuss applications of the PoP level graphs by various disciplines.

---

<sup>2</sup>As called by Qwest, before Qwest was acquired by CenturyLink

## II Related Work

### 2.1 Classification of PoPs

The first attempts to explore the PoP level graph were done by Andersen *et al.* [9] and Spring *et al.* [105]. Spring *et al.* [105] tried to infer ISP topologies both on the router and the PoP level. The focus of their contribution was in alias resolution and router identification based on in-order IP identifiers and introducing Rocketfuel, their mapping engine. The PoP resolution was entirely DNS based. To this end, they inferred ISP naming convention. For example, s1-bb11-nyc-3.0.sprintlink.net is indicated to be a Sprint backbone (bb11) router in New York City (nyc). The naming convention was deduced from the list of router names they gathered during the alias resolution and router identification stage with some city names taken from [82]. For routers with no DNS names or where the names lacked location information, the locations of neighbor routers were used. The generated PoP map did not distinguish between backbone network nodes, data centers, or private peering points.

Ten ISPs were tested by Spring *et al.* and the number of PoPs discovered per ISP ranged from 11 (AS4755, VSNL India) to 122 (AS2914, Verio US). The PoPs' analysis showed that the designs of PoPs were relatively similar: generic PoPs are built from a few routers connected in a mesh while in large PoPs the access routers connect one or more routers from a neighboring domain and to two backbone routers for redundancy. The backbone routers connect both to routers within the same PoP as well as to routers in other PoPs that connect to the ISP's backbone. The result showed that small PoPs had for redundancy two backbone PoPs, but in large PoPs with 20 routers or more, the number of backbone routers varied significantly, from two to thirty.

Andersen *et al.* [9] used passive monitoring of BGP messages to cluster IP prefixes into PoPs: In the preprocessing stage, BGP messages are grouped into time intervals of  $I$  seconds and massive updates due to session resets are filtered. The clustering stage is based on a distance metric, which is a function that determines how closely two items are. The distance metric used is the correlation coefficient between every pair of BGP update vectors.  $u_p^{(t)}$  denotes the update vector for each prefix

p:

$$u_p^{(t)} = \begin{cases} 1 & \text{if } p \text{ updated during interval } t \\ 0 & \text{otherwise} \end{cases} \quad (2.1)$$

$C(p1, p2)$  is the correlation coefficient between two prefixes, with  $\bar{u}_p$  being the average of  $u_p^{(t)}$  and  $\sigma_p$  its variance.

$$C(p1, p2) = \frac{\frac{1}{n} \sum_{t=1}^n (u_{p1}^{(t)} - \bar{u}_{p1})(u_{p2}^{(t)} - \bar{u}_{p2})}{\sqrt{\sigma_{p1}^2} \sqrt{\sigma_{p2}^2}} \quad (2.2)$$

A Single-linkage clustering algorithm [132] is applied for grouping prefixes. Using the distance metric presented by Equation 2.2, each pairwise distance between two prefixes is computed and prefixes with time window of 30 seconds are grouped.

Andersen *et al.* used BGP updates from two upstream feeds: a commercial feed via Genuity (AS 1), and an Internet2 feed via the Northeast Exchange (AS 10578). Due to their configuration, only the best route to every prefix was recorded, thus some paths were omitted from their dataset. The clustering was conducted on 2338 prefixes announced by UUNET (AS 701) and 1310 prefixes announced by AT&T (AS7018) and ended up with 6 clusters in UUNET and 5 clusters in AT&T, with the number of clusters strongly dependent on the number of pairwise comparisons during the clustering phase. The analysis observed the effect of the number of matches on the number of clusters and their accuracy. The validation was conducted based on three methods: IP address similarity (the number of IP addresses that separate two prefixes), Ratio of shared to unshared traceroute path length (in hops) and DNS-based PoP comparison. The last means that they extracted a router location from the ISP's naming convention, managing to assign 97% of UUNET hops and somewhat less for AT&T. Their results showed that correlation-based clustering grouped the UUNET prefixes into about 1200 clusters while with over 95% PoP-level accuracy as well as 900 clusters in AT&T with 97% accuracy. The accuracy is defined as a match between the naming conventions. The concluding observation is that clusters that are announced and withdrawn together tend to be located at the same PoP.

The iPlane project [73, 72] generates PoP level maps based on the Rocketfuel's approach, with several improvements: First, they determine the DNS names assigned to network interfaces, using two data sources: Rocketfuel's undns utility [105] and data from the Sarangworld project [2]. DNS alone is

not enough, as some interfaces have no DNS names, others have no rules to infer their DNS name and some interfaces may be misnamed, thus incorrect locations can be inferred [130]. For the last, interfaces are probed using ICMP ECHO packets and interfaces where the RTT is smaller than expected are filtered. The main new contribution in this work is an algorithm that clusters router interfaces based on their responses when probed from a large number of vantage points. iPlane estimates the number of hops on the reverse path back from a router to the vantage point, guessing the initial TTL value used by the router. The assumption is that routers in the same AS and geographically co-located take the same reverse path back to the vantage point from which they were probed, while routers that are not co-located will not display similar reverse path. iPlane detects about 135K PoPs, about 56K of them in singleton clusters, meaning a single router in a cluster. iPlane uses the inter-cluster connectivity to generate PoP level connectivity, using bidirectional links. The delay measured on links is as follows: For every inter-PoP link, iPlane considers all the corresponding inter-IP links that are measured in traceroutes. From every traceroute in which any such inter-IP link is observed, obtain one latency sample for the link as the difference in RTTs to either end of the link and drop all latency samples that are below 0. Compute the latency for the inter-PoP link as the median of all the remaining samples for it. If there are no samples left after ignoring all the negative latency samples, the latency of the link is indicated as -9999 (about 6% of the links).

Yoshida *et al.* [129] mapped PoP-paths in Japan using thirteen dedicated measurement nodes and measuring the delay between these nodes. They tried to map the core network delay, derived from the end-to-end delays and access delays and their corresponding PoP level paths, using a set of delay equations:

$$\text{delay}(src, dst) = ad_{src} + ad_{dst} + \sum_{p,q \in N} x_{p,q} \times cd_{p,q} + E_{src, dst} \quad (2.3)$$

In the equation,  $N$  denotes a set of candidate PoP locations of a measured ISP;  $p$  and  $q$  satisfy  $p, q \in N$ ;  $ad_{src}$  and  $ad_{dst}$  denote the access delay at the source and the destination;  $cd_{p,q}$  denotes a core delay between  $p$  and  $q$ ;  $E_{src, dst}$  is the measurement error of the delay;  $x_{p,q} = 1$  if a direct path between  $p$  and  $q$  exists and the path is used to connect between  $src$  and  $dst$ , otherwise  $x_{p,q} = 0$ .  $\text{delay}(src, dst)$ ,  $ad_{src}$  and  $ad_{dst}$  are measurable through end-to-end measurements and  $cd_{p,q}$  can be derived leveraging the distance between  $p$  and  $q$ . To solve the equation, several restrictions are applied. One of the assumptions used is that the network connections are deployed along other infrastructure services,



such as railroads and expressways.

The work distinguished between five types of networks, differing by the way the backbone routers are structured and by the way layer two is used. For example, if layer three being used in every location in the network, or are layer three routers being used only in highly populated cities.

A different approach to PoP level maps is presented by Rasti *et al.* [89]. They term an eyeball AS as an individual Autonomous Systems that directly provides service to end-users and use the eyeball ASes to estimate the PoP-level footprint. The basic assumption is that each AS must have a PoP in areas it has a high concentration of customers. Therefore, the AS eyeball offers a view of that AS's PoP-level infrastructure, referred to as PoP-level footprint. The algorithm begins by gathering a large number of end-user IP addresses, collected by crawling P2P applications. The users are then mapped to cities using geolocation services (discussed in Section 2.2) and are grouped to ASes based on Routeview's BGP tables [117]. Given the locations of the users, the geographical regions where the AS offers service to end-users is inferred using KDE (Kernel Density Estimation). To extract the PoP footprint, local maxima  $D(i)$  are detected in the density function, with the highest peak denoted by  $D_{max}$ . PoPs are indicated by any peak  $D(i)$  that is within a given range from  $D_{max}$ , meaning  $D(i) > \alpha \times D_{max}$ , with  $\alpha$  set to 0.01. The work focused on 672 ASes and found an average of 13.6 PoPs per AS when using 40km range as the kernel function bandwidth.

To conclude, there are several different approaches to the classification of IP addresses into PoPs. These methods are mostly stochastic and individual IPs may be placed in the wrong PoP. Still, grouping the IP addresses into PoPs is just the first stage of generating PoP level maps, as discussed in the following sections.

## **2.2 Geolocation of PoPs**

An important feature of PoP level maps is the ability to assign a geographical location to PoPs. The assignment is done using geolocation mapping services, providing longitude and latitude or a city and a country per IP address. Geolocation mapping services can be divided to several groups. The first group of geolocation mapping services is geolocation databases, holding a table mapping every IP address to its geographical location. Geolocation databases range from free services to services that cost tens of thousands of dollars a year. The most basic services use DNS resolution as the

basis for the database [105], while others use proprietary means such as random forest classifier rules, hand-labeled hostnames [4], user's information provided by partners [24], and more.

Another group of geolocation mapping services is based on network measurements. IP2Geo [82] was one of the first to suggest a measurement-based approach to approximate the geographical distance of network hosts. A more mature approach is constraint based geolocation [49], using several delay constraints to infer the location of a network host by a triangulation-like method. Later works, such as Octant [126] used a geometric approach to localize nodes within a 22 miles radius. Katz-Bassett *et al.* [63] suggested topology based geolocation using link delay to improve the location of nodes. Yoshida *et al.* [129] used end-to-end communication delay measurements to infer PoP level topology between thirteen cities in Japan. Eriksson *et al.* [30] applied a learning based approach to improve geolocation. They reduced IP geolocation to a machine learning classification problem and used Naive Bayes framework to increase geolocation accuracy.

One online geolocation service that allows querying specific IP addresses is Spotter, which is based on a work by Laki *et al.* [70]. Spotter uses a probabilistic geolocation approach, which is based on a statistical analysis of the relationship between network delay and geographic distance. To approximate the location of a target, Spotter measures propagation delays from landmarks to the target, and then converts the delays into geographic distances based on a delay-distance model. The resulting set of distance constraints is used to determine the target's estimated location with a triangulation-like method.

Not many works have focused on the accuracy of geolocation databases, but those who did showed them to be inaccurate: In 2008, Siwpersad *et al.* [103] examined the accuracy of Maxmind [76] and IP2Location [51]. They assessed their resolution and confidence area and concluded that their resolution is too coarse and that active measurements provide a more accurate alternative. Gueye *et al.* [47] investigated the imprecision of relying on the location of blocks of IP addresses to locate Internet hosts and concluded that geolocation information coming from exhaustive tabulation may contain an implicit imprecision. Muir and Oorschot [81] conducted a survey of geolocation techniques used by geolocation databases and examined means for evasion/circumvention from a security standpoint. Poese *et al.* [84] studied five databases and showed that while on the country-level they are rather accurate, the databases are highly biased towards a few popular countries. Using ground truth information from one large European ISP and using DNS names as clues for two large other major ISPs,

Poese *et al.* showed that the evaluated databases performed poorly on those ISPs.

Most of the PoP extraction algorithms described in Section 2.1 use a crude method of geolocation as the basis for their geolocation: DNS names. This is an easy to use method, leveraging the fact that the router's location is often written in the router's name used by the ISP. However, DNS suffers from several problems: many interfaces do not have a DNS name assigned to them, and incorrect locations are inferred when interfaces are misnamed [130]. In addition, rules for inferring the locations of all DNS names do not exist, and require some manual adjustments. Assigning a geographical location to PoPs is therefore a difficult task which is hard to validate without ground truth information.

### **2.3 PoP Maps Validation**

An important question when examining PoP level maps is how the map was validated. Accuracy is the most important validation evaluation aspect, and it entails multiple facets.

- How accurate is the classification of IP addresses to PoPs?
- How accurate is the assignment of PoPs to geographic locations?
- How accurate is the inference of PoP level links and their delay?

In addition, one may also want to evaluate the coverage of PoPs, meaning how many of the actual ISP's PoPs are covered by the extracted PoPs map, and how many IPs of a PoP are assigned to it. The effort required to validate PoP level maps is thus considerably high.

Spring *et al.* [105] verified completeness with the help of three ISPs. The ISPs verified that no PoPs or inter-PoP links were missing. However, in two of the cases there were spurious links. In addition, some access PoPs were missing. Further validation was conducted on the router level, both for completeness, impact of measurement reductions and alias resolution. The alias resolution, used for PoPs detection, failed for about 10% of the IP addresses, and in Sprint network, 63 out of 303 routers were resolved incorrectly.

Andersen *et al.* [9] did not focus on the validation of their PoP maps results, rather they presented the impact of different aspects of their clustering algorithm on the results. The PoP level maps were in fact used to validate the clustering results.

The iPlane PoP level maps [72], which are mostly based on the Rocketfuel's approach, focus their validation efforts on the inter-PoP connectivity. The validation uses measurements taken from 37 Planet Lab nodes to destinations in 100 random prefix groups. The first step in the validation is end-to-end latency error estimation. Next, the two path based and latency based delay estimations are compared to the results of Vivaldi [20]. They find that 73% of their predictions obtained using the path composition approach are within 20 ms of the actual latency.

AS eyeballs [89] was validated by comparing the AS eyeballs results with public PoP maps information published by 45 ASes. The scope of the averaging done using the KDE method is controlled by the bandwidth of a kernel function. The validation showed that when kernel bandwidth was 40km, for 60% of ASes only 20% of the PoP locations matched the service provider's map. However, for the top 10% ASes the locations match was over 50%. On the average, 41% of the PoP locations matched the location on the reference ISP's map. Increasing the kernel bandwidth to 80km increased the match to 60%, but decreased the number of PoPs found. Rasti *et al.* found that two causes for inaccuracy in their approach were the existence of multiple PoPs within a short distance and the placement of some PoPs away from major end users concentrations. They also compared their map with DIMES' map and found that for 80% of the eyeball ASes, the identified PoPs were a superset of DIMES'.

The Internet Topology Zoo [66] maps originate from the network operators, and are thus considered reliable. While an ISP may present a somewhat simplified network map, this aspect can be considered negligible. A possible concern is the accuracy of maps' translation into transcripts: The maps are manually annotated by the project's team, with one researcher doing the annotation and another reviewing his work, however both works are manual. The project also omits large networks with graphic links that are tangled or hard to follow.

For all the cases presented above, the validation of the generated PoPs was a very hard task: While service providers provide graphic maps of their PoPs, the PoP's actual details and the address range used within the PoP's routers are being kept confidential. PoP maps are therefore best validated when checked by the ISP, yet this is not possible on a large scale map.

## 2.4 Applications of PoP Maps

PoP level maps can be used for a variety of applications. Understanding network topology and dynamics is one clear usage, as was done by Spring *et al.* [105]. Teixeira *et al.* [109] used PoP level topologies to study path diversity in ISP networks. The PoP level maps can also be used to evaluate and validate results of other properties of the networks, as done by Andersen *et al.* [9] who used them to check their clustering algorithm. Several works have considered the PoP level topology for delay estimation and path prediction [71, 72].

A new look at the Internet's topology is through dual AS/PoP maps: maps of the Internet that combine both the AS and the PoP level graph views, leveraging the advantages of each level of aggregation. One application of dual AS/PoP maps is the study of types of relationships between ASes. Using the geographical location of PoPs, one can explore not only the connectivity between ASes on the PoP level, but also how the relations between service providers change based on the location of the PoPs. Some work in this field was done by Rasti *et al.* [89], who looked at AS connectivity at the "Edge" in AS1267 (Infostrada) and AS8234 (RAI). They found that actual peering is significantly more complex than expected, e.g., a single PoP may use five peering PoPs in different ASes for upstream. Another application is distance estimation: instead of using router-level path stitching, one can find the shortest path between every two nodes on the dual map. The shortest path can then be used to find the distance between the two nodes. PoP level maps reduce the number of edges used for the path stitching, as multiple routers are aggregated into a single PoP, and the delay-based distance estimation is more accurate as the delay estimation of a PoP level link is better than that of a single IP-level edge. Last, the PoP location can be used to improve geolocation of each node and thus the distance estimation between the pair of nodes.

PoP level maps may also be useful for research related to homeland security. Schneider *et al.* [94] used DIMES' PoP level maps, which were generated as part of this work, to study the mitigation of malicious attacks on networks. They considered attacks on Internet infrastructure and found that cutting the power to 12% of the PoPs and 10% of power stations will affect 90% of the networks integrity. Following, they suggested ways to improve the robustness of the network by using link changes.

Annotating the PoP level maps with geographic, economic and demographic information, one can

achieve an understanding of the dynamics of the Internet’s structure at short and medium time scales, in order to identify the constitutive laws of Internet evolution. These can be used to develop a realistic topology generator and a reliable forecast framework that can be used to predict the size and growth of the Internet as economies grow, demographics change, and as-yet unattached parts of the world connect.

## **2.5 AS Relationship Inference**

Inferring the commercial relationships between service providers is an important line of research. The knowledge gained through the understanding of commercial relationships is used in research on Internet routing, can improve network performance as well as help increase its robustness. However, commercial relations between service providers are interesting first and foremost as they determine BGP routing policies between ASes. Contractual commercial agreements between Administrative Domains (which control Autonomous Systems) are usually not publicly disclosed, as so inferring them from measurement data has been a focus of many works. These relationships can be classified into three Types of Relationships (ToR) [57]: customer-to-provider (c2p), peer-to-peer (p2p), and sibling-to-sibling (s2s). Gao [35] was the first to present a method of inferring these relationships from publicly available BGP route data, and introduced the **valley free** AS path rule. An AS path is considered **valley free** if it consists of an uphill segment (customer to provider links), followed by an optional peer to peer link and a downhill segment (provider to customer links). Subramanian *et al.* [107] formally defined the “ToR Problem” as an optimization problem that seeks to find a ToR labeling for an AS graph which maximizes the number of valley-free paths. Di Battista *et al.* [13] and Erlebach *et al.* [31] showed that the ToR problem is NP-complete, and developed mathematically rigorous approximate solutions to the problem. Dimitropoulos *et al.* [26] acknowledged that a solution that maximizes the number of valley-free paths is not necessarily correct, and improved AS relationship detection by taking AS degrees into consideration. Shavitt *et al.* [98] suggested a near-deterministic algorithm for solving the ToR problem using an Internet Core, a subgraph of the Internet graph which contains the top-level providers. Their algorithm inferred AS relationships in AS paths by examining their relation to the Internet core.

The relationship between two ASes is sometimes more complex than a single ToR between all border routers. Gao [35] mentioned complex AS relationships as a cause for excessive sibling-sibling ToR

inference. Subramanian *et al.* [107] introduced AS path anomalies as specific patterns which cause paths not to be valley free. Dimitropoulos *et al.* [26] conducted a survey with several large ISPs, and revealed backup links and hybrid c2p/p2p relationships. A hybrid relationship is one in which two ASes connect in multiple peering points and have different types of relationships at these points.

## **2.6 Internet Evolution Models**

Understanding the Internet's infrastructure and topology alone is not enough. It is also important to learn the dynamics of the network and correlate its structure to its drivers in the physical world. These drivers may stem from economic incentives, geographic limitations or any other day-to-day life aspect, as was shown in previous works; Many models have been suggested over the years to explain the Internet's evolution, most of them were surveyed and discussed by Pastor-Satorras and Vespignani [83], but there are also later works such as Dhamdhere and Dovrolis [22], Wang and Loguinov [124] and Shakkottai *et al.* [96]. The models are mostly evolving in the abstract Internet AS graph with no connection to the real world geography, or with some naive connection with the Internet underlying geography. Some of these works, such as [23], look at the economic aspects of AS level network topology from the ISP's Type of Relationship (ToR) direction.

As time goes by, there is a growing understanding that the evolution of an Internet region should be estimated by tightly correlating the Internet structure with its underlying geography, and the changes in the economic, social, and even political evolution of the region in question. For example, as the economic status of a developing country improves, it results in a greater demand for Internet connectivity, leading to a growth in the Internet graph related to this region. There are only a few works in this research direction due to the difficulty of obtaining a good Internet map: Yook *et al.* [128] compared router, domain and population density in economically developed areas of the world and indicated that each of the three sets form a fractal with dimension  $D_f = 1.5 \pm 0.1$ . Combined with preferential link attachment they proposed an evolution model. Lakhina *et al.* [69] studied the geographical locations of Internet routers and showed that its density varied widely across the world, but that there is a strong superlinear relationship to population density in economically homogeneous regions. They also showed that the majority of link formation is based on geographic distance, and applied both aspects to the AS graph. Hameed *et al.* [50] used Rocketfuel's [105] PoP Topology and combined it with geographical locations based on population density and technology penetration.

They validated their results against the published PoPs locations of seven ISPs within the US. Mátray *et al.* [74] examined the spatial properties of the Internet topology and routing using Spotter. They analyzed the direction-dependence of geographic deviations and gave a description of router density in terms of the geographic layout of end-to-end paths.

The evolution of the Internet and its relationships to geographic and economic factors is also researched in other fields of study, though applying different methods and on a different scale. Roller and Waverman [91] studied how telecommunications infrastructure affects economic growth. This work was followed by other works, such as [19] and [68] that studied the economic impact of broadband infrastructure on growth. A recent research by Kolko [67] studied the relationship between broadband expansion and local economic growth in the US, but surveyed more indicators, such as industry type, population density, employment and income. He found limited economic benefits for local residents stemming from broadband infrastructure. A different type of research comes from the field of urban studies, such as Vinciguerra *et al.* [123]. They modelled the evolution of infrastructure networks as a preferential attachment process, yet assumed that geographical distance and country borders provide barriers to link formation in infrastructure networks.



### III Classification of PoPs

#### 3.1 PoP Extraction Algorithm<sup>1</sup>

We define a PoP as a group of routers which belong to a single AS and are physically located at the same building or campus. In most cases [93, 45] the PoP consists of two or more backbone/core routers and a number of client/access routers. The client/access routers are connected redundantly to more than one core router, while the core routers are connected to the core network of the ISP. Figure 3.1(a) shows a simple interconnection of four routers with a small number of interfaces. Assuming that during traceroute measurements ICMP replies are received from the incoming interfaces of the routers, the graph shown in Figure 3.1(b) is obtained. For example a traceroute measurement that enters our network through *interface A* on *router a* and leaves the network from *interface L* on *router b* will create an  $A \rightarrow I$  path on the graph. In a similar way a measurement that enters the network from *interface L* on *router b* and leaves it from *interface W* on *router c* will create a  $L \rightarrow C \rightarrow Y$  path on the graph. At the core of the Interface graph, which results from performing many traceroute measurements through a PoP, there is clearly a bi-partite graph. We look for this specific structure when trying to discover PoPs. Alon *et al.* [80] showed that many complex networks have repetitive patterns of interconnections, called ‘network motifs’, which became a standard term in the networks analysis community. Their work showed that real-world networks outside the communication field are not purely random, but have a higher than (or lower than) expected number of specific motifs. We have used their *mfinder* [1] package to search for motifs in graphs obtained by the DIMES measurements. In order to show the significance of a specific motif, the software uses the Z-score measure, which is calculated according to equation 3.1.

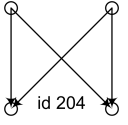
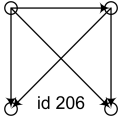
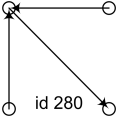
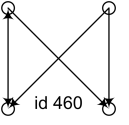
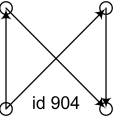
$$Z = \frac{X - \mu}{\sigma} \quad (3.1)$$

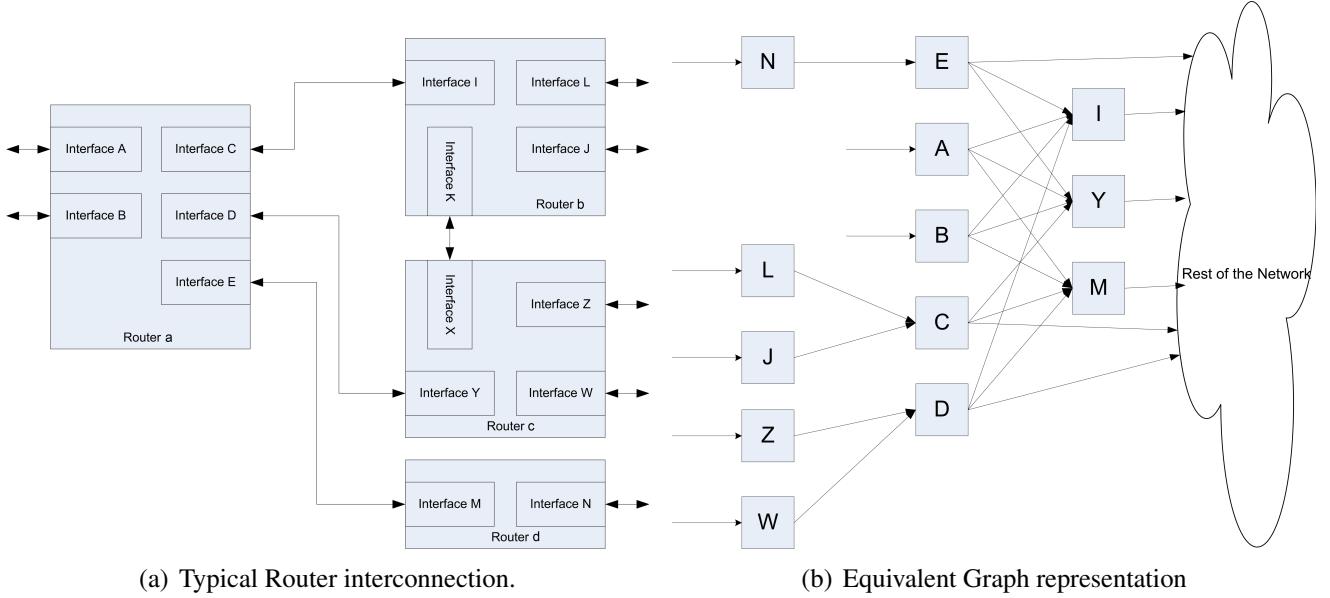
Where  $X$  is a number of a motif occurrences in a specific network, and  $\mu$  and  $\sigma$  are the mean and standard deviation of the motif occurrences within a certain random network. The number of motif

---

<sup>1</sup>The algorithm presented in this section is based on an algorithm introduced by Feldman and Shavitt in [33] and later improved in this work

**Table 3.1: Common network motifs in IP interconnections networks of three ASes.**

AS Number	Z-Score				
					
AS6395	377	-	9.51	43.84	148.39
AS5111	329.29	36.42	-	74.63	73.57
AS3549	154.8	5.38	37.87	19.51	-



**Figure 3.1: Typical Network Connection**

appearances in a random network is a stochastic function with mean and variance. The Z-score reveals how many units of the standard deviation a specific count of a motif is above or below the mean. Unsurprisingly, we have found a number of motifs with a high Z-score across all AS networks in the graph; partial results displayed in table 3.1 show the clear dominance of the ‘bi-fan’ motif (number 204) in three large providers, Global Crossing, France Telecom and Broadwing (now Level3). Note that motif 460 is bi-fan with one additional measurement in the reverse direction and motif 206 is a bi-fan with an additional measurement.

Although *mfinder* [1] is a very useful tool for identification of important motifs, it is not designed to be used for network clustering. In our work we do not look for a specific motif in the network, but for highly connected clusters as described in the previous chapter. However, we do search for ‘bi-fan’s (id204) repetitions under certain weight constrains as cores of the PoPs. The cores are extended with other close by interfaces. The following steps, introduced in [33], are used to reduce the IP level

graph  $G(V,E)$  to a PoP level network:

**Initial Partition.** Remove all edges with a delay higher than  $PD_{max\_th}$ , the PoP maximal diameter threshold, and edges with number of measurements below  $PM_{min\_th}$ , the PoP's edges measurements threshold.  $PM_{min\_th}$  is introduced in order to consider only links with a highly reliable delay estimation to avoid false indication of PoPs. As a result, a non-connected graph  $G'$  is obtained. Then, for each connected component of  $G'$  an induced sub graph is built by adding back all the edges that connect nodes of the connected component. Each connected group is a candidate to become one or more PoPs.

There are two reasons for a connected group to include more than a single PoP. First and most obvious is geographically adjacent PoPs, e.g., New York, NY and Newark, NJ<sup>2</sup>. Second is wrong delay estimation of a small number of links. For instance a single incorrectly estimated link between Los Angeles, CA and Dallas, TX might unify the groups obtained by such a naive method.

**Refined Partition.**

(a) *Parent-Child classification.* The next stage in the algorithm uses a classification to *parent pairs* and *child pairs*.

**Definition 3.1.1** *A pair of nodes is marked as parent if **both** of them point to two or more nodes.*

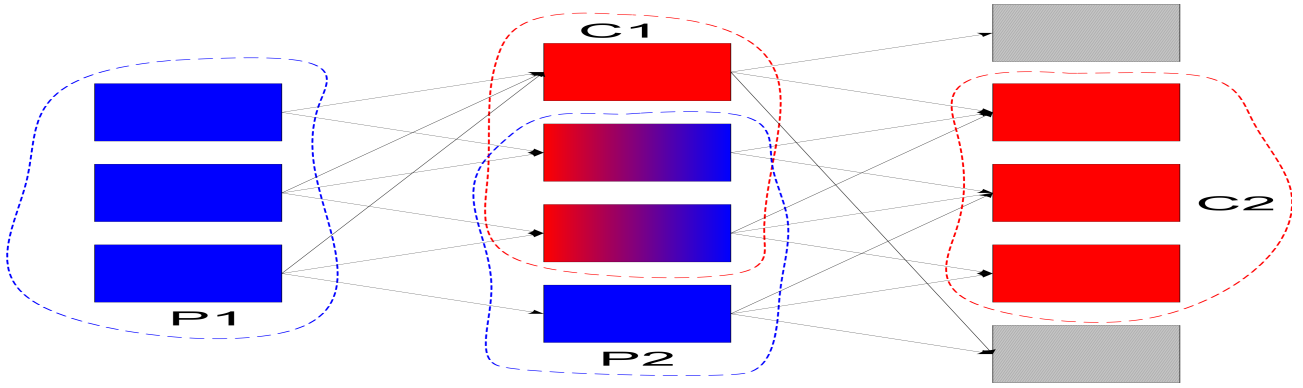
**Definition 3.1.2** *A pair of nodes are marked as child if **both** of them are pointed to by at least two nodes.*

All *parent* nodes are assigned to groups by pairwise unifying *parent* nodes. For example in figure 3.3, nodes  $\{1,2\}$ ,  $\{2,5\}$  and  $\{3,4\}$  are defined as *parent*, thus we obtain two *parent* groups  $\{1,2,5\}$  and  $\{3,4\}$ . The groups of *child* nodes are created according to the same process as defined for *parent* groups. Some nodes might belong to both categories and it is allowable for a node to belong to one *parent* group and to one *child* group. By definition, if a node belongs to two or more groups of the same kind, these groups are unified. Figure 3.2 shows an example of *parent/child* classification.

The PoP algorithm checks for each connected group extracted in the initial partitioning of the algorithm, if it contains more than one possible PoP. Note that each candidate partition looks like a collection of highly connected bipartite graphs with rich connectivity between them. The considered

---

<sup>2</sup>This situation was also identified by [89] after our initial publication

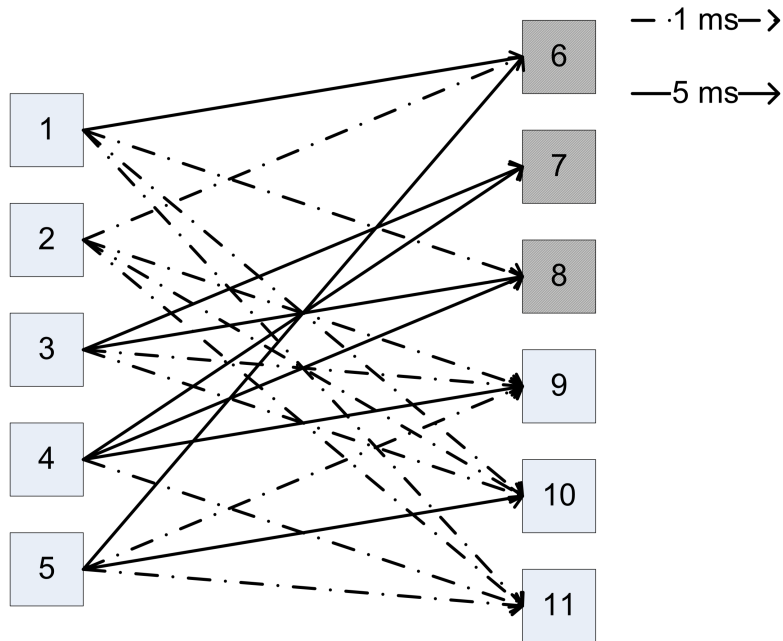


**Figure 3.2: Parent-Child classification: blue nodes (left) - *parent*, red nodes (right) - *child*, blue and red nodes (middle) - both parent and child, gray stripes nodes (right) - not classified**

partition of parents and children is then divided according to the measurement direction in the bipartite graph (each node or a group of nodes simultaneously can be a parent of one bipartite and a child of another). In this operation the weights of the edges are ignored. The minimal size of each group is two nodes.

(b) *localization*. Dividing the parents and children groups into physical collocations using the high connectivity of the bipartite graph. The input for the localization stage algorithm is a highly connected bipartite graph  $G(V, E)$  with a weight function  $W : E \rightarrow \mathbb{R}$  representing the estimated physical link delay, as shown in Fig. 3.3. The other input to the algorithm is a partition of the graph to the *parent/child* groups as previously described. The localization algorithm checks whether nodes of the same type (*parent/child*) belong to the same physical collocation. For this task the algorithm takes advantage of the topological structure of the group. For instance, if we check the parent group  $P$  we note that each child node of the group is pointed to by at least two parent nodes. Comparing the delays from the *child* nodes we can partition nodes of the *parent* group to one or more geographic collocations.

Formally, we represent each member of a group of two or more nodes (either *parent* or *child* group) in a coordinate space of the nodes that points to them using the weight of the edges. Next, we check the distance between each pair of nodes in that coordinate space. We assume that the link delay estimation errors in [32] are caused mainly by an impulse noise, i.e., most of the measurements are fairly precise or have only small noise, while a small portion of the measurements may have large errors. Therefore, unlike the Gaussian noise case, where Euclidean distance is used as a representation of the distance between nodes, we compare the similarity over the coordinates.



**Figure 3.3: Bipartite graph example, on the right side dark and bright nodes belongs to different collocation**

An example of the difficulties in determining geographic co-location is shown in Figure 3.3. By looking at the delay spread, one can easily determine that nodes 6-8 (darken) are not co-located with nodes 9-11. Looking at the distance between nodes 1-3 and nodes 9-11 it becomes clear that the former are also co-located. However deciding whether node 5 is also collocated with nodes 1-3 is not straightforward. Examining the delay spread between nodes 5 and 1-3 to nodes 9 and 11, gives a positive answer for collocation, since the measurement to node 10 that puts node 5 away from nodes 1-3 may be discarded as noise. The existence of yet another group of measurements to node 6, which is indecisive in its results, complicates the picture, and shows the difficulties in automating these decisions.

We propose the following deterministic algorithm to classify the locations of nodes in the bipartite graph. For each pair of parent nodes  $(u, v) \in P, u \neq v$ , we define the ‘common children’ group,  $CC$  by

$$CC(u, v) = \left\{ w \in G \mid (u, w) \in E \cap (v, w) \in E \right\} \quad (3.2)$$

We denote the members of  $CC(u, v)$  as  $\{cc_1, cc_2, \dots, cc_m\}$ . Then using the weights of the edges from the pair of parent nodes to the ‘common children’,  $W(u, cc_i)$  and  $W(v, cc_i)$ , we calculate the ‘Error

Ratio' vector,  $ER$ :

$$\overline{ER}(u, v) = \left[ \frac{W(u, cc_1)}{W(v, cc_1)}, \frac{W(u, cc_2)}{W(v, cc_2)}, \dots, \frac{W(u, cc_m)}{W(v, cc_m)} \right] \quad (3.3)$$

The selection between  $(u, v)$  and  $(v, u)$  for a numerator and a denominator results in identical results when observing  $|\log(\overline{ER}(u, v))|$  due to the properties of logarithms. Another important property of  $|\log(\overline{ER}(u, v))|$  is that for coordinates with a small relative error, the values of the elements in  $ER(u, v)$  will be rather small, and will increase with a loss of the accuracy. Therefore comparing  $er(u, v) = \text{median}(|\log(\overline{ER}(u, v))|)$  to a certain threshold gives a proper indication of the accuracy in the majority of measurements.

We use the  $er$  values for the parents, to partition parents groups into smaller parent groups which are geographically collocated. To this end, we produce a weighted clique of all the parent nodes in a group, where the weight of the edge  $(u, v)$  is  $er(u, v)$ . We remove all the links with a weight above a certain small threshold. Each connected component in the remaining graph becomes a parent group for the next step. To summarize, we partitioned the parent group to several parent groups that are geographically co-located.

The same process is repeated for child groups, where the error vectors are calculated by the distances to the common parents.

This kind of localization helps us to overcome a relatively large number of errors. However, if more than half of the measurements to a certain node are incorrect, the algorithm may fail to determine its location. Otherwise, there is no impact on the overall performance. Those 'badly' measured nodes might not become a part of the correct PoP, but the PoP map will be formed correctly in spite of them, i.e., no new PoPs will be created.

(c) *Unification.* Unifying *parent/child* group to the same PoP. If a *parent* and a *child* groups are connected, then the weighted distance between the groups is calculated (if they are connected, by definition more than one edge connects the two groups); if it is smaller than a certain threshold,  $PPC_{max\_th}$ , the pair of groups is declared as part of the same PoP.

### **Final Refinements.**

*Unification of loosely connected components.* In some cases, e.g., due to insufficient measurements, different parts of a PoP are only loosely connected in a way that does not form even a 2x2 bi-partite; in the extreme case only a single link connects two parts of a PoP. This will not allow the unification

process, just described above, to identify the parts as belonging to the same PoP. Thus, the algorithm looks for connected components (PoP candidates) that are connected by links whose median distance is very short (below  $PD_{max\_th}$ ). Note that at this point, due to the unification process, the graph has shrunk considerably, and thus the search for 'close' components is inexpensive.

(b) *Singleton Treatment* At the end of the process, the ISP graph has evolved through the multiple node unifications described above into a graph that is comprised of several multi-nodes (the PoPs) and a larger number of nodes (IP interfaces) that were not assigned to any PoP. Typically, these nodes have only one or two links connecting them to the rest of the graph, and the path from a node to the closest PoP is in most cases one hop and sometimes two. This final step assigns many of these nodes to existing PoPs. The assignment is conducted by running a Dijkstra shortest path algorithm from a node to all PoPs, and connecting a singleton to the closest PoP, providing the distance (in mSec) is below a given threshold  $PD_{max\_th}$ .

While this step has some advantages, it typically degrades the algorithm accuracy and does not add to the number of discovered PoPs. Therefore, unless noted differently, it is eliminated in most presented results. We discuss the effect of Singletons in Section 3.2.

### **3.2 PoP Extraction Validation**

Following, we present our validation tests and the results of a full implementation. The validation is then extended to discuss tradeoffs in the algorithm's implementation and their effect on result's accuracy.

Two collected datasets for PoP extraction are taken from DIMES [97]. One is from 2009, with a focus on weeks 27 to 30 for specific examples, and the other taken from weeks 42 to 43 of 2010. The database from weeks 27 to 30, 2009 includes 56 million traceroute measurements, collected by 1415 agents. The 2010 database, from weeks 42 to 43, has a total 33 million measurements, an average of 2.35 million measurements a day. The measurements were collected by 1308 agents, which were located in 49 countries around the world.

First, we examine the best time period length for collecting measurements for PoPs, and select it to be two weeks. DIMES produces five to six million daily measurements, both traceroute and ping, meaning thirty to forty million measurements per week, which typically result in 5.5M to 6.5M distinct

Compared Time Frame	#PoPs	#IPs in PoPs	#Distinct Edges
1 Week to 1 Week	< 1%	< 1%	±20%
1 Week to 2 Weeks	+58%	+79%	+43%
2 Weeks to 4 Weeks	+10%	+15%	+59%

**Table 3.2: Changes in PoP maps between different time frames**

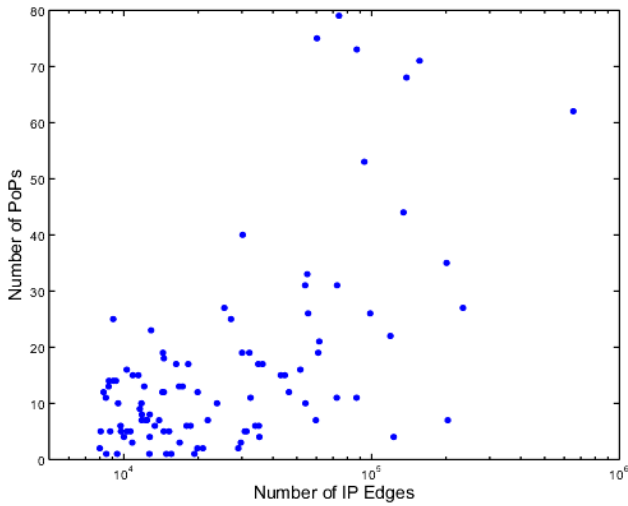
IP edges being discovered. The selection of a two weeks time period balances between two delicate tradeoffs: the number of distinct edges used for the PoP construction and the sensitivity to changes in the network. A time frame of a single week is too short, with considerably fewer distinct edges than those from two weeks. A month, on the other hand, does add many more edges, but it is insensitive to changes in the network, which we would like to track. In addition, the algorithm runs considerably slower on such large data sets. Table 3.2 shows the changes in PoP maps between different time frames. The first row in the table shows the difference in PoP maps between two consecutive weeks. The second row refers to a one week period compared to two weeks, and the last row compares two to four weeks measurements collection periods. The columns "#PoPs" and "#IPs in PoPs" refer to the change in number of discovered PoPs and IPs included in these discovered PoPs over the compared periods. "#Distinct Edges" refers to the change in distinct IP edges measured by DIMES. This number is independent of the PoP algorithm.

We set  $PM_{min\_th}$ , the minimal number of node's measurements, to be 5. This threshold was found to be optimal over many heuristic test cases, cleaning noisy measurements while filtering out only a small number of edges. We then ran the median algorithm described in [32] to find the delay between two adjacent nodes.

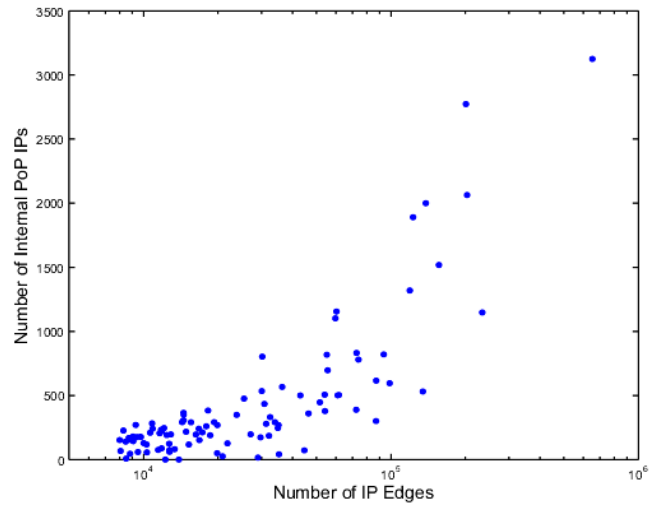
The resulting IP address to PoP mapping table typically consists of over 50,000 IP addresses, in about 4000 different PoPs. The average size of a PoP is 16 IP addresses, with a median of 6. The largest PoP size observed was 2500. The size of the discovered PoPs depend both on our measurement method and the ISP's policies. When a PoP is measured from many different agents or there are many paths between the source and destination nodes, the size of the PoP will be larger. However, measuring from one direction or if there is a relatively small number of alternative routes, the size of the discovered PoP will be small. The policies of the ISP can cause nodes inside the PoP to not answer traceroute messages and become anonymous or transparent e.g., due to use of MPLS.

On a single day, DIMES may run several experiments in parallel, however, the vast majority of the





**Figure 3.4: Number of Discovered PoPs vs. Number of measured IP Edges**

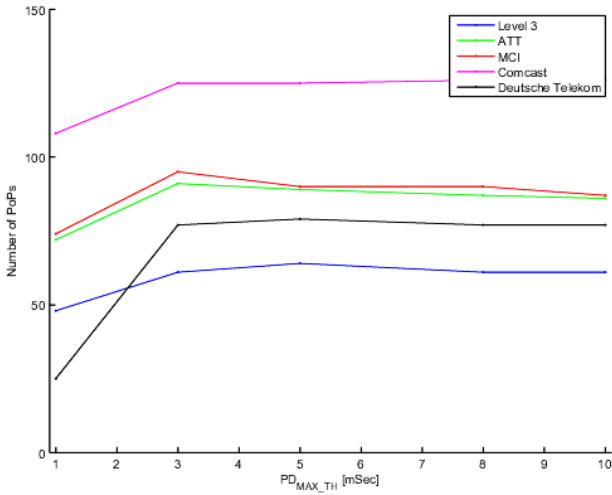


**Figure 3.5: Number of IPs in PoPs vs. Number of measured IP Edges**

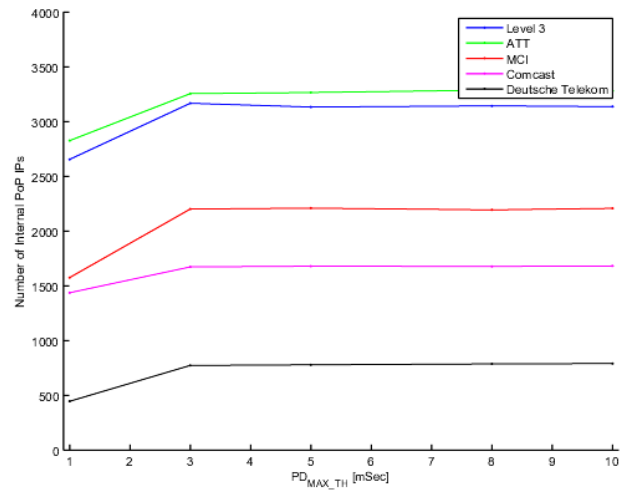
measurements performed over a week belong to the DIMES default experiment where a set of roughly 2.5 million target IP addresses, selected to cover all the allocated IP address prefixes, are cyclically sent to the agents. To test whether the target set limits us from discovering more PoPs, 2.5 million IP addresses were added to this basic experiment, identified by the iPlane project [73] as belonging to PoPs. The addition of the iPlane IP addresses increased the number of PoPs discovered by less than 20%, yet did not reach the numbers in iPlane. We believe that the immense number of IPs grouped by iPlane into PoPs partly represent IPs which are not part of the PoP.

The number of PoPs found in an AS network correlates with its measured size. Figure 3.4 shows that the number of PoPs discovered per AS depends logarithmically on the number of IP edges measured. Figure 3.5, showing the number of IPs included in PoPs compared to the number of IPs edges measured, demonstrates even better the logarithmic relation between the number of measurements and the discovered PoPs. As the number of IP edges reflects measurements through unique IPs and not PoPs, this is an expected outcome.

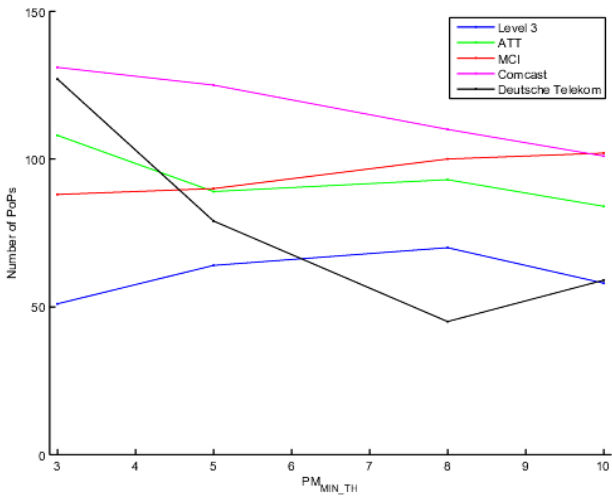
Figures 3.6 to 3.9 explore the PoP extraction algorithm's sensitivity to its two parameters  $PD_{max\_th}$  and  $PM_{min\_th}$ . In each figure five ISPs are explored: Level 3, ATT, Comcast, MCI, and Deutsche Telekom. In Figure 3.6 the number of discovered PoPs is compared with  $PD_{max\_th}$ , the maximal delay threshold. Figure 3.7 presents the number of IPs included in these PoPs under these conditions. Neither the number of discovered PoPs nor the number of IPs within the PoPs are sensitive to the delay threshold, as long as the threshold is  $3mS$  or above.  $PD_{max\_th}$  was therefore selected to be  $3mS$ ,



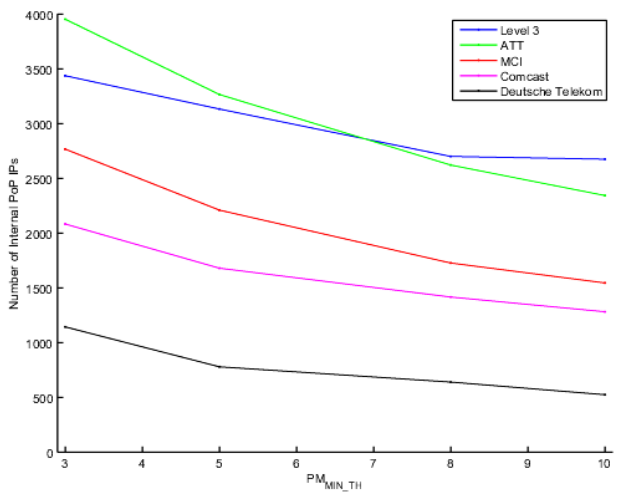
**Figure 3.6: Number of PoPs vs. Maximal Delay**



**Figure 3.7: Number of IPs in PoPs vs. Maximal Delay**



**Figure 3.8: Number of PoPs vs. Minimal Number of Measurements**



**Figure 3.9: Number of IPs in PoPs vs. Minimal Number of Measurements**

as it presents a good tradeoff between delay measurement error and location accuracy. Figures 3.8 and 3.9 show the effect of  $PM_{min\_th}$ , the minimal number of measurements threshold, on the number of discovered PoPs and the number of IPs included in them. The number of IPs included in PoPs clearly decreases as the minimal number of required measurements increases, as can be expected. The number of discovered PoPs shows a mixed behavior as the reduction of IP level links may have two conflicting outcomes; An increase is caused by a loss of connectivity inside a PoP which in turn causes it to split to several PoPs located at the same place, while a decrease is caused by the loss of the ability to identify a PoP. In our experiments,  $PM_{min\_th}$  was selected to be 5.

Additional validation tests repeatedly targeted previously identified PoP IP addresses within several large ASes, such as Level3, ATT and MCI, from agents within the AS. They did not increase the num-

ber of discovered PoPs, but proved that discovered PoPs are stable. To show that the PoP algorithm succeeds when enough measurements are provided, two ASes were taken as an example: GEANT, the pan-European academic network, and Proxad, a French ISP. Both were selected since their PoP topology is public and since DIMES did not have many measurements in them by default. Comparing the amount of PoPs and IPs within PoPs discovered based on default DIMES measurements and directed measurement tests, the number of discovered PoPs more than doubled and the number of IPs within PoPs grew by a factor of ten. In both cases, the directed tests doubled the number of distinct measured edges within the AS, thus increasing the connectivity required to discover PoPs. We conclude that increasing the number of measurements improves the algorithm's performance.

Other stability tests examined the IP addresses identified as part of PoPs and found 85% similarity between consecutive fortnights. The difference between PoPs was due to lack of measurements through the PoP connecting nodes, rather than the PoP extraction algorithm. In addition, not all the traceroutes are identical every week, due to the community based nature of DIMES. Additional validation actions taken are detailed in the next chapter. Validation of PoP maps was always an issue in related work, e.g., in iPlane [73] or RocketFuel [104], and we find that the level of validation introduced in this work is at least at the level of previous efforts.

## IV Assigning a Geographical Location to PoPs

### 4.1 Naive Geolocation Algorithm

Automatically assigning every discovered PoP to a geographical location is the second contribution of this work. The first mean for PoPs geolocation, referred to as *Naive Geolocation Algorithm* uses geolocation services in order to find the PoP's geographic coordinates. Geolocation services provide location information regarding a given IP address, including country, city, longitude and latitude.

We use several geolocation services to maximize the accuracy of our PoP location. The initial results from 2009 used MaxMind GeoIP [76], IPLigence [62], and Hostip.info [53]. Later results were extended to use also IP2Location DB5 [51] and GeoBytes [37]. Information from Netacuity [24], Spotter [70] and Neustar [5] (formerly Quova [4]) was used to some extent as well.

To identify the geographical location of a PoP, we use the geographic location of each of the IPs included in it. As all the PoP IP addresses should be located within the same campus, or within its vicinity if singletons are considered, the location confidence of a PoP is significantly higher than the confidence that can be gained from locating each of its IP addresses separately. The algorithm operates as follows:

**Initial Location** Each of the geolocation databases used is queried for the location (longitude, latitude) of each IP included in the PoP. Next, the center of weight of the PoP location is found by calculating the median of all PoP's IP locations. Unlike average calculation, where a single wrong IP can significantly deflect a location, the median provides a better suited starting point, but does not guarantee good results if there is complete disagreement between geolocation databases. We discuss this further in Section 4.2.

**Location Error Range** Every PoP location is assigned a range of convergence, representing the expected location error range based on the information received from the geolocation databases. For every IP address in a PoP and for every geolocation database we collect the geographic coordinates. Thus if there are  $N$  IP addresses and  $M$  databases, for each of the IP addresses we get at most (if all

are resolved)  $N \times M$  location votes. The algorithm finds the smallest radius which has at least 50% of the votes, with  $1km$  granularity. If the radius is above a given threshold, typically  $100km$  or  $500km$ , the algorithm outputs the threshold radius and the percentage of location votes within it. If one of the geolocation databases lacks information on an IP address, this IP element is not counted in the majority vote.

**Location Refinement** After a range of convergence is found, the PoP location accuracy is further improved. The new PoP location is set to the median of the location votes inside the range of convergence. This ensures that deviations in the PoP location caused by a small number of IP elements outside the range of convergence are discarded, and the PoP is centered based only on credible IP addresses locations.

To summarize, the PoP geolocation algorithm provides per PoP longitude, latitude, range of convergence and the percentage of location votes within its convergence range.

## **4.2 Validation of the Naive Geolocation Algorithm**

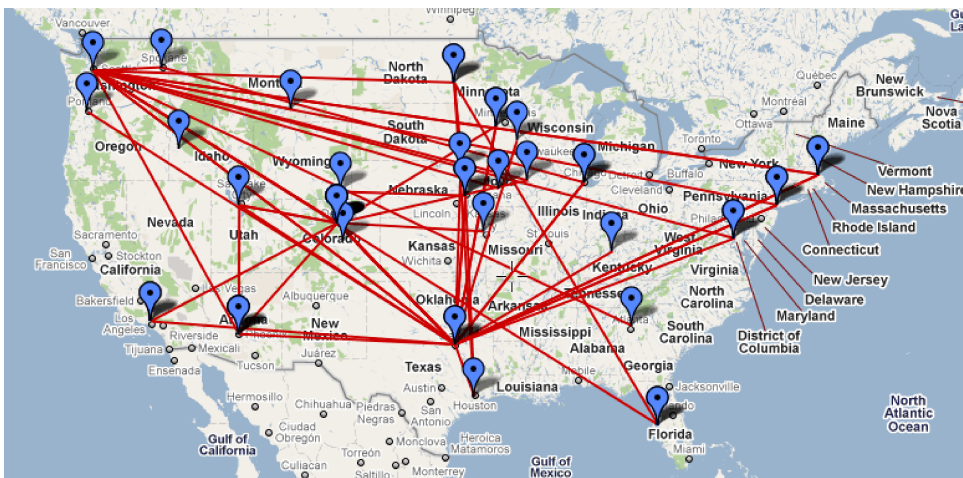
The geolocation algorithm has two interesting outcomes. First, it validates the PoP extraction algorithm by showing that PoPs are indeed scattered geographically, and locates points of presence around the globe. Second, it examines the quality of the geolocation services and finds their faults. This section discusses only PoPs location, and a survey of geolocation services is provided in section 4.3.

The algorithm converges successfully based on its validation's results. 70% percent of the PoPs have a range of convergence of ten kilometer or less. Although 89% of the PoPs have more than the minimal requirement of 50% of the IP location votes within the convergence range, for only 9.1% of the PoPs have over 90% of the location votes within the convergence range, indicating inaccuracies in some of them. To strengthen this point, when requiring the PoP location to be agreed upon by any three geolocation databases instead of five, over 90% of the PoPs converge within ten kilometers range, which comes to show that the disagreement between the geolocation database is the cause to the above.

Figure 4.1 shows the discovered PoPs located on a world map. Clearly, the US and Europe have very good coverage. In East Asia many PoPs are discovered as well, but only a few are found in South



**Figure 4.1: PoPs World Map**



**Figure 4.2: QWEST US PoP Map**

America and Africa.

We then proceed and generate a PoP location map per Internet service provider. The maps display the PoPs of all the ASes residing under the same provider (sibling ASes), to provide a full picture of the vendor’s network. The provider maps also show the connectivity between the different PoPs, as measured by DIMES. Figure 4.2 shows as an example provider map of Qwest with its internal network connectivity.

To validate our generated maps we compare them against the PoP maps published by the ISP, such as Sprint [106], Qwest [87], Global Crossing [39], British Telecom [16], ATT [10] and others. The PoP algorithm detects most of the large points of presence, but it detects very few small, local PoPs. There are several explanations for this behavior. First, we measure mainly to and through nodes that pass a lot of traffic, and filter out edges that were hardly measured, in order to filter out noise. Even when we add the PoP IPs discovered by iPlane, most of these small PoPs are still not found. This leads

us to the second reason some PoPs are not discovered: due to security reasons, many routers do not answer traceroute ICMP packets, which reduces the algorithm's ability to discover the PoP structure. Last, some of the vendors employ encapsulating protocols such as MPLS, which may hide most of the routing path. Luckily, as our results show, these protocols are not deployed widely enough to harm our measurements.

As another method of validation, fifty PoPs that belong to universities around the globe were selected, and the location given to them by the algorithm was compared against the institute's actual location. For 49 out of 50 universities, the location was accurate within a 10 kilometers radius. The last PoP, belonging to The University of Pisa, was located by the algorithm in Rome instead (330km away), due to an inaccuracy in the MaxMind and IPLigence databases. Only Hostip.info provided the right coordinates for this PoP. Each PoP location was also validated against its DNS name, yet many interfaces had no DNS name assigned to them.

We compare our PoP geolocation also for GARR, the Italian research network. In weeks 42-43, 2010 we found eight PoPs in GARR, containing 99 IP addresses. GARR has a total of fifty eight PoPs in Italy; however in several cases a few PoPs are located in a small area. For example, there are eight PoPs in Milan's area, and six in Florence's vicinity. Our extraction algorithm thus merges such PoPs into a single entity. Checking the assignment of PoPs to locations, based on DNS, information provided on GARR's website [36] and information from users, we successfully geolocate five of the PoPs in their correct location based on 100% of the IP locations. In two PoPs, the PoP is located correctly, however it seems to include a single IP address which is supposed to reside in a different location. In both cases we observe that the edge delay to other IP addresses included in this PoP is less than 2mS. For the last PoP, the PoP is located correctly in Milan, however it includes several IP addresses that are supposedly part of different PoPs. We note that the geolocation databases are also missing information for many of these IP addresses - only 55% of the IPs which are part of the PoP have location information, and the agreement level that we assign for the PoP is low as well: 66%.

For less than 10% of the PoPs we fail to find the location with high confidence using five geolocation databases. In almost all these PoPs the cause is lack of location information in the databases, mostly in HostIP.Info, GeoBytes and MaxMind (MaxMind provides country level information). When a majority is requested only amongst three databases, more than 99% of the PoPs are located with high confidence. When IP location information is available, the main cause of PoP location failure is due

to disagreement between the location services. To summarize, while in some cases the disagreement is a result of incorrectly estimated links, as suggested in 3.1, the majority is caused by geolocation database inaccuracies.

### **4.3 A Survey of Geolocation Databases**

Relying on geolocation databases for PoP's naive geolocation algorithm requires a high level of database's accuracy. In the following chapter we survey a large number of geolocation services, of various kinds, in order to assess their accuracy.

#### *4.3.1 Background*

In the recent years, geolocation services have become a necessity in many fields and for many applications. While the end user is usually not aware of it, many websites visited everyday use geolocation information for targeted localized advertising, localized content (such as local news and weather), and compliance with local law.

The last decade presented a new threat to the world: cyber terrorism. Cyber terrorism and warfare targets communication networks as well as important infrastructure facilities, and thus threatens to cause havoc through online attacks. Finding and blocking such cyber attacks is in a high priority for national security forces, and IP geolocation can help by providing geographic information about the attacker hosts. The DHS cyber security center [52] classified geolocation research to be in the field of situational understanding and attack attribution, with the intent to identify attackers. The DHS also comments that geolocation improves visualization, thus simplifies large-scale data analysis. A patent filed by the NSA [56] notes that geolocation can be used to monitor remote access and prevent login using stolen passwords or login ID. It can only be speculated that military and government based agencies use geolocation techniques to detect the source of activity on terrorist related websites as well as trying to track down enemy communication centers.

Perhaps the most highlighted purpose of geolocation information is for fraud prevention and various means of security. Banking, trading, and almost any other type of business that handles online money transactions are exposed to phishing attempts as well as other schemes. Criminals try to break into user accounts to transfer money, manipulate stocks, make purchases and other illegal activities. Ge-



olocation information provides means to reduce the risk, for example by blocking users from certain high-risk countries and cross-referencing user expected and actual locations.

The IETF has also commenced in defining standards for geolocation and emergency calling through IETF GEOPRIV working group [60], which discusses internet geolocation standards and privacy protection for geolocation. Some examples are DHCP location, as in RFC3825 and RFC4776, and defining protocols for discovering the local location information server [112]. Even common emergency services, such as dispatching emergency responders to the location of emergency use it.

Geolocation information is also important in many research fields. It improves internet mapping and characterization, as it ties the internet graph to actual node positions, and allows exploring new aspects of the network that are otherwise uncovered, such as the effect of ISP location on its services and types of relationships with other service providers.

Many previous papers from various fields have discussed the usage of geolocation information in day-to-day applications ([108, 41, 65] and more). However, not many works have focused on the accuracy of geolocation databases. We survey these works as well as the improvement of location accuracy by measurements in Section 2.2

In this chapter, we study the accuracy of geolocation databases. The main problem in such a study is the lack of ground truth information, namely a large and diverse set of IP addresses with known geographic location to compare against the geolocation databases. We avoid this need by mapping IP addresses to PoPs. The PoP maps, based both on delay measurements and graph structure, have a very small probability of mapping two IP addresses which are not co-located to the same PoP. Thus, while we do not know the location of the PoP we know that all the IP addresses within a PoP should reside in the same location. This serves as a mean to check a geolocation database coherency: if two IP addresses in the same PoP are mapped to different locations, there is a database problem, and we can use the distances among the various locations of IP addresses in the same PoP as a measure of database accuracy. The results are presented in Section 4.3.4.1. We take a step further and compare multiple databases results for the same PoP (Section 4.3.4.3) and study their spread.

### 4.3.2 Geolocation Services

Geolocation services range from free services, through services that cost a few hundreds of dollars and up to services that cost tens of thousands of dollars a year. This section surveys most of these services, focusing on the main players.

Free geolocation services differ from one another in nature. Three representatives of such sources are discussed here: DNS resolution, Google Gears and HostIP.Info. DNS resolution was probably the first source for geolocation information. In 2002 Spring *et al.* [104] used DNS names to improve location information as part of the Rocketfuel project. However, DNS suffers from several problems: many interfaces do not have a DNS name assigned to them, and incorrect locations are inferred when interfaces are misnamed [130]. In addition, rules for inferring the locations of all DNS names do not exist, and require some manual adjustments. Google Gears provides geolocation API [42] that can be used to query a user's current position. The position is obtained from onboard sources, such as GPS, a network location service, or from the user's manual input. When needed, the location API also has the ability to send various signals that the device receives (from nearby cell sites, WiFi nodes, etc.) to a third-party location service provider, who resolves the signals into a location estimate [43]. Thus, the service granularity is based on a single IP address granularity and not on address blocks. HostIP.Info [53] is an open source project. The data is collected from users participating in direct feedback through the API, as well as ISP's feedback. In addition, website visitors are updating their location, which in turn is updated as a database entry. The city data comes from various sources, such as data donation and US census data (for the USA). The data is provided as /24 CIDR blocks.

Another type of geolocation services emerges from universities and research institutes. These services tend to use measurements alone or combined with other methodologies to improve geolocation data quality. While many of the measurement based geolocation services that we discussed in Section 2.2 do not provide the ability to query specific IP addresses [63, 126, 129], one online geolocation service that does allow it is Spotter, which is based on a work by Laki *et al.* [70]. Spotter uses a probabilistic geolocation approach, which is based on a statistical analysis of the relationship between network delay and geographic distance. This approach is shown to be independent of the landmark's position from where the measurement was performed. To approximate the location of a target, spotter measures propagation delays from landmarks to the target, and then convert the delays into geographic distances based on a delay-distance model. The resulting set of distance constraints is used to deter-

mine the target's estimated location with a triangulation-like method.

Mid-range cost geolocation services include databases such as Maxmind GeoIP, IPLigence, and IP2Location. All these databases cost a few hundreds of US Dollars and supply to the user a full database, typically as a flat file or MySQL dump. Some of the companies, such as MaxMind, also provide a geolocation web service.

MaxMind [76] is one of the pioneers in geolocation, founded in 2002, and it distributes a range of databases: from country level to city level, longitude and latitude. Information on ISP and net speed can be retrieved as well. In addition to all the above, MaxMind offers to enterprises a database with an accuracy radius for geolocation information. In this work, the MaxMind GeoIP City database is being used for geolocation information. IPInfoDB [3] is a free geolocation service that uses MaxMind GeoIP lite database and adds on top of it reserved addresses and optional timezone.

IPLigence [62] is a geolocation service provider, existing since 2006. The company's high end product, IPLigence Max, contains geographic information such as country, region and city, longitude and latitude, in addition to general information such as owner and timezone. Hexasoft Development maintains IP2Location [51], a geolocation database with a wide range of geolocation information; from IP to country conversion, to retrieving information such as bandwidth and weather. For this study, we used their DB5 database, which maps IP addresses to country, region, city, latitude, and longitude. In all the above products, the IP addresses' location is given in ranges, which vary in size and reach the granularity of a handful of addresses per range.

High end geolocation services are often priced by the number of queries and their cost may reach tens of thousands of dollars a year for large websites. Amongst these services, and based on their pricing level, are Quova, Akamai Edge Platform [8], Digital Element's Netacuity Edge and Geobytes. Each of these companies pride themselves of large tier-1 customers from different fields, who use their services for targeted advertising, fraud prevention, and more.

Quova [4], founded in 1999, sells three levels of data information, bronze, silver, and gold. The advanced services contain attributes such as location confidence level, Designated Market Area (DMA), and status designations for anonymized Internet connections. Quova's database is based on data mining classification techniques, hand-labeled hostnames and research note.

Akamai [8] was founded in 1998 and launched its commercial service in 1999. It provides IP location

information through Edge Platform product. Akamai's IP location services are a part of a much larger package of tools and applications used for traffic management, dynamic sites accelerations, performance enhancement and more.

Digital Element [24], founded in 2005, publishes two levels of geolocation information under the products NetAcuity and NetAcuity Edge, with over thirty nine data points, including demographics, postal code, and business type. The IP geolocation data source is anonymous data gathered from interactions with users. One source for this user information is partner companies that use the product. The information is validated using a proprietary clustering analysis algorithm.

Geobytes [37] launched its GeoSelect product, for geolocation information, in 2002. The extent of data provided by Geobytes matches mid-range companies, but it is part of a broader package of services, including reports, users redirection, etc. While in the past Geobytes used ICMP packets to create an infrastructure map, current methods also include gathering information from websites that require users to enter their location information and then processing this data onto Geobytes' infrastructure map of the Internet [78]. No DNS information is used by Geobytes for their location resolution.

In this work, databases from all three groups are being used. From the no-charge databases: HostIP.Info, Spotter and DNS (partial). Mid-range databases used are MaxMind GeoIP City, IPLigence Max, and IP2Location DB5. GeoBytes and NetAcuity are the last two databases used in this work. Unfortunately, we failed to reach a collaboration with Quova and Akamai for this project.

#### *4.3.2.1 Databases Accuracy*

The geolocation service provider is, in many cases, the sole source for database accuracy information. Some vendors, such as IPLigence, do not publish such figures at all, while others announce precision figures without explaining how they were obtained. A few geolocation services, such as Akamai and Quova provide accuracy information as obtained by external auditors. Table 4.1 presents a summary of accuracy figures, as given by the geolocation service providers on their websites [4, 8, 24, 37, 51, 76]. The table includes information on country level, city level worldwide level and the USA city level accuracy.

---

<sup>2</sup>US State level accuracy

Database	Country Level	City Level	USA City Level
IP2Location	99%	80%	
MaxMind	99.8%	Varies	83%
GeoBytes	97%	85%	
NetAcuity	99.9%	95%	
Akamai		97.22%	100%
Quova	99.9%		97.2% <sup>2</sup>

**Table 4.1: Geolocation Database Accuracy As Reported By Vendor**

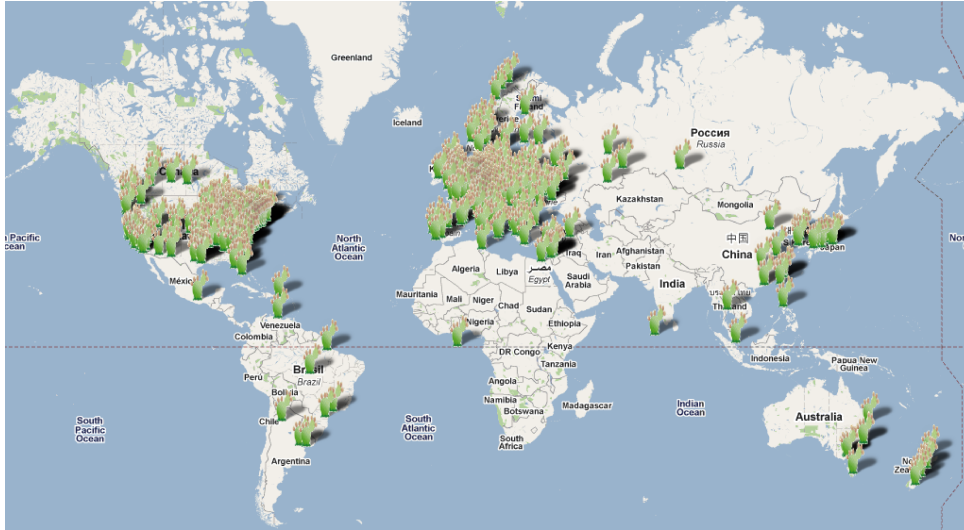
All the databases claim to have 97% precision or more at the country level and 80% or more at the city level. MaxMind publishes detailed expected accuracy on city level based on country [75]. The published figures range from 40% – 44% in countries like Nigeria and Tunisia to 94% – 95% in countries like Georgia, Qatar and Singapore. A correct location resolution here is considered to be within 25 miles from its true location. Netacuity’s information is based on a test by Keynote Systems. Quova’s precision results are based on an audit by PriceWaterhouseCoopers [86], which used 3 reference third party databases.

The accuracy of the figures in Table 4.1 cannot be easily evaluated. For example, neither the means by which Keynote Systems tested Netacuity nor the reference databases used to test Quova are revealed. Akamai claims for 97.2% correct resolution at the city level worldwide and 100% accuracy at the city level in the USA. The source for Akamai’s figures is a report by Gomez [40], which defined a node location to be unique on /23 CIDR subnets. In addition, a Census Metropolitan Area (CMA) is the basis of the naming convention used by Gomez to identify the physical location of its measurement nodes. The precision of this method is thus debatable, as described in Sec. 2.2.

### 4.3.3 The Evaluation Model

#### 4.3.3.1 Data Evaluation Method

The geolocation databases evaluation is conducted using the classification of IP addresses into PoPs as described in the previous chapters. Since the classification is based on both structure and delay measurements, the chances that two IP addresses, which our algorithm maps to the same PoP, are not located in the same geographical location are slim. We do recognize that when two PoPs are very close (within a few tens of kilometers) our algorithm may unify them to one. However, in this case the median of their location is half their distance, namely not far.



**Figure 4.3: Map Of DIMES Agents, March-2010**

To identify the geographical location of a PoP, we use the naive PoPs geolocation algorithm. The extraction of PoPs and assignment to geolocation based on active measurements requires careful data filtering. To this end, our PoPs extraction algorithm takes several precautions. First, at least  $PM_{min\_th}$  measurements are required per IP level edge in order for it to be considered by the PoP extraction algorithm, and a median algorithm [33] is applied in order to reduce the delay measurement error. Second, the distribution of the DIMES vantage points results in the measurement of an IP edge to be done by different agents from different locations, thus reducing the inherited measurement error of a specific path. Last, when DIMES measures a certain path, it sends four consecutive traceroutes per destination, and the best time is used. If a path has several alternate routes all the hops from the first divergence point are removed from the dataset.

#### 4.3.3.2 Dataset

The collected dataset for PoP level maps is taken from DIMES [25]. We use all traceroute measurements taken during March 2010, totaling 126.7 million, which is an average of 4.2 million measurements a day. The measurements were collected from over 1750 vantage points, which are located in 74 countries around the world, as shown in Figure 4.3. About 16% of the vantage points are mobile.

The 126.7 million measurements produced 7.85 million distinct IP level edges (no IP level aliasing was performed). Out of these, 642K edges had less than the median delay threshold, and had sufficient number of measurements to be considered by the PoP extraction algorithm. As described above, two



**Figure 4.4: Map Of Discovered PoPs, March-2010**

PoP level maps were generated by the PoP extraction algorithm, with and without singletons added. A total of 3800 PoPs were discovered, containing 52K IP addresses from the first run, and 104K IP addresses from the second run with singletons. Although the number of discovered PoPs is not large, as the algorithm currently tends to discover mainly large PoPs while missing many access PoPs, the large number of IP addresses and the spread around the world (see below) allow a large scale and meaningful geolocation databases evaluation.

Figure 4.4 shows the geographical location (as calculated by our algorithm) of the PoPs discovered by the PoP algorithm. The PoPs are spread all over the world, in all five continents, with high density of PoPs in Europe and North America. As can be seen, PoPs are located even in places such as Madagascar and Papua New Guinea, which comes to show the vast range of location information required from the geolocation databases in this evaluation.

For most of the databases, the data which was used, was updated on the first week of April 2010. NetAcuity database was obtained on the third week of April and Spotter located the IP addresses during April and the beginning of May 2010.

#### 4.3.4 Results

##### 4.3.4.1 Basic Tests

**Null Replies** We first check the number of NULL replies returned for IP address queries. There are four flavors to this question. First we distinguish between IP addresses in the core of the PoPs and the

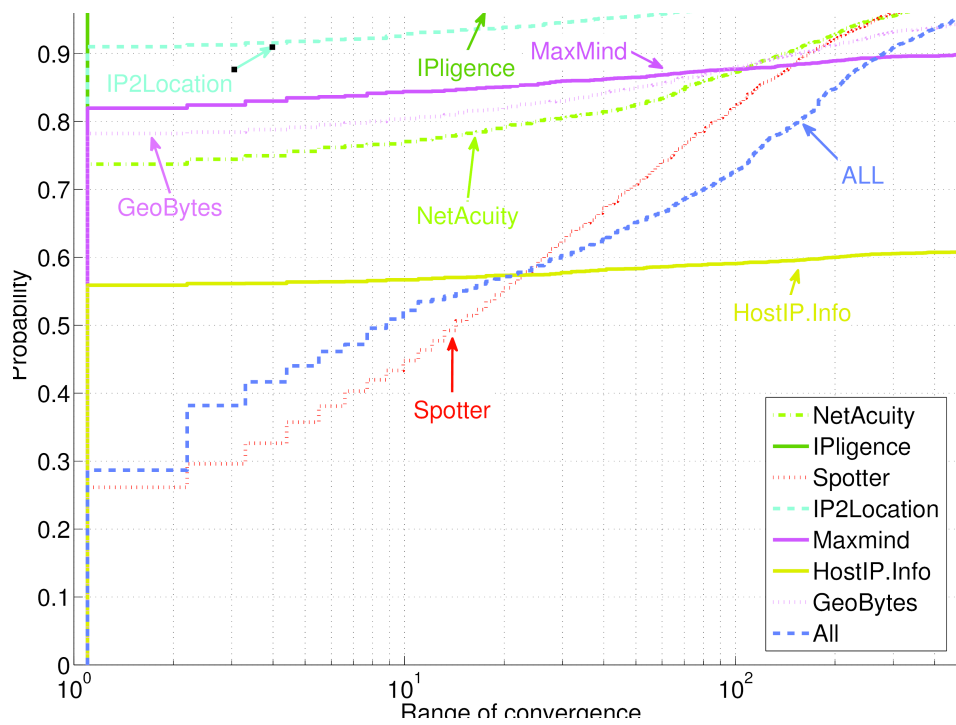
Database	Core PoP IP		With Singletons	
	Null IP	Null PoP	Null IP	Null PoP
IPligence	3.9%	1.5%	2.9%	1.4%
IP2Location	0%	0%	0%	0%
MaxMind	36%	10.6%	30.1%	6%
HostIP.Info	64%	38.6%	64%	29%
GeoBytes	20.7%	4.3%	17.8%	2.7%
NetAcuity	0%	0%	0%	0%
Spotter	37%	18.1%		
DNS	14.3%	12.2%	28.4%	2%

**Table 4.2: Null IP Address Information**

ones in singletons. As some databases may have better information on end users or access interfaces than on core routers and main PoPs, this can be meaningful. The next observation regards NULL replies that apply to all the IP addresses within a certain PoP: does the database fail to cover a range of addresses or a physical location range, or are the NULL replies a matter of a single IP address lack of information? This is considered both with and without singletons. Table 4.2 shows for each of the databases the percentage of IP addresses which returned a NULL reply for each of these cases.

NetAcuity and IP2Location were the only databases to return a location for all the queried IP addresses. This alone does not indicate that the returned addresses are correct, only that an entry exists. On the other end of the scale, HostIP.Info failed to locate most of the IP addresses, however on the PoP level this percentage drops by half. This may imply that the nature of the failure for HostIP.Info is the lack of information on specific IP addresses and not IP ranges. Furthermore, in most cases HostIP.Info did return a reply with country information, but without longitude and latitude. Spotter did not locate about a third of the IP addresses. The reason for such a failure can either be that the IP did not respond to ping or that the roundtrip delays were too high to provide approximations for the algorithm. Only core PoP IP addresses, without singletons, were tested here. For MaxMind, the percentage of Null replies refers to events where no specific location information was available. In most of these cases, MaxMind did return longitude and latitude information, which represent the center of the country where the IP used. DNS NULL replies were less than 15% for core PoP IP addresses, and almost 29% when taking into account singletons. As there is a probability that singletons represent end users and not router interfaces, this is expected. The effect of grouping into PoPs when looking at DNS is significant: when taking into account singletons, only 2% of the PoPs have no DNS-based location information.

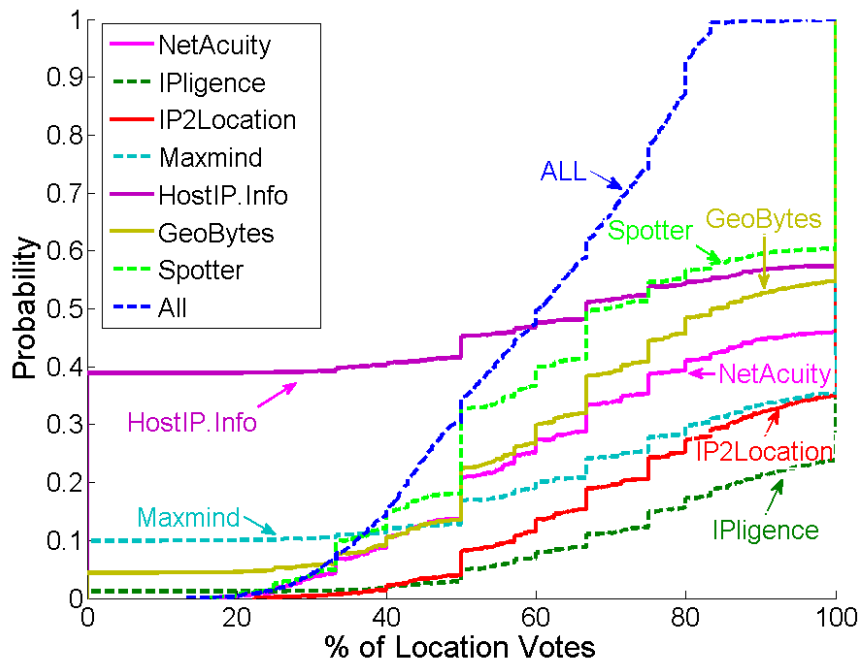




**Figure 4.5: Range of Convergence Within Databases**

**Agreement within database** By definition, IP addresses belonging to the same PoP reside in the same area. One can leverage this information to evaluate the accuracy of a geolocation database: If IP addresses that belong to the same PoP are assigned different geographical location, then the accuracy of this information should be questioned. This statement is based on the assumption that the PoP algorithm is correct and does not assign IP addresses from different locations to the same PoP. Our experiments here further support this assumption: In all the PoPs evaluated, with no exception, there are always databases that support the PoP vicinity requirement.

We run the algorithm separately for each database. Figure 4.5 presents a CDF of the convergence range within each database without singletons, with the X-axis being the range of convergence in kilometers. The convergence range is the radius which covers at least 50% of the IP addresses locations within a PoP. IPelligence and IP2Location clearly have a range of convergence far better than other databases: over 90% of the PoPs located using these databases have the minimal range of convergence, one kilometer, which is in practice the exact same location. MaxMind, GeoBytes and NetAcuity have 74% to 82% of their PoPs converge within one kilometer. For HostIP.Info, a bit less than 57% of the PoPs converge within the minimal range, and almost all the rest fail to converge. This is caused mostly due to lack of information on IP addresses, as many PoPs do not have even a single IP with location information. Spotter is different than the others. As Spotter information is acquired by measurements having almost a third of the PoPs converge within one kilometer is an indication

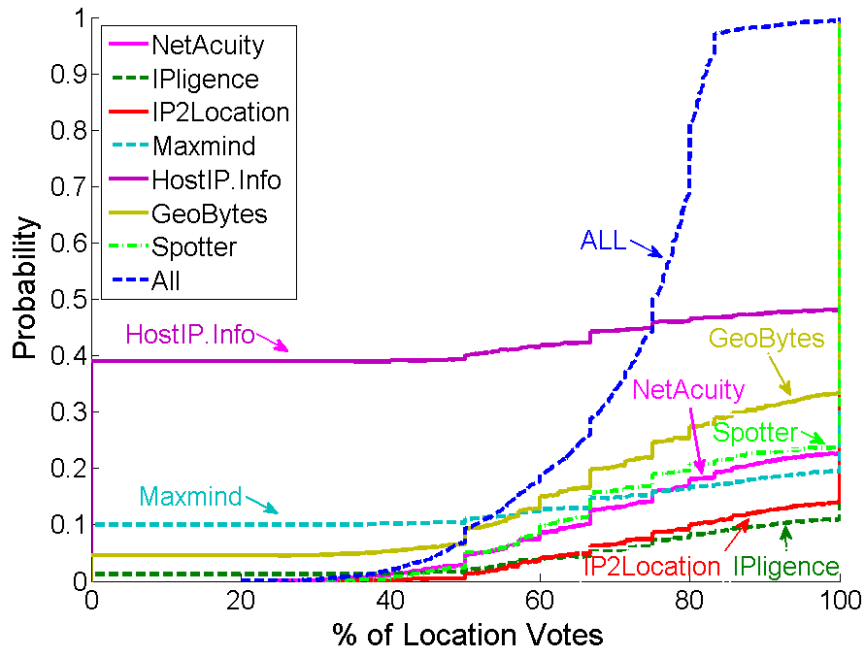


**Figure 4.6: CDF of Location Votes Percentage Within 100km From PoP Center**

of good performance. In addition, over 82% of the PoPs converge within 100km, and close to 98% within 500km, which is similar or better than most of the other databases. The slow accumulation is expected due to measurements errors. An interesting result is the curve marked as *All*, showing the range of convergence when combining the information from all databases. Though all databases have most of their PoPs located within the minimal range, less than 30% of the *All* PoPs converge within this range, meaning that there is a disagreement between the databases, though as the range grows so does the percentage of converged PoPs. This does not necessarily mean that all the databases have agreed on the same location, nor that this location is correct, as databases which reply with a location for every IP have more influence than databases with some NULL replies. We further explore this question in Section 4.3.4.3.

Figures 4.6 and 4.7 present a CDF of the agreement within databases without singletons. The X axis marks the percentage of IP addresses in PoPs that represent the majority, and the Y axis presents the probability for this majority vote. For Figure 4.6 we set a radius of 100km and in Figure 4.7 the used radius is 500km, within which a majority is required. Most databases have 95% or more chances to have at least 50% of the location votes within the 500km radius, and at least 90% within 100km radius.

For all databases there were PoPs with no majority vote, meaning that less than 50% of the location



**Figure 4.7: CDF of Location Votes Percentage Within 500km From PoP Center**

votes were within the tested radius. IP2Location and IP2Location have the highest probability to reach a majority vote, while HostIP.Info, and Geobytes grow at the slowest pace. For a radius of 100km, Spotter does not reach full agreement for almost 60% of the PoPs, probably due to measurement accuracy limitations. Interestingly, for less than 4% of the PoPs there is 100% agreement by all databases, which once again does not correlate with single-database observations and points to a mismatch between databases.

#### 4.3.4.2 Ground Truth Location

The best way to assess the accuracy of a database is to compare its information with the true geographical location of each IP address, through some "Ground Truth" database. Unfortunately, there is no ground truth database of all IP addresses. A small number of IP addresses are covered by a ground truth database provided by CAIDA. The database, described in [54], includes private data from one tier-1 and one tier-2 ISPs. In addition it contains public data from five research networks. The geographic location is provided based on host names, with their encoding provided by the ISP and verified.

We use this database to evaluate the accuracy of the geolocation databases. The ground truth database used is from January, 2010 and includes 25K IP addresses, their ISO code, country, region and city.

Database	Matched IPs	Country Match	City Match
GeoBytes	67.3%	80.1%	26.5%
HostIP.Info	28.1%	89.0%	17.9%
IP2Location	93.9%	80.9%	14.16%
IPligence	93.9%	81.0%	0.8%
MaxMind	79.6%	84.7%	29.4%
NetAcuity	67.9%	96.9%	79.1%
Spotter	54.1%	85.6%	27.8%

**Table 4.3: Comparison with Caida’s Ground Truth Database**

Each database is compared with the ground truth dataset to the maximal extent. For databases where only PoP-IP data is available (Spotter, NetAcuity), CIDR/24 is used to match missing addresses. We note that despite this extended match, our database is still too limited to match all. Before describing the results, it is important to understand that this group of IP addresses is not necessarily representative, which may bias the results.

Table 4.3 presents the results of the comparison. The column "Matched IPs" presents the percentage of IP addresses matched between the ground truth and the evaluated database, and returned with a non-NULL value. Out of the matched IP addresses, "Country Match" presents the percentage of matches on country level and "City Match" presents the percentage of matches on city level. We allow a distance of 100km between a pair of returned city coordinates to consider a reply as a city match.

It is interesting to observe that, with the exception of NetAcuity, none of the databases is close to its acclaimed accuracy on the country level. In most cases, the databases indicate that the IP is located in United States, while the ground truth database places them elsewhere. For IPligence and IP2Location 99% of the wrong placements are of this type, and 88% to 90% of the mistakes for MaxMind and HostIP.Info. Geobytes, on the other hand, has an almost equal number of mismatches between the USA and other countries, with no dominant trend. An expected mistake, common to IPligence and IP2location, is the interchange between the USA and Canada.

On the city level, IPligence’s and IP2Location’s results are remarkably poor. The reason that we observe is the large amount of IP addresses assigned to Washington DC by both databases: IPligence assigns no less than 20.4K of the mismatched IP addresses to Washington, while IPligence does so for 10.1K of the IPs. This phenomenon is not evident in other databases, where the results tend to spread across cities. Other cases of a large bias for a city are Geobytes, with 3.8K of the wrong assignments

---

<sup>2</sup>US State level accuracy

set to New-York City, and MaxMind with 4.8K of the wrong assignments set to Washington DC. Spotter measurements accuracy affects these results, too, which is evident when increasing the match range from 100km to 300km: the number of matched cities to longitude/latitude doubles.

#### 4.3.4.3 *Comparison Between Databases*

While some of the databases have proprietary means to gather location information, a large portion of geolocation data is likely to come from the same source, such as getting country information from ARIN. To examine this we use the 52K PoP IP addresses that are used in the first evaluation as a sample of the IP space. We calculate for each pair of databases a distance vector that holds the distance difference between their answers to the geolocation of each of the IPs in the list. Cases where at least one of the databases returned a NULL answer were ignored. Figures 4.8 and 4.9 depict the root mean square (RMS) of each distance vector and its median, respectively. The fairly large RMS values are due to the long fat tail of large errors in the databases, which we discuss later in this section. Maxmind, IP2Location, and IPLigence seem to be much closer to each other than the rest. This is evident in the median heatmap that shows a median distance of only 5-11km. Netacuity has a small median distance to Maxmind, but much larger one to the other two. This is due to the many country level values returned by Maxmind and ignored by this analysis; naturally these are the IPs which are harder to locate and thus most databases have their acute errors in this group. As a result Maxmind seems closer to databases more than others.

The large values in the RMS distances heatmap (Figure 4.8) are explained by Figure 4.10. The figure presents the CDF of distance vectors for several selected database pairs. The pairs that had very small median distance, such as IP2Location to MaxMind and IPLigence or Maxmind to Netacuity, grow at a very fast rate until a probability of about 0.6. This leads to a median that is only a few kilometers. However, about 10% of the IP-distances will be between 500km to 1000km range. Some of the addresses are even located very far away, thousands of kilometers apart. We assume that most of these differences are caused by anomalies in at least one of the databases. Databases with high median and RMS distance have the same trend of CDF as the other pairs, however the main difference is that the initial distance between most IP addresses is larger: For Geobytes to HostIP.Info, only 30% of the IP addresses are located within a close range, while 20% more are within 500km to 1000km range. Note that here the tail of CDF distance values is even longer than in the previous pairs.

	MaxMind	IP2Location	IPligence	HostIP	NetAcuity	Spotter	GeoBytes
MaxMind		979	1201	2037	2457	2808	3108
IP2Location	979		1310	2509	2842	3139	3253
IPligence	1201	1310		2396	2895	3162	3386
HostIP	2037	2509	2396		3072	3079	3543
NetAcuity	2457	2842	2895	3072		1862	3839
Spotter	2808	3139	3162	3079	1862		3850
GeoBytes	3108	3253	3386	3543	3839	3850	

**Figure 4.8: RMS Distance[km] Between Databases - Heatmap**

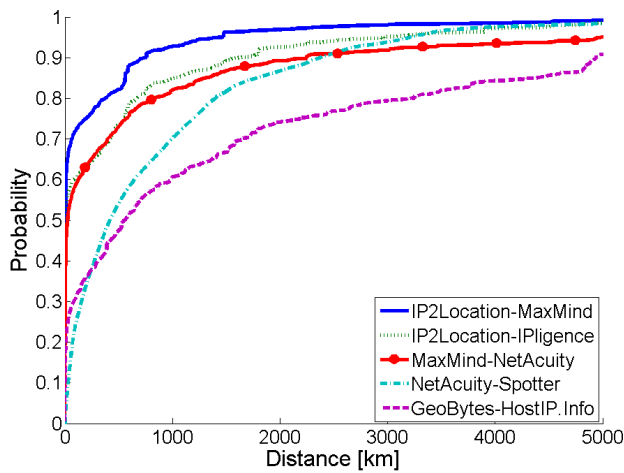
	MaxMind	IP2Location	IPligence	HostIP	NetAcuity	Spotter	GeoBytes
MaxMind		5	6	158	24	666	216
IP2Location	5		11	382	171	836	48
IPligence	6	11		452	148	821	433
HostIP	158	382	452		264	890	528
NetAcuity	24	171	148	264		392	418
Spotter	666	836	821	890	392		1053
GeoBytes	216	48	433	528	418	1053	

**Figure 4.9: Median Distance[km] Between Databases - Heatmap**

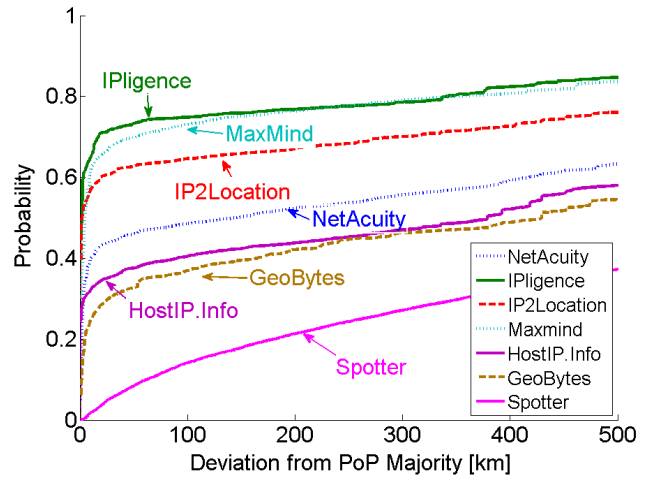
NetAcuity to Spotter pair, selected for their relatively close median value, demonstrate the effect of active measurements: the CDF curve is smooth, and there are almost no IP addresses placed within a few kilometers range. The cause is that while most databases who place an IP address within the same city will give it identical coordinates, like city center, while Spotter will triangulate the location within the city premises.

Next, we compare the databases based on aggregated data collected from all sources and look at the distance of each IP from the PoP median location. Note that due to the high correlation between 3-4 databases, that may be the result of using similar location sources, the PoP median location may be shifted and not always correct.

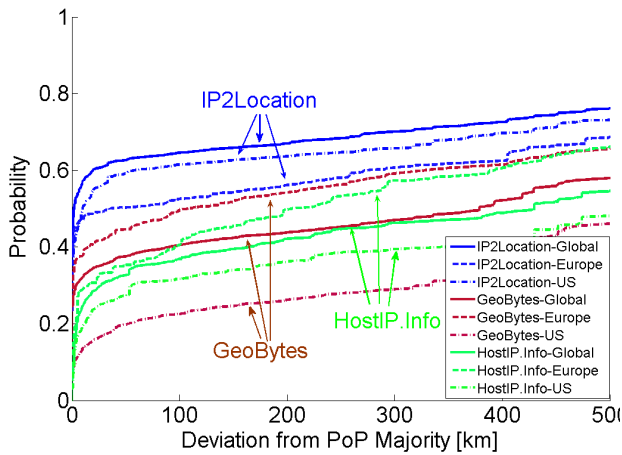
We assess the similarity between databases by comparing an IP location in every database to the location of its PoP as voted by all databases. Figure 4.11 depicts the CDF of the deviation of each IP from the PoP median location for each database. The interesting observations here are at the



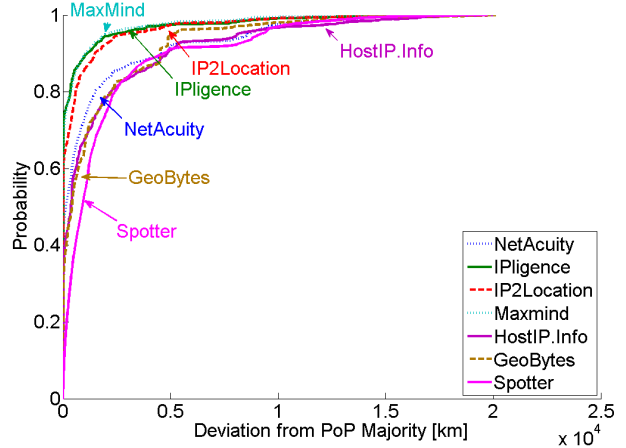
**Figure 4.10: CDF of Distances[km] Between Databases**



**Figure 4.11: CDF of Database Location Deviation From PoP Median - 500km Range**



**Figure 4.12: Breakdown of deviation from PoP majority CDF By region - 500km Range**



**Figure 4.13: CDF of Database location deviation from PoP majority**

40km range, which is a city range, and 500km range, which can be referred to as a region. IPLigence, MaxMind and IP2Location have a probability of 62% to 73% to place an IP within 40km from the PoP median location, with IPLigence and MaxMind placing over 80% of the IP addresses within 500km radius. Geobytes, HostIP.Info and Netacuity place 33% to 47% of the IP addresses within a city range, and 48% to almost 60% within 500km from the majority vote. Spotter places only 10% within 40km range and 30% within the same region.

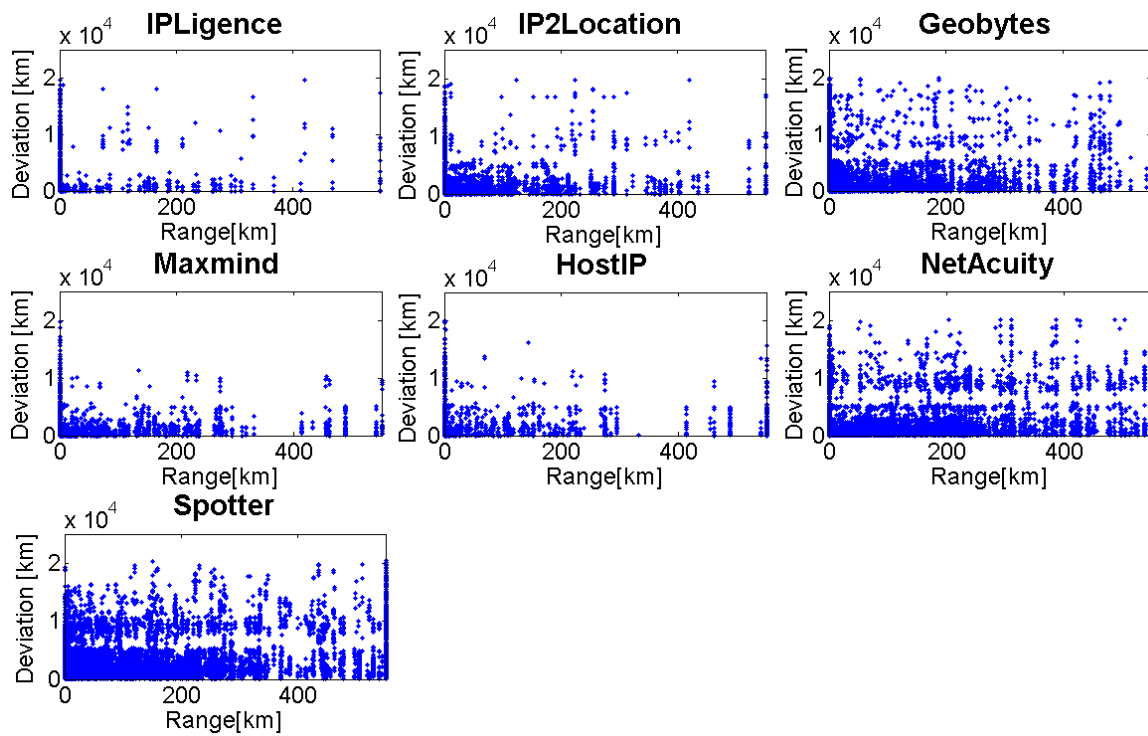
Comparing Figure 4.11 with the median heatmap of Figure 4.9 shows that indeed the three strongly correlated databases tend to bring the PoP median close to them. Looking at the distance error from the PoP median (The horizontal line at 0.5 in Figure 4.11) we see that it crosses IPLigence, Maxmind, and IP2Location at a few kilometer distance, Netacuity at 150km, and HostIP.Info and GeoBytes at roughly 400km, numbers that match the median distance in columns 2 and 3 in Figure 4.9. Spotter values are above 500km, and indeed in Figure 4.11 at 500km its CDF is below 0.5.

Some of the databases, like HostIP.Info, Netacuity, Geobytes and Spotter, deviate less in Europe than in the USA and the rest of the world, as depicted in Figure 4.12. Other databases, like IP2Location, have greater deviation in Europe than the rest of the world. For clarity, only three of the databases are shown in Figure 4.12. A drawback of all databases is that there is a long tail of IP addresses locations which are placed 5000km or more from the PoP median location calculated from the majority of all votes. Figure 4.13 shows that in some databases this tail can hold 15% of the IP addresses. Although the majority vote may be incorrect, this demonstrates that at least one of the databases is significantly far off from the real IP address location.

Figure 4.14 depicts a scatter plot of the range of convergence (X axis) versus the deviation of the IP location from the PoP median location (Y axis) for each database. The figure demonstrates that in many cases the range of convergence is small, yet the deviation from the PoP median location may be thousands of kilometers. Furthermore, a large range of convergence does not imply that the PoP center is necessarily wrong; In all databases we see cases where the range is large, yet the selected IP address location is the same as the majority location from all databases. IPLigence and IP2Location demonstrate an interesting phenomenon: though their range of convergence is very small, the variation from the PoP median location is very large. This can indicate, as is demonstrated next, that large groups of IP addresses are assigned a single false location.

For MaxMind and HostIP there are many PoPs at the far end of the graph, with a large range of





**Figure 4.14: Database Location Deviation From PoP Median vs. Range of Convergence**

convergence. This is caused by lack of information on specific IP addresses which does not allow them to reach a majority vote. Netacuity and Spotter seem to have no strong correlation between the range of convergence and the deviation from the PoPs median location. For Netacuity these may mean that IP addresses are assigned distinct locations within the same area, as with different users in the same city. Spotter suffers from large range of convergence for some PoPs due to NULL replies, however there is an obvious trend that places most PoPs IP addresses within 300km range from each other, with a small number scattered at larger range of convergence, as can be expected in a triangulation based method.

#### 4.3.4.4 Database Anomalies

Though the results above may indicate that some databases have superb location information, this is not the case. In many cases the returned data is deceiving, and actually may represent lack of information in the database. For example, we identified 266 IP addresses in the PoPs that belong to Qwest Communications. Out of those, 253 IP addresses are located by IPLigence in Denver, Colorado. Looking at the raw IPLigence database, there are 20291 entries that belong to Qwest communications, each entry representing a range of IP addresses. Out of those, 20252 entries are located in Denver, which is the location of Qwest’s headquarters. The phenomenon was first detected by our algorithm

in July/2009: 70 Qwest PoPs were detected. Maxmind assigned them to 55 different locations, HostIP.Info to 46 locations, IP2Locations to 35 locations and IPLigence located them all in Denver. In response to a query back then, IPLigence have replied that "In some occasions you could find records belonging to RIPE or any other registrar, these are most likely not used IP addresses but registered under their name, anything else should be empty or null".

Quite a similar case exists with IP2Location. For Cogent, 2365 out of 2879 IP addresses were located in Washington DC, which is Cogent's headquarters location. Out of 57 PoPs belonging to Cogent, only one was not placed by IP2Location in these exact same coordinates. For IPLigence, all the PoPs were located in the same place, too. However, Maxmind placed the PoPs in 13 locations, Geobytes in 23 locations and Netacuity in 31 locations (only a handful in Washington's area). In the Akamai audit by Gomez [40] a similar case is described: A node in Vancouver, Canada was reported to be in Toronto, and a node in Bangalore, India was reported to be in Mumbai, India. In both cases those were ISP headquarters known locations.

Sometimes differences between databases may be very acute, with a reported node location being far off by thousands of kilometers and even countries far apart. In Figure 4.15 one such example is shown. We take a 4-nodes PoP in ASN 703 (Verizon/ UUNET/ MCI Communications) and display on a map the location of the PoP based on each of the geolocation databases. IPLigence, IP2Location, Geobytes, Netacuity and DNS all internally have the PoP four IP addresses at the same location, however each of the databases locate it differently: IPLigence and IP2Location in Australia, Netacuity and DNS in Singapore and Geobytes in Afghanistan. MaxMind and Spotter lack information on these nodes and HostIP.Info places the PoP with 66% certainty in China. Extending our PoP view to include singletons, thus including 10 nodes, the picture does not change. MaxMind and Spotter have location on one of the IPs and they place it in Singapore. IPLigence and IP2location place 9 out of 10 IPs in Australia, and one in Singapore. Geobytes places this last IP address in Singapore too, yet 6 out of 10 IP locations still point to Kabul, Afghanistan. The rest three nodes are located in Australia. Geobytes does give low certainty rate to the location, being 50 or less to both country and region. Netacuity places 8 out of 10 IPs in Singapore and 2 in Australia. HostIP.Info has location information on 6 IPs, 3 of them are placed in China and 3 in Australia, but in Melbourne, far from IPLigence and IP2location designated location. Notably, all the edges in this PoP have less than 3.5mS delay and are measured five to 173 times each.



**Figure 4.15: Mismatch Between Databases - UUNET**

The mismatch between databases is not uncommon. Some examples exist inside the United States, too: in Figure 4.16 we show one PoP in ASN 3549, Global Crossing, as it is placed by the different geolocation databases all across the country. This PoP has over 160 IP addresses, counting singletons, and as such a majority in each database has more substance. IPligence places the PoP with more than 90% majority in Springfield, Missouri. MaxMind and IP2Location point to Saint Louis, Missouri with 92% and 82% accordingly. NetAcuity indicates that the PoP is in San-Jose, California with 100% certainty, while DNS and Spotter place the PoP in this vicinity, in a radius of a few tens of kilometers. GeoBytes has somewhat above 59% of the locations pointing to New York, with other common answers being spread across California (25%). Geobytes country certainty here was 100% with 42% region certainty for the IP addresses it located in New York. HostIP.Info placed the PoP in Chicago with 65% majority (28% of the locations had pointed to Santa Clara, California).

The examples given above are not single incidents. Similar cases have been found in other AS as well, such as REACH (AS 4637), where IPligence, IP2location and Maxmind located a PoP in China, Geobytes located it in Australia, while Netacuity and Spotter put it in the silicon valley, USA. Other cases range from AS16735 (CTBC/Algar Telecom) where PoP locations in Brazil were set thousands of kilometers apart, to Savvis (AS3561) which is another case of locations spread across the USA.



**Figure 4.16: Mismatch Between Databases - Global Crossing**

#### 4.3.4.5 Database Changes

One of the motivations to update geolocation databases is the claim that IP geolocation changes significantly over time. Maxmind [76] claim that it loses accuracy at a rate of approximately 1.5% per month. IP2Location [51] states that on average, 5%-10% of the records are updated in the databases every month due to IP address range relocation and new range available. Based on the PoPs dataset, we compare this information versus the databases at our disposal. For IP2location, an average of approximately one percent of the addresses change every month, with some minimal changes in some consecutive months, such as 0.6% between November and December 2009. In HostIP.Info, 18% of the IP addresses changed their location within nine months, meaning an average of 2% a month. IP2Location changed only 1% of the locations over 4 months, meaning 0.25% per month, however the reference set here included only 10K address range entries. For Netacuity, running only on our dataset of 104K IP addresses, we observe that 2.4% of the IP addresses have changed location in less than a month.

#### 4.3.5 Discussion

Before we discuss our results, it is important to note that the assessment is based on a PoP extraction algorithm, and thus relies on its accuracy. The validation that we described here and in previous sections make us believe that the results are valid. Furthermore, the fact that the results of each standalone database are very good in most cases, and problems appear mainly when comparing databases, strengthens the algorithm correctness. Measurement errors can lead to the unification of interfaces

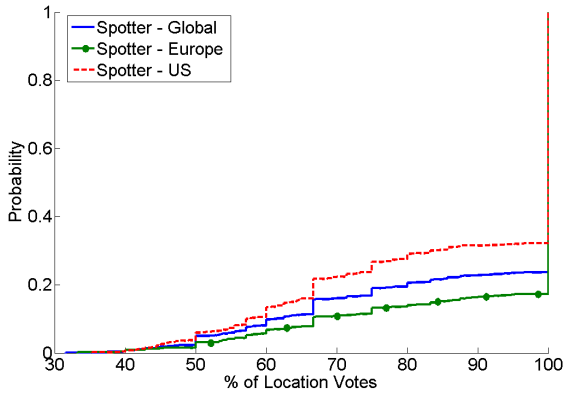
of different locations in the same PoP, but since we use a conservative approach, the more common error is for a PoP to be divided by our algorithm to multiple groups. The later will have some effect on the geolocation database evaluation: it will not affect the IP level analysis (like in Figures 4.9 or 4.15), but where PoP level analysis is considered, the numbers may be slightly altered, yet the overall results will stay the same.

Overall we see that on a region level (500km) the databases are mostly self-consistent, meaning they place all the PoP IP addresses within the same region. This may be sufficient for many location aware applications. At the city level, most databases are still consistent within 82% or more. Note that some of the databases (IPligence, GeoBytes, HostIP.Info) have city-level granularity, namely all the IP addresses within a certain city are placed in a single location. Other databases (like MaxMind) provide sub-city granularity and as a result they may incorrectly seem to perform worse under the 40km or so range of convergence. Some databases (IP2Location and NetAcuity) provide latitude and longitude at city level granularity, but also add zip or postal codes in several countries. These increase the geolocation granularity but could not be leveraged in this work.

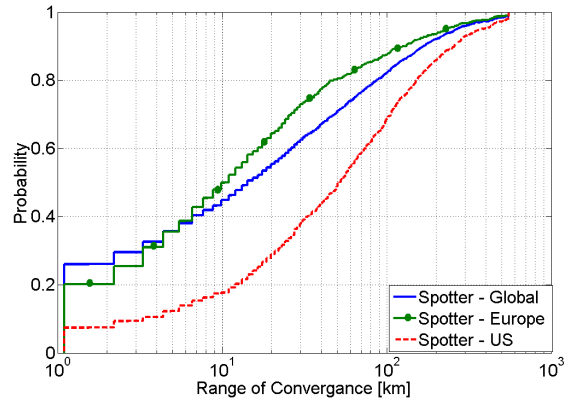
There is a big difference between the region level coherency of different standalone databases and their aggregate. While for all databases 70% to 90% of the PoPs have 100% of the IPs within 500km range, except for HostIP.Info with only 60%, the aggregate has 100% of the nodes only in 4% of the cases. If one is willing to accept an aggregate majority vote among the databases, then at 500km range close to 95% of the PoPs will be successfully located. This percentage drops to less than 70% for city level.

Some faulty locations are easy to detect by users. Most evidently, the case of Qwest and Cogent, where some of the geolocation services provided a single location for the vast majority of the AS's IPs. This is worrisome as geolocation services are probably aware of this fault, and still provide this data. Other services, such as MaxMind, prefer to return NULL reply or only the country. On some occasions, the geolocation service acknowledges the problem and fixes it. For every selected geolocation database it is recommended to check with the vendor the default location returned for unresolved IP addresses before starting to use it.

We find it troubling that there are too many cases where database disagreement spans across huge geographic distances. The problem appeared not only in small PoPs, that may be affected by sporadic errors in the database, but also in PoPs with hundreds of IP addresses, where the databases had high



**Figure 4.17: Breakdown of location votes percentage CDF for Spotter by region.**



**Figure 4.18: Breakdown of convergence range CDF for Spotter by region.**

certainty on their indicated location (as shown in Figure 4.16).

#### 4.3.5.1 Active Measurement Accuracy

Active measurements are used by many geolocation services [63, 126, 70] and by other projects for different localization tasks, most notably for assigning IP addresses to PoPs [104]. Spotter geolocation is based solely on active measurements, thus we selected to study its performance in greater depth due to the importance of understanding the limitations of this approach.

Figures 4.17 and 4.18 show Spotter’s overall performance compared with its performance for PoPs located only in Europe or in the USA. It is clear from both figures that in Europe Spotter performs much better than in the USA and slightly better than the world average. For example, for 40km radius (which is frequently used as a city diameter), Spotter reaches about 78% convergence in Europe compared to 67% convergence worldwide, and only 44% for the USA. The difference can be explained<sup>1</sup> by the spread of vantage points used by Spotter, which are almost entirely based on PlanetLab nodes. While in Europe PlanetLab nodes are well spread geographically, in the USA, most PlanetLab nodes are located along the coasts making localization of IP addresses in the middle of the USA less accurate. Interestingly, other databases which are based on other geolocation techniques also achieve better results for European addresses than for USA addresses.

Spotter convergence (Fig. 4.5) starts as the lowest which is an outcome of the measurement error that tend to spread the results for different IPs around the ‘true’ location. However, at a radius of 100km it closes the gap with most databases and reaches over 80% convergence (and close to 90% for Europe).

<sup>1</sup>We consulted Peter Haga and Peter Matray from the Spotter project on this aspect.

However, 20% ‘error’ may make distance measurements unfit as the sole method for assigning IP addresses to PoPs.

#### *4.3.6 Conclusion*

Section 4.3 presented a comprehensive study of geolocation databases, comparing a large number of databases of different types. The results show that while most of the databases provide results that seem coherent, the accuracy of the returned location can not always be trusted. There is a strong correlation between some databases, which indicates that the vast majority of location information replies are correct. However, there is a long and fat tail of errors in the databases; These errors are in the range of thousands of kilometers and countries apart. The use of geolocation databases should therefore be careful.

Our results also show that measurement based geolocation can achieve fair results that may compete, at least in Europe, with geolocation information gathered by other means and that the achieved accuracy of geolocation using such tools can be reasonably high. However, this accuracy may not be high enough to be used as the sole tool to map IP addresses to PoPs. There is room for better understanding the roots of measurement based geolocation services inaccuracy in order to improve them. The following chapter focuses on means to decide on ground truth when there is a disagreement between the databases.

### **4.4 Crawling Geolocation Algorithm**

The PoP geolocation algorithm was found to work well, however it is not error free as it depends on the quality of the geolocation databases it uses. When the differences between databases are extreme, as shown in the previous chapter, it fails to locate the related PoPs. We thus propose a method to improve this initial geolocation using a crawling algorithm. This method can be further expended to improve IP-level geolocation.

A property of our initial geolocation algorithm is that it gives a confidence to the PoP’s location. Each PoP is assigned a range of error, being the minimal radius covering 50% or more of the PoP’s IP addresses locations (but no more than 100 kilometers, a threshold set in the algorithm), and the confidence is derived from the percentage of IP-level locations included within this radius. Using

PoPs with a known location (such as universities) and PoPs marked with a high level of confidence, we find the location of PoPs with a lower level of confidence.

The algorithm starts by identifying and marking the PoPs for which the location is certain. The algorithm then discovers PoPs that are located in the same place as the marked PoPs, based on a PoP-level link delay. Next it examines all the PoP's IP-level locations in the geolocation services, and finds one that gives the best delay-distance matching to marked neighboring PoPs. If no location passes a goodness threshold, multilateration from the marked neighboring PoPs is used. The algorithm then iterates and attempts to improve the location of PoPs that were not handled yet using the location of newly marked PoPs. Figure 4.19 shows the stages of the algorithm, detailed as follows:

**Initial PoP Map** The algorithm begins running on a given PoP level map, shown in Figure 4.19(a). Each vertex represents a PoP and each edge represents a link between two PoPs, annotated with delay information. Each PoP is initially assigned a location based on the naive PoP geolocation algorithm described in Section 4.1.

**Primary Anchor PoPs Marking** Mark all PoPs with a definite known location as *anchors* (dark nodes), the rest of the PoPs (light nodes) are placed based on the previous naive geolocation algorithm [34]. Anchor PoPs belong to universities, research facilities, and other known locations.

**Additional Anchor PoPs Marking** Mark all PoPs with a high level of confidence as *anchors*. An anchor PoP can be used to geolocate other PoPs with a lower level of confidence. For high level of confidence the following three conditions are required:

- $P_{tot} \geq P_{tot\_th}$ , where  $P_{tot}$  is the percentage of IP level locations within the PoP's error range and  $P_{tot\_th}$  is a threshold for this parameter.
- $P_{IP} \geq P_{IP\_th}$ , where  $P_{IP}$  is the percentage of IP level locations located within the PoP's location error range when "no location" replies are excluded and  $P_{IP\_th}$  is a threshold for this parameter.
- $R \leq R_{th}$ , where  $R$  is the location error range of the PoP and  $R_{th}$  is the range radius threshold for this parameter.

The anchor PoPs  $\{B, F, I, N\}$ , marked during the primary and additional marking stages, are shown as dark nodes in Figure 4.19(b).

**Co-Locate PoPs** For each unmarked PoP node with a link delay below a certain threshold ( $D_{co\_th}$ , typically less than  $1ms$ ) to a marked PoP, one can assume that both PoPs are located in the same place.



We thus define the PoPs as *co-located*, assign to the unmarked PoP the same location as the marked PoP, and add it to the group of marked PoPs. In Figure 4.19(c) the co-located PoPs are A,C,D,O since the link delays on edges (A,B), (B,C), (D,F), and (O,N) are all less than  $D_{co\_th} = 1ms$  (1mS was selected for demonstration). After updating the geolocation of PoPs A,C,D,O they are marked.

**Location Update by Delay and Geolocation Data** For each unmarked PoP in the map with at least one neighboring marked PoP,  $POP_{M_i}$   $1 \leq i \leq k$  ( $k$  is the number of such neighbors), go over locations  $L_{IP}$  of all the IP addresses included in the unmarked PoP, as indicated by each geolocation service. For every  $L_{IP}$ , calculate the delay to distance ratio  $RA$  from the location  $L_{IP}$  to every  $POP_{M_i}$  and its corresponding link delay. If there is  $L_{IP}$  such that  $RA_{min} \leq RA \leq RA_{max}$ , set the location of the PoP to  $L_{IP}$  and mark it. If there is more than one such location select the location with the best lexicographic ordered  $RA$  ratio value. In our example (Figure 4.19(d)), PoP M has two neighboring marked PoPs: N and D. The delay from M to N is  $7ms$  and from M to D -  $10ms$ . If we set  $RA_{min} = 95km/ms$ ,  $RA_{max} = 110km/ms$ , we expect PoP M to be in the range of 665km to 770km from PoP N and 950km to 1100km from PoP D, as indicated by the shaded circles in the figure. While the location of PoP M was initially set by the majority vote of all geolocation databases, two more alternative locations were indicated by some of the databases: M1 and M2. Since M2 is located within the expected range from N and D it is selected as the location of PoP M, and the PoP is marked.

**Location Update by Delay Only** For each unmarked PoP node with at least three neighboring marked PoPs,  $POP_{M_i}$   $1 \leq i \leq k$  ( $k$  is the number of such neighbors), update the PoP location such that the ratio  $RA$  between the PoP's geographic distance from every  $POP_{M_i}$  and its corresponding link delay will be closest to the optimal ratio value  $RA_{opt}$ . In other words, multilaterate the PoP's location based on the delay from the marked PoPs and their geographic location and mark it. A constraints based approach is currently used for the multilateration, but other methods of multilateration may be used. In Figure 4.19(e) example, only node J is a candidate for location update by delay. The three shaded circles around PoPs B,I,N indicate the expected location of PoP J relative to each one of them. The location of PoP J is thus updated to the crossing area of all three ranges and the PoP is marked.

**Crawling** Iterate the Co-Locate and Location Update stages, using previously marked PoPs to update the location of non-marked ones. As a result, the PoPs locations are propagated through the PoPs network, such that PoPs with a high level of accuracy update the location of PoPs with a medium level of accuracy, and those in turn update others. The process ends after no PoP can be added to the

group of marked PoPs. Figure 4.19(f) shows the map of geolocated PoPs after the first iteration: there are ten marked PoPs and four unmarked PoPs. PoPs E and K will be relocated in the second iteration, as their neighboring PoP is now marked and their link delay is less than 1ms. PoP L will be relocated in the Location Update by Delay Only step. PoP H either will be relocated in the Location Update by Delay and Geolocation Data step or will not be marked. The third iteration will have no updates and thus the crawling algorithm will terminate, as shown in Figure 4.19(g).

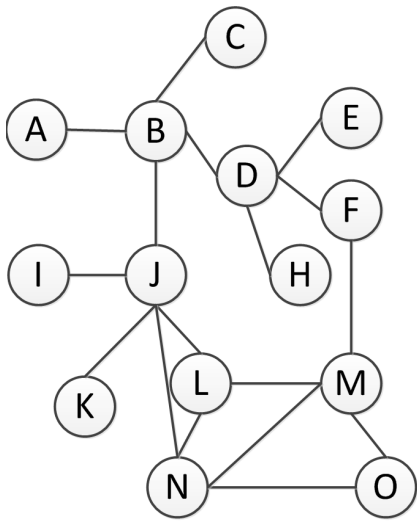
The algorithm description above provides the skeleton of the Crawling algorithm. The algorithm includes several refinements, intended to identify and clear noisy data. First, the selection of marked PoPs in the **Additional Anchor PoPs Marking** stage is refined in a manner that excludes PoPs with an initial distance to delay ratio significantly different from other neighbor marked PoPs. Such PoPs are likely to have their position incorrect in all geolocation databases, as rarely happens [99]. Even if their location is correct but delay measurements to them are inaccurate, they should not be used for positioning of further PoPs and are thus unmarked. Second, in the co-location stage, if a PoP has multiple possible co-locations, due to link delay below  $D_{co\_th}$  to more than one marked PoP, while the marked PoPs are not co-located, the PoP will not be marked. Furthermore, such a PoP will be flagged as erroneous and will be manually checked later, as it may indicate that previously marked PoPs were erroneous themselves. Note that if two PoPs are positioned in the same region but not in the same place (e.g. 200km apart), such an occurrence is possible. Yet, executions of the algorithm so far flagged no such PoP.

The algorithm can be further extended to IP-level geolocation. For a given target IP address, take the following steps:

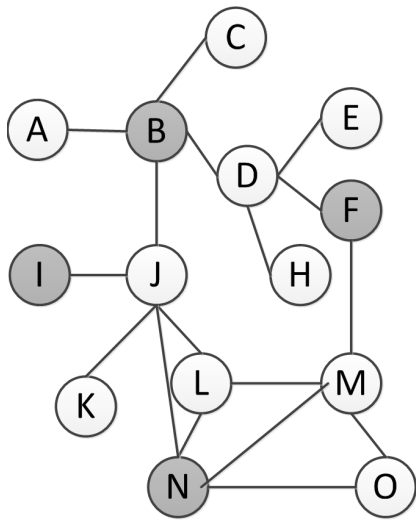
**PoP Located** If the target is part of a PoP, assign it the location of its hosting PoP.

**CIDR/24 based** If there is an IP address in a PoP with the same CIDR/24 as the target, assign to the target the location of the PoP. If multiple such IP addresses exist, use the location of the longest prefix match IP. We note that some loss of accuracy exists in such a case, but this provides at least the same level of accuracy as most geolocation databases.

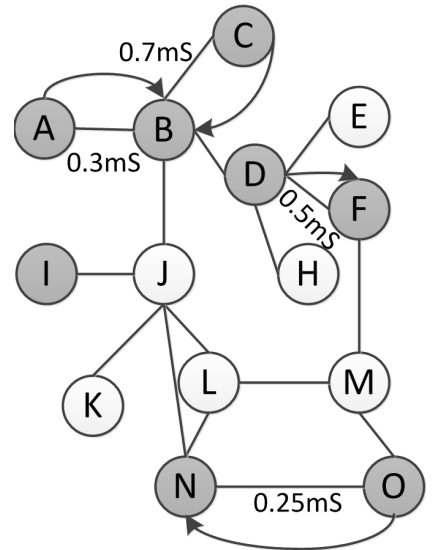
**One-Hop Location** If the target is one hop away from an IP in a PoP or an IP conforming with the CIDR/24 rule, and the edge delay is less than  $D_{co\_th}$ , assign the target the same location as its one-hop neighbor.



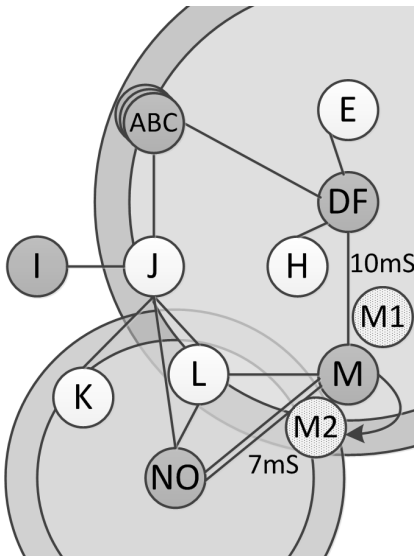
(a) Initial PoP Map



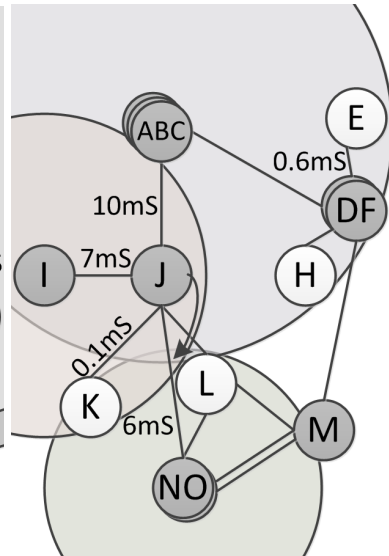
(b) Anchor PoPs Marking



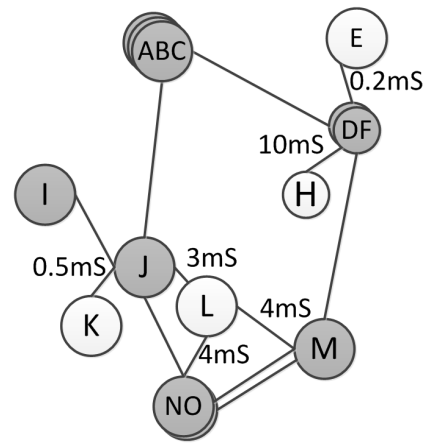
(c) Co-Locate PoPs



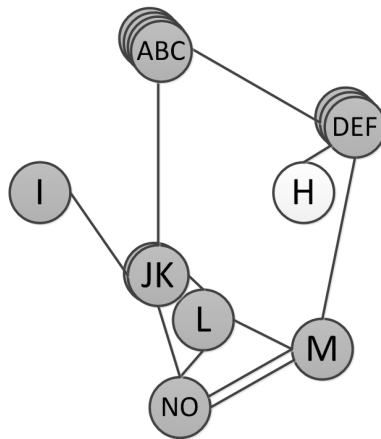
(d) Update by Delay and Geo-Data



(e) Update by Delay Only



(f) Crawling



(g) Final PoP Map

**Figure 4.19: Crawling Algorithm**

**Two-Hop Location** same as above but with two hops.

**PoP-IP Multilateration** Find the three IP addresses which are part of different PoPs, with minimal delay from the target, and use multilateration for the target location.

#### **4.5 Crawling Geolocation - Datasets**

Two datasets are used for the validation of the crawling algorithm: one from 2012, and one from 2010, which was selected as it was carefully studied in sections 4.2 and 4.3 and its characteristics are known. Both datasets use measurements from DIMES [25] and iPlane [73]. We note that the traceroute measurements are performed differently by DIMES and iPlane, as every DIMES measurement is combined of a train of four traceroute measurements, and only the best time of every hop is used for an edge delay calculation. This affects the results beyond a ratio of 1 : 4 in the number of measurements. For example, we filter out faulty traceroute hops, such as IP and AS level loops on edge level. Over 170 million measurements are filtered out of the iPlane measurements, while only 61K such measurements are filtered from the DIMES data (DIMES filters some of the measurements before adding them to the database). Due to the differences, edges discovered by DIMES are annotated with delay information measured only by DIMES, and iPlane data is used to add edges that were not discovered by DIMES. iPlane typically increases the number of discovered edges by  $\tilde{20}\%$ , but it measures only a small subset of the edges that DIMES discovered.

**2010 Dataset** The dataset is comprised of 478 million traceroutes conducted in weeks 42 and 43 of 2010, measured by 1308 DIMES agents and 242 iPlane vantage nodes. Five geolocation databases are used for the naive geolocation of the PoPs: MaxMind GeoIP City[76] , IPLigence Max [62] , IP2Location DB5[51] , GeoBytes [37] and HostIP.info [53]. Two more geolocation services, NetAcuity [24] and Spotter [70], were tested for the geolocation of PoPs measured by DIMES alone. The generated PoP level map contains 4750 PoPs and 87.3K IP addresses in 1697 different ASes. 4098 PoPs are discovered using the DIMES data alone. We further extend the map by adding universities, research institutes and exchanges points, which were measured by DIMES and iPlane and whose location is known.

**2012 Dataset** The measurements in this dataset are taken from weeks 19 and 20 of 2012, starting the 6th of May. 203 million traceroutes were collected from 988 DIMES agents and 153 iPlane

Crawling Algorithm Stage	Relocated PoPs	
	2010	2012
Anchors	17.9%	29.3%
Co-Location	28.3%	20.1%
Delay and Geolocation Data	21.3%	13%
Delay Only	7.6%	6.1%
Not Relocated	24.8%	31.5%

**Table 4.4: PoP Relocation By Algorithm's Stage**

vantage points. Five geolocation databases are used for the naive geolocation of the PoPs: MaxMind GeoIPLite City [76], IPelligence Max [62], HostIP.info [53], DB-IP [21] and NeuStar's IP Intelligence (formerly Quova) [5]. The generated PoP level map contains 5215 PoPs and 98650 IP addresses in 2636 different ASes. This map contains also universities, research institutes and exchanges points, as in the 2010 dataset.

#### **4.6 Validation of the Crawling Geolocation Algorithm**

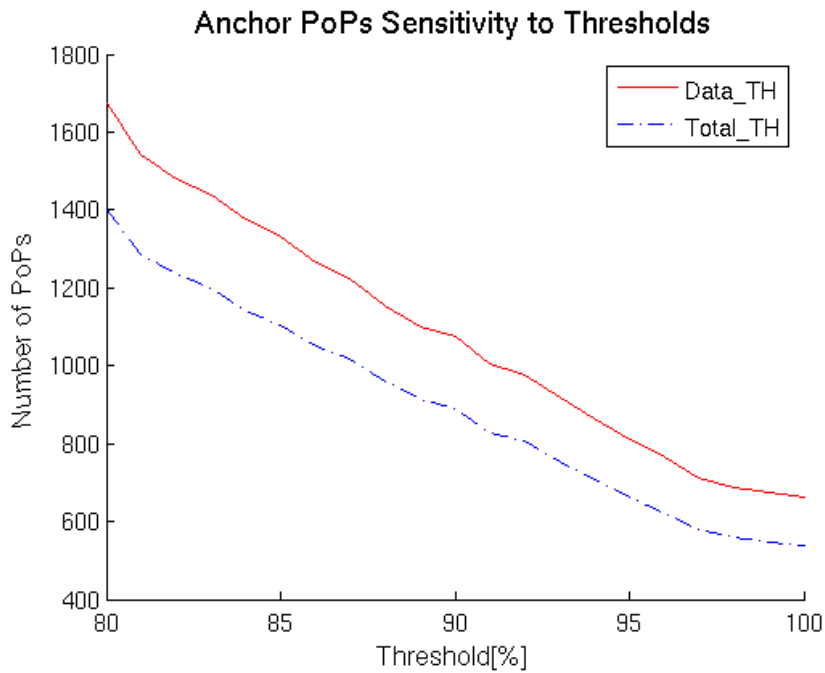
The PoP Crawling algorithm is initially executed using very conservative thresholds, described in subsection 4.6.1. The sensitivity of the algorithm to these parameters is studied later in this section.

Running the algorithm converges fast for both datasets: only five crawling iterations, the last iteration without any update. 47.3% of the 2010 PoPs and 34.7% of the 2012 PoPs are marked on the first iteration. The crawling algorithm results are broken in Table 4.4 by the algorithm stages (or relocation method). Less than 8% of the PoPs were relocated based on link delay only.

Between a quarter to a third of the PoPs are not affected by the crawling algorithm, and maintain their naive original position, as shown in Table 4.5. There are several reasons for not marking a PoP: First, the PoP may not be connected to any other PoP, which is the case for over a quarter of the unmarked PoPs in 2012. Note that such a PoP is connected to other nodes with IP-level edges, otherwise it would have not been detected and it is likely connected to other PoPs, but such PoPs or links were not measured by iPlane or DIMES. For many of the PoPs, there are no other marked PoPs in their vicinity to allow crawling, thus creating "islands" of unmarked PoPs. Last, some PoPs fail the relocation by delay only, mostly because their marked neighbor PoPs do not allow multilateration, e.g., if their (three) neighbor PoPs are co-located.

Cause for Failure	% of PoPs Not Relocated	
	2010	2012
No Neighbors	8.4%	26.3%
No Marked Neighbors	41.8%	40.6%
Not Enough Marked Neighbors For Multilateration	2.4%	1.6%
Multilateration Not Possible	43.6%	28.8%
Multilateration Failed Within Required Thresholds	3.8%	2.7%

**Table 4.5: PoP Relocation Failures Breakdown**



**Figure 4.20: Number of High Level Confidence PoPs vs. Thresholds Values**

#### 4.6.1 Algorithm's Parameters

As several thresholds are involved in the algorithm, it is important to check their effect on its performance. Our goal is to maintain the accuracy of relocation while minimizing the number of relocation failures. The first thresholds to be tested are those controlling the selection of high level of confidence PoPs:  $P_{tot\_th}$  and  $P_{IP\_th}$ . Figure 4.20 demonstrates the algorithm's sensitivity to these parameters, with the solid line showing the number of PoPs marked during the "Additional Anchor PoPs Marking" step as a function of  $P_{IP\_th}$  and the dashed line showing the effect of  $P_{tot\_th}$ . For both thresholds the number of anchored PoPs linearly grows as the threshold decreases. However, even when selecting the most conservative values, meaning setting  $P_{DATA\_TH} = 100\%$  and  $P_{tot\_th} = 100\%$ , which provide both a highest level of accuracy, enough PoPs are marked to use as anchors in the crawling process. The

Year	Delay [mS]					
	0	1	2	3	4	5
2010	28.3%	28.7%	26.5%	26.6%	23.8%	23.8%
2012	20.1%	22.9%	23.7%	23.8%	23.9%	24.3%

**Table 4.6: Number of Co-Located PoPs vs. Threshold Value**

threshold  $D_{co\_th}$  is evaluated in the same manner, testing the stability of the "Co-Locate PoPs" stage. Table 4.6 shows the percentage of Co-Located PoPs as a function of  $D_{co\_th}$ . We set  $D_{co\_th} = 0$ , the most conservative value possible. Testing the algorithm's sensitivity to  $D_{co\_th}$ , varying its value from zero to 5mS, we find up to 4% variance in the number of co-located PoPs both in 2010 and 2012. In 2010 increasing the threshold sometimes reduced the number of co-located PoPs. This is caused by the crawling nature of the algorithm, as a PoP that was marked as co-located in a late iteration is now marked in an earlier iteration, for example in the Location by Delay and Geolocation stage, as one of its neighbor PoPs was marked for co-location using the higher  $D_{co\_th}$ .

It is possible to find errors in the location of PoPs with a high level of confidence. Such can occur if all databases share the same error. The algorithm searches for such errors during the **Co-Locate PoPs** stage. Having a PoP co-located with multiple marked PoPs with a different location indicates that the marked PoP location is incorrect and should be unmarked. Several such events were flagged and the affected PoPs were unmarked and relocated, being treated as PoPs with a low level of confidence. These cases happened within ISPs such as Cogent (AS174), which were shown [99] to have a large number of false locations in the geolocation databases. The algorithm is thus tested using only anchors with a definite known location. We find that the PoPs marked as anchors during the **Additional Anchor PoPs Marking** stage fall within one of two categories: the algorithm either keeps them in their original place (i.e., by co-location), or fails to relocate them, as they have no marked neighbors. In 2010, these PoPs were 37.5% of all anchor PoPs, while in 2012 they were 70.3%. Consequently, the overall number of PoPs that fail due to lack of neighbors or marked neighbors rose in 2010 by 240% when PoPs with a high level of confidence were not used, while in 2012, the usage of the **Additional Anchor PoPs Marking** contributed only 1% of additional relocation rate.

As many previous works have shown [70, 63] delay measurements for multilateration purposes tend to be inaccurate due to additive latency. The use of PoP level links allows to aggregate multiple edges into a single PoP link and reduce the measurement inaccuracy, as shown in Figure 5.5. The spread of edge level delays per PoP shows the importance of aggregating multiple edges into a single link.

Thus, if a PoP level link is comprised of a single edge, and this edge was measured only a few times, its latency is likely not to be sufficient from multilateration purposes. We note that 37%-39% of the PoP links contain a single edge, and over 94% of those are measured less than five times. Such links are not used for geolocation by multilateration, as they might introduce large errors.

We rigorously checked the thresholds used for multilateration, and report the main findings.  $RA_{opt}$  is selected based on delay to distance ratio on links between anchor PoPs. The ratio of the 2012 dataset is more stable and less sensitive to a change of the thresholds than in 2010 with the average ratio always being in the region of 1.3ms/100km-1.5ms/100km.  $RA = 1ms/100km$  is a commonly used value but was shown to be an under-estimate [49], we thus set  $RA_{min} = 0.95ms/100km$  and  $RA_{max} = 2ms/100km$ . As fiber infrastructure depends on terrain conditions and obstacles bypass it is expected that the routed fiber length will be closer to  $\sqrt{2}$  air distance, which complies with the ratio measured on 2012. We thus select  $RA_{opt} = 1.44ms/100km$ .

The values of  $RA_{min}$  and  $RA_{max}$  do not have a considerable effect on the relocation of PoPs: for  $RA_{min} = 0$  and  $RA_{max} = \infty$  the PoPs relocated by delay only have an average delay to distance ratio of  $1.002 \times RA_{opt}$  and only 8.5% of these PoPs have a ratio outside the default  $(RA_{min}, RA_{max})$  range. The maximal measured ratio is 204 and the minimal is 2.1.

Since the algorithm is oblivious to the multilateration algorithm used, and less than 8% of the PoPs are relocated by multilateration, we refrain from further analysis of this aspect, which was studied by [49, 63, 70].

#### 4.6.2 Validation of Location Assignment

When examining the Crawling algorithm location, we need to verify two points: the algorithm must not damage the location of correctly assigned PoPs, and it should correct the location of PoPs that were wrongly assigned. Since the initial location of PoPs is already verified to be very good [34], by keeping the damage close to zero, any improvement in the location of wrongly placed PoPs will result in a very accurate map. The lack of ground truth make geolocation validation difficult, but as we show below, we manage to show that indeed the crawling algorithm performs well.

First, we compare the location assigned to PoPs that we already verified in previous works, and find that relocation assignments are within 200km range. Next, we focus our efforts on ASes where



geolocation issues exist, e.g., where the geolocation databases assign all the PoPs to a single location. Validation of ISP's PoPs is done based on the service providers maps. To this end, we use providers maps that were collected by the Internet Topology Zoo project [66] at the same period as our dataset or published by the ISP: Abilene, UUNET (AS701 through AS703), China Telecom, both within China and International, and more. The validation shows that most PoPs are located where expected, with the only exception applying to PoPs placed using multilateration, which are sometimes located with an error of a few hundred kilometers.

Next, we check the correctness of the algorithm using primary anchor PoPs. By unmarking a primary anchor PoP and applying the crawling algorithm to it, one can verify that the location assigned to this PoP is correct and that the algorithm does not relocate PoPs away from their correct location. To this end, we tested 180 primary anchor PoPs. 124 of the PoPs were assigned a location using co-location, 20 PoPs were relocated using *Location Update by Delay and Geolocation Data*, and the rest were assigned a location in the *Location update by delay* stage. Table 4.7 shows the breakdown of these PoPs relocation by crawling. Most of the PoPs (82%) retain their original position or are located within 100km from their original location (5%). All the PoPs that are relocated using co-location maintain their original position. A total of 94% of the PoPs are located within 500km range of their original location.

We examine the PoPs that were relocated with an error larger than 500km, and find that the cause is noise in the dataset. The PoPs that were located by *Location Update by Delay and Geolocation Data* are characterized by lack of location in most of the Geolocation databases. For example, Harvard's GigaPoP (AS10578) does not have location information in three of the geolocation databases at all. In one database (NeuStar) a single location appears for all the IP addresses, matching the original PoP location. In the last database, location information appears only for some of the IP addresses. The location information differs from NeuStar's and also points to two different locations, one of them later selected as the new relocation position. This PoP is also characterized by noisy link delay to neighboring anchor PoPs, manifesting as long link delays (hundreds of milliseconds) to PoPs located within a few hundreds of kilometers from Harvard. We find that also within the AS there are long link delays that reach almost 30mS, even between IP addresses in the same CIDR/26. The combination of disinformation in the geolocation database and noisy delay measurements leads to the algorithm's error. Similar noisy delay measurements also affect the PoPs located in the *Location update by delay*

<b>Crawling Stage</b>	<b>Number of PoPs</b>	<b>Same Place</b>	<b>Within 100km</b>	<b>Within 500km</b>	<b>Beyond 500km</b>
Co-Location	124	100%	0%	0%	0%
Delay and Geo-Data	19	16%	47%	16%	21%
Delay Only	37	54%	0%	25%	21%
Total	180	82%	5%	7%	6%

**Table 4.7: Known PoPs Relocation Accuracy**

stage.

One way to clean noisy link delay is to increase the threshold of required edges between two PoPs above one for the *Location Update by Delay and Geolocation Data* and *Location update by delay* stages. As the median number of edges between a pair of PoPs is two, with an average of 19.7, this is a conservative rule. The advantage is an increased accuracy, while the disadvantage is the decrease relocation success rate. For example, increasing the number of edges threshold to two, reduces the number of known PoPs relocated from 180 to 145. Out of the 12 PoPs that were located with an error range larger than 500km, 25% of the PoPs are now correctly relocated (within 100km range) and 25% more can not be crawled. Increasing the number of required edges between a pair PoPs to at least three, corrects the location of two more PoPs, thus eventually only 5 PoPs (2.8%) are located outside 500km error range.

The multilateration method used for the geolocation assessment was constraint based. To evaluate possible improvement by using other multilateration methods, we use Spotter [70]. Due to Spotter's resources limitations, we were able to evaluate only the location of the 12 PoPs that were located outside the 500km range, which include 980 IP addresses. Spotter provides a location to 88% of the IP addresses, covering all PoPs. Using Spotter, the location of 58.3% of the PoPs is set within 100km of their true location, 25% more within 500km range, and only 2 PoPs are set outside the 500km region. One of these PoPs is located even further than was estimated by the constraint based approach. We note that Spotter does have some accuracy leverage in the geolocation of the 12 PoPs under study, as most of them belong to universities with, or very close to, PlanetLab [18] nodes. This increases the measurement's accuracy compared to a target located far from PlanetLab nodes.

An advantage of Geolocation using PoPs rather than other methods is shown when considering Spotter's results on the IP level: 41.7% of the IP addresses are located beyond the 100km range of error, and 3.5% beyond the 500km range. Most of the IP addresses outside the 500km error range are lo-

cated far from adjacent IP addresses, though the PoP to which they belong is located correctly. On one extreme case, of Hong-Kong University, 86 out of 91 IP addresses are located correctly in Hong Kong, three more are within 500km range, one in the Philippine Sea and one far off in Zimbabwe.

An example of an AS where the PoP Crawling algorithm has a significant effect is Telefonica. In Telefonica (AS12956) at 2010, 26 pops are detected (using DIMES data only), and all are originally assigned to Madrid, Spain. After running the PoP Crawling algorithm, the PoPs are assigned to 16 different locations, including Santiago, Chile, Amsterdam, the Netherlands, and more. At 2012, 21 PoPs are detected. The Telefonica PoP map misses some business users' PoPs and some of the Latin America PoPs, but is otherwise accurate. Figure 4.21 shows the PoP level map that was validated, with the red pin indicating the PoPs location before running the algorithm, and the blue icons showing the location of the relocated PoPs.

We also corroborated the data with one large ISP and the error range CDF is depicted in Figure 4.22. 65% of the PoPs were located by the algorithm within 100km from their true location, and 85% within 300km range. Less than 4% of the PoPs are not located within 500km from their true location and in only one case there is a country-level error, where a PoP is located close to the Chilean border. The ISP indicated that before the crawling algorithm was executed, only 23% of the PoPs were placed within 500km range of their true location.

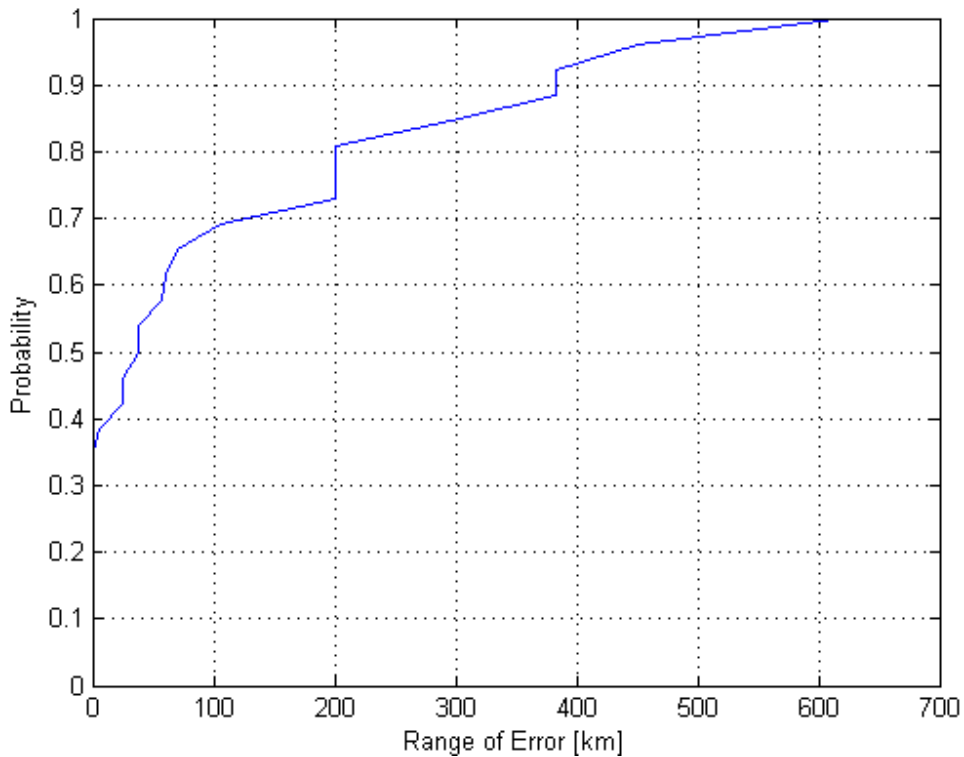
#### 4.6.3 *Comparison to Geolocation Databases*

The effect of the crawling algorithm on a PoP's location is demonstrated in Figure 4.23. The figure presents a heatmap of the median distance between all the geolocation services used with each dataset and the PoP geolocation algorithms, both naive and crawling geolocation, excluding IP addresses that were not marked during the crawling process. As we have shown in our previous work [99], the databases IPLigence, IP2Location and MaxMind have high correlation. Due to the majority vote of the naive algorithm, its median distance from these three databases is very small, less than 45-55km. In the 2012 dataset we observe that Neustar's database is also correlated with MaxMind and IPLigence, and consequently close to the location by the naive algorithm, as well.

The crawling algorithm leads to a median displacement of PoPs by over 400km (compared to the naive algorithm) in 2010 yet only 80km in 2012. This points to a possible improvement in the geolocation databases. The crawling also results now with locations closer to those indicated by IPLigence and



**Figure 4.21: Telefonica PoPs Location Map**



**Figure 4.22: A CDF of a Large ISP Relocation Range of Error**

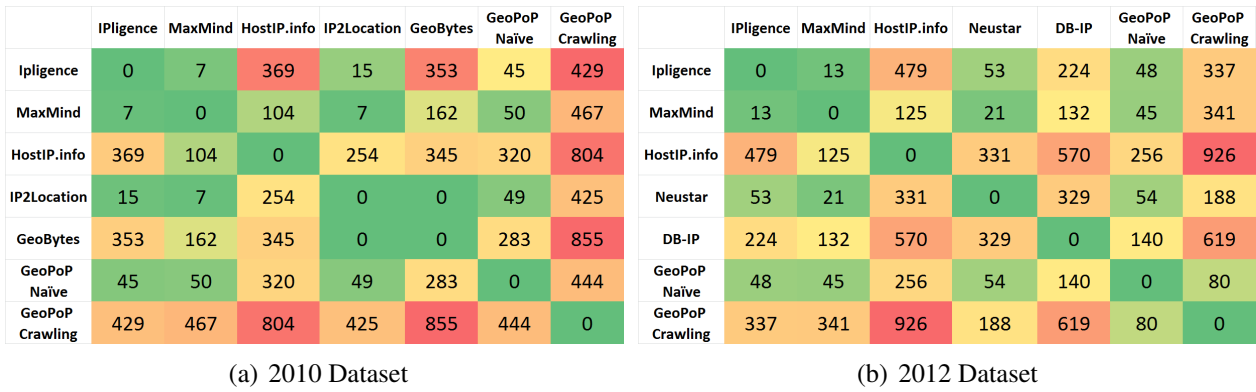


Figure 4.23: Heatmap of Median Distance Between Geolocation Services

IP Geolocation Algorithm Stage	2010	2012
PoP Located IP Addresses	87.3K	98.65K
CIDR /24 Blocks of PoP's IP Addresses	38.56K	45.96K
One-Hop IP Addresses	729K	441K
Two-Hop IP Addresses	1004K	672K
Two-Hop Routing Blocks	90525	87785
IP Addresses Measured	1.69M	1.32M

Table 4.8: IP Geolocation By Algorithm's Stage

Maxmind (compared to 2010). From all the databases, the crawling results are closest to Neustar, which is priced highest from all the geolocation services in use. The relatively small median distance (region range), is a possible indicator of the database's accuracy.

#### 4.6.4 IP Geolocation

The contribution of IP-level geolocation using PoPs manifests mainly in the first four stages of the IP geolocation algorithm, thus we evaluate the coverage of IP addresses by these stages as presented in Table 4.8. Of all the IP addresses measured by DIMES and iPlane, 50% to 60% can be co-located within 2-hops from a PoP with an overall delay of less than 2mS (up to 1mS per edge). As not all the IP addresses are targeted for measurement, it is important to consider also the number of routing blocks covered by this range. 90525 routing blocks (out of 219750 routing blocks, as indicated at the time by Routeviews [117]) were covered in 2010 and 87785 routing blocks (out of 260954) were covered in 2012.

We next use a set of IP addresses with a known location. They are taken from a 2010 ground truth database provided by CAIDA, described in [54], includes private data from one tier-1 and one tier-2

ISPs. In addition it contains public data from five research networks. The geographic location is provided based on host names, with their encoding provided by the ISP and verified. The database covers 25K addresses, but only 2241 are covered by our 2010 dataset (no aliasing used). 2201 of these addresses were wrongly assigned by the original PoP algorithm. 1656 IP addresses were not marked by the crawling algorithm, and out of the remaining 545 relocated IPs 418 were correctly relocated within 100km, and additional 18 within 500km (The ground truth is highly biased to 2 ISPs and is thus not representative [99])

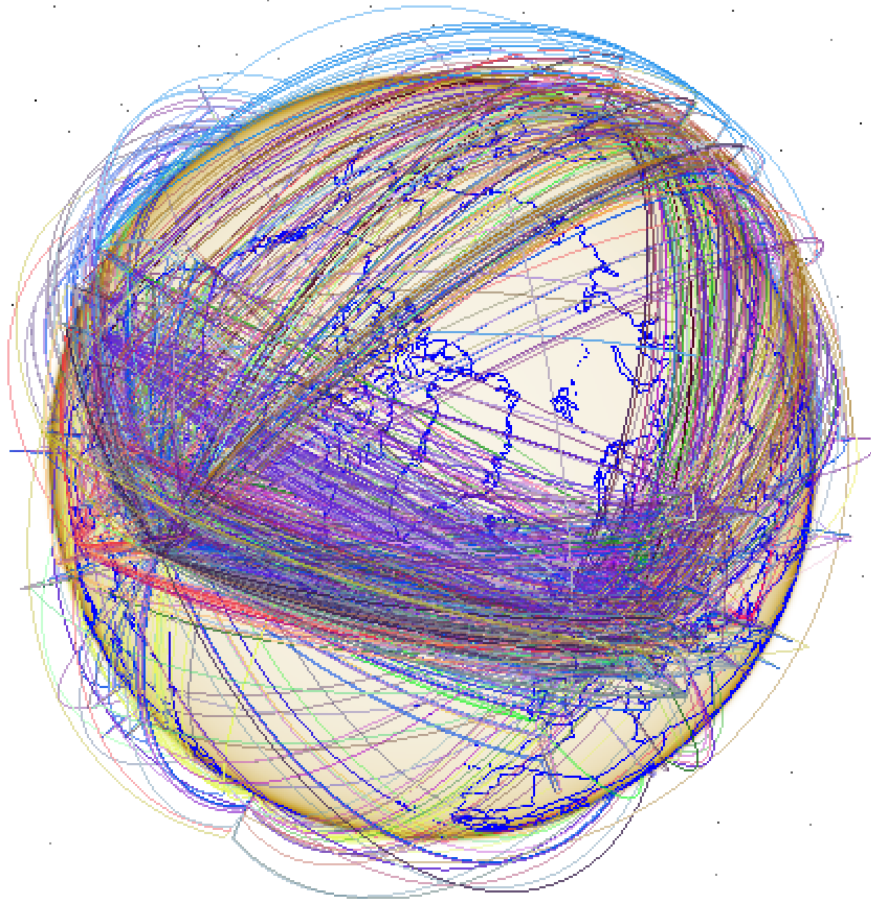
## V Connectivity in PoP Level Maps

### 5.1 Defining PoP Level Connectivity

The connectivity between PoPs is an important part of PoP level maps. DIMES generates PoPs connectivity graph using unidirectional links. We define a link  $L_{SD}$  as the aggregation of all unidirectional edges originating from an IP address included in a PoP  $S$  and arriving at an IP address included in a PoP  $D$ . Each of the IP level links has an estimate of the median delay measured along it, with the median calculated on the minimal delay of up to four consecutive measurements. Every such four measurements comprise a basic DIMES operation. All measured values are roundtrip delays [34]. A Link has the following properties:

- Source and Destination PoP nodes.
- The number of edges aggregated within the link.
- Minimal and Maximal median delays of all IP edges that are part of the PoP level link.
- Mean and standard deviation of all edges median delays.
- Weighted delay of all edges median delays. The edge's weight is the number of times it was measured.
- The geographical distance between source and destination PoP, calculated based on the PoPs geolocation.

A weighted delay of a link is used to mitigate the effect of an edge with a single measurement on the overall link delay estimation, where a link is otherwise measured tens of times through other edges.



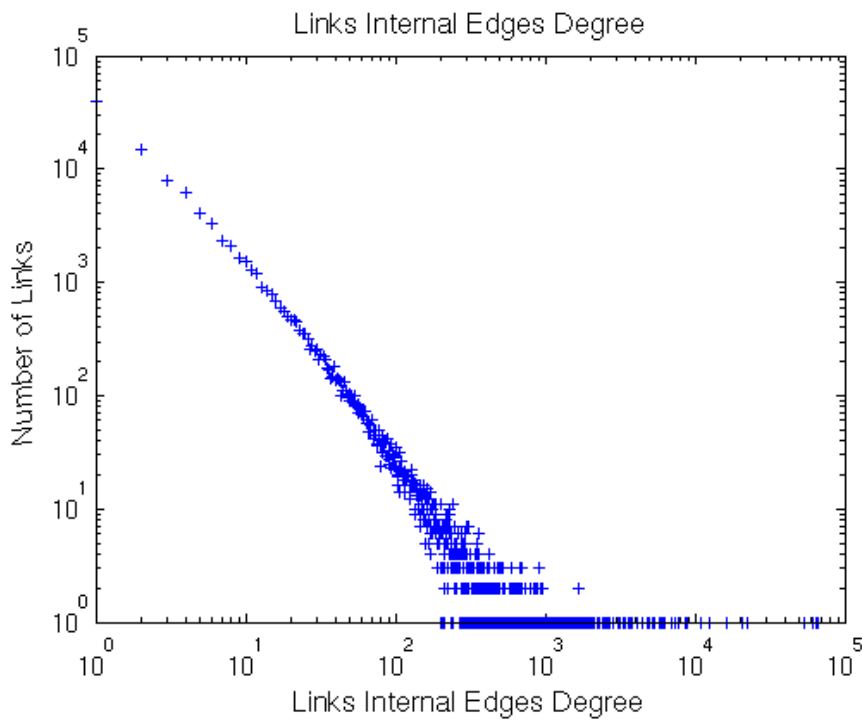
**Figure 5.1: An Internet PoP Level Connectivity Map - A Partial DIMES Map of Week 42, 2010**

## **5.2 Exploring PoP Level Connectivity**

Using a dataset comprised of 478 million traceroutes conducted in weeks 42 and 43 (late October) of 2010, measured by 1308 DIMES agents and 242 iPlane vantage nodes and applying DIMES' algorithm to all the measurements, result in a PoP level map that contains 4750 PoPs, 82722 IP addresses within the PoPs and 102620 PoP level links [101]. The links are an aggregation of 1.98M IP level edges. All the PoPs have outgoing links, with only 2 PoPs having only incoming links and one PoP with no PoP level links (only IP-level). As a full PoP level map is too detailed to display, a partial map is shown in Figure 5.1. The figure demonstrates the connectivity between randomly selected 430 ASes at the PoP level.

Most of the IP edges that are aggregated into links are unidirectional: 96.6%. This is a characteristic of active measurements: vantage points are limited in number and location, thus most of the edges can be measured only one way. However, at the PoP level, 18.8% of the links are bi-directional: six times more than the bi-directional IP edges. This demonstrates one of the advantages of using a PoP



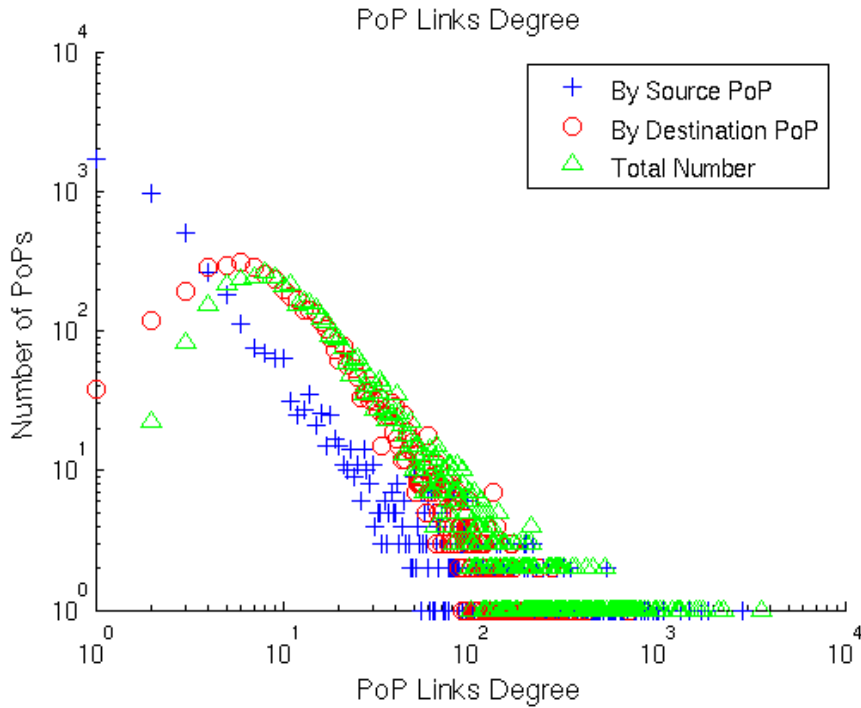


**Figure 5.2: Number of Edges within a Link vs. Number of PoP Level Links in DIMES Dataset**

level map, as it provides a more comprehensive view of the networks' connectivity without additional resources. The average number of edges within a unidirectional link is 6.9, and the average number of edges within a bidirectional link is 72.9. This is not surprising, as it is likely that most of the bidirectional links will connect major PoPs, within the Internet's core and thus be easily detected.

An additional view of edges aggregation into links is given by Figure 5.2. The X-axis shows the number of edges aggregated into a link, while the Y-axis is the number of PoP-level links. The graph shows a Zipf's law relation between the two, as 81.5% of the links aggregate ten edges or less, and less than 2.5% aggregate 100 edges or more. The large number of edges per link is explained by the fact that a measured edge is not a point-to-point physical connection: Take two routers, A & B, connected by a single fiber; If one of the routers has 48 ports, and one measures through each and every port, he will detect 48 edges between the two routers (incoming port  $i$  on router A and the single connected incoming port of router B).

The number of links per PoP also behaves according to Zipf's law, as shown in Figure 5.3. The figure shows the total number of links per PoP, the number of outgoing links (source PoP) and the number of incoming links (Destination port). The connectivity between PoPs is very rich: only twenty two PoPs have one or two links to other PoPs, while 70% of the PoPs have ten or more links to other PoPs.

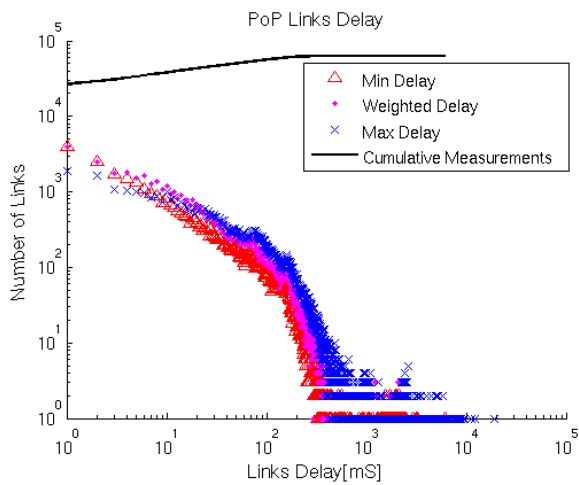


**Figure 5.3: Number of Links per PoP vs. Number of PoPs in DIMES Dataset**

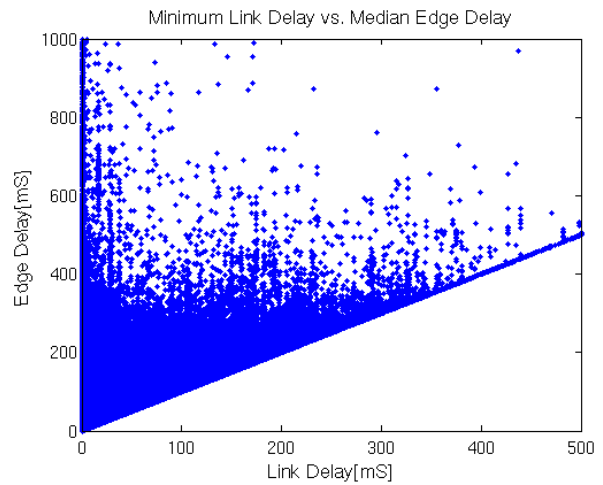
Many of the links are between PoPs that are co-located, which we define as links with a minimal delay of 1mS or less, and over 90% of the PoPs have such links. Almost all the PoPs (over 97%) are connected to PoPs outside their AS.

Figure 5.4 shows the minimum, weighted average and maximal delay per link, plotted on a log-log scale with the delay (X-scale) measured in milliseconds. The solid black line shows the cumulative number of measurements up to a given link delay. We omit from this plot links that include only a single edge, which distort the picture as their minimal, weighted and maximal delay are identical. An interesting attribute of this plot is that all three plotted delay parameters behave similarly and are closely grouped. As all the links are an aggregation of multiple edges, this indicates the similarity in the delay measured on different edges. One can also see that most of the measurements represent a delay of 200ms or less, and that the extreme cases are rare (see the cumulative measurement line). In almost all the cases where a minimal delay of 1sec or more are measured, this is a link that is made of a single edge. The same logic applies also for links with a small maximal delay, meaning the maximal delay was defined by only one or two measured edges. Here, however, a small maximal delay may also indicate co-located PoPs.

Traceroute measurements are known to introduce delay errors [48, 95]. The errors tend to be of an additive nature, though sometimes a measured single-edge delay may be lower than its physical de-



**Figure 5.4: Links Delay vs. Number of Links in DIMES Dataset**



**Figure 5.5: Link Delay vs. Edge Delay in DIMES Dataset**

lay, due to an additive delay of the previous edge in the measured traceroute. This phenomenon is demonstrated by Figure 5.5: The X-Axis of the figure shows the estimated minimal link delay (in milliseconds), and the Y-Axis shows the spread of edge delay measurements. The figure focuses on the interesting range of delays, up to 500mS link delay and one second edge delay. A few measurements exist outside these boundaries, but their contribution to this discussion is small. Figure 5.5 clearly demonstrates the effect of a single edge measurement error: some links have a minimum delay of zero yet some of their measurements reach one second. Thus the aggregation of multiple edges into PoP level links significantly cleans noise from the collected data.

## VI Improving AS Relationship Inference Using PoPs <sup>1</sup>

### 6.1 Background

Inferring AS relationship is an important line of research [35, 107, 26, 98], yet over the years it was conducted mainly on the AS level alone, assuming that the same relationship between ASes is kept in all their peering points.

This chapter proposes a method that accepts as an input a collection of traceroutes and IP to PoP mapping, converts the traceroutes to PoP level traceroutes, and analyzes the ToR at the PoP level. The analysis at this level reveals oddities that help us make several contributions, which can be roughly classified into two classes. First, by looking at Valley-freedom violation we can easily detect imperfections in our data-set inputs: errors in the initial ToR assignment, missing sibling relationships, missing IXP address prefixes, and erroneous IP to AS mapping. Second, using the same method we can identify complex ToRs, a holy grail in the field. An interesting subgroup of complex ToRs we identified are "academic oddities": cases where academic networks do not follow the strict commercial rules of relationships. While some of our findings can be achieved at the IP level, it is important to point out that the analysis at the PoP level dramatically reduces the processing amount.

### 6.2 Analysis Process

We start by converting a traceroute dataset to a PoP level traceroute (preprocessing); then we deduce missing ToRs, based on the ones we have; and finally we flag out anomalous ToRs, some of which are clear suspects of complex ToRs. Some of the anomalies we find in the last stage are errors in our input datasets, which are then corrected for future use. Thus, as we keep using the analysis method periodically, we end up flagging only true anomalies and new changes in the Internet ToRs (like a new merger between two ASes). A detailed description of the method stages is as follows:

---

<sup>1</sup>This part of the work was partially done by guiding Lior Neudorfer towards his MSc thesis

### 6.2.1 Preprocessing

The algorithm receives as inputs:

1. A dataset of IP-level traceroute measurements.
2. A mapping of IP addresses to PoPs and ASes.
3. A dataset providing initial classification of AS-level links ToR: c2p / p2c / p2p / s2s.

The first step of the preprocessing is identifying for each IP address in the traceroutes dataset its corresponding PoP and AS. IP addresses whose AS can not be identified (i.e., internal IP addresses) are discarded. IP addresses whose AS is known but their PoP is not are retained. The next step is identifying the path's AS borders, by finding pairs of consecutive hops which belong to different ASes. All IP-level hops which are not located on AS-AS border links are discarded, as we are only interested in links between different ASes. Finally, the algorithm discards repeating paths and paths which are fully contained within other paths. This last stage reduces the amount of paths, leaving only paths that contribute new information over others.

Traceroute paths may contain PoP loops or cycles, caused by load balancing artifacts, misconfigured routers or measurements taken during routing convergence periods [11]. For any path that contains a loop, the algorithm trims the path's prefix and suffix in order to retrieve the longest possible segment which does not contain a loop. We discard traceroute measurements which, when repeatedly measured, show artifacts of load-balancing routers.

The last step in the preprocessing stage is discarding IXP hops from the traceroutes. As some IXPs appear on traceroute paths as an additional AS hop, they may introduce errors in the following phases. Thus, if hop  $N$  in the traceroute represents an IXP, we drop this hop and stitch hops  $N - 1$  and  $N + 1$  together, forming an AS-to-AS level link. We further discuss the reason and effect of this step in section 6.4.

### 6.2.2 ToR augmentation

The ToR augmentation method, which is based on ideas from [98] and conducted on AS level, assumes validity of the valley-free rule on existing paths and infers new ToRs in a way which preserves

this rule. This assumption is used to assign a ToR to links that have no ToR classification in the initial ToR database (See Section 6.4.1 for the initial ToR database coverage).

To find the ToR of unclassified links, we consider AS-level link paths generated in the preprocessing stage. Only AS-level link paths that have a single unclassified link and that are otherwise valley-free are considered. For each undetermined link in a given path, a vote is cast for each type of ToR which will not violate the valley-free path property: A c2p vote is cast for links which are in the middle of an “uphill” segment or links between an uphill segment and a p2p link. A p2c vote is cast for links which are in the middle of a “downhill” segment or links between a p2p link and downhill segments. For links which are before downhill segments in a path where only a downhill segment is detected, or which are after uphill segments in a path where only an uphill segment is detected, or for links which are located exactly between the uphill and downhill segments, all three possible votes are casted: c2p, p2p and p2c.

After traversing all eligible paths, a new ToR is inferred for cross-AS PoP-links that had no ToR assigned. Such a link is assigned a ToR if the percentage of votes which agreed on a ToR is larger than a VOTING-THRESHOLD, and there were more than MIN-VOTES votes for the ToR. In case that multiple ToRs pass the above thresholds, we give precedence to the p2p ToR. The process is then repeated, taking newly discovered ToRs into consideration, and trying to infer ToR for the remaining unassigned links, until no new ToRs are discovered.

### 6.2.3 Complex ToRs and anomaly detection

A path which is not valley-free and can be corrected by changing a single link’s ToR, is termed a *single-error path*. For example, the path with the ToRs c2p-c2p-p2c-c2p-p2p-p2c can be corrected to c2p-c2p-c2p-c2p-p2p-p2c. Single-error paths always contain one or two links whose ToR can be changed in order to make the path valley-free. These links are denoted *candidate anomalous links*. Each candidate anomalous link has one or two alternative ToRs: the ToRs which if assumed will make the path valley-free. For each PoP-PoP link A-B, the algorithm finds:

1.  $P$ : the group of paths that link A-B is part of.
2.  $n$ : the overall number of unique PoP and AS nodes in the graph created by combining all the

paths that contain link A-B. A large number means A-B was measured by traceroutes with many diverse sources and destinations.

3.  $VP$ : the group of valley-free paths A-B is part of.
4.  $FP_{c2p}$ : the group of paths which are not valley-free, in which A-B is a candidate anomalous link, who can be made valley-free by assuming c2p ToR for A-B.  $FP_{p2p}$  and  $FP_{p2c}$  are defined similarly.

The algorithm outputs anomalous PoP-PoP links which satisfy the following conditions:

- The link has a minimal measured graph size ( $n > min-nodes$ )
- The percentage of valley-free paths containing the link is smaller than an arbitrary *min-valid-percentage* ( $|VP|/|P| < min-valid-percentage$ )
- There is a new ToR which, when assumed for the link, turns a large percentage of paths to be valley free ( $|FP_{ToR}|/|P| > min-fixed-percentage$ )

The three conditions capture cases when a PoP-PoP link has a significant evidence for a problem (first two conditions) and a fix in the link ToR, which seems to correct the problem. The algorithm also outputs the set of PoP-level links which comply with the first two rules, but for which a new ToR could not be determined with a high level of confidence.

### 6.3 Datasets

Three types of datasets are used for this study:

**DIMES traceroutes** All the traceroutes measurements are taken from the DIMES project [25] The dataset includes 29.2 million traceroute measurements and 506.3 million IP-level hops. The measurements targeted 2.39 million destination IP addresses and were collected by 1017 DIMES agents. RouteViews [117] and WHOIS databases were used to infer every IP address to an AS.

**DIMES PoPs** The DIMES IP to PoP mapping dataset is taken from weeks 19 and 20 of 2012. The mapping of IP to PoP was based on traceroutes taken by both DIMES and iPlane [72] over the same period of time. The map contains 5215 PoPs and 98650 IP addresses in 2636 different ASes.

**CAIDA ToRs** The initial AS ToR mapping dataset is taken from CAIDA's AS Rank Website<sup>2</sup> from August 2012. The dataset relies on BGP paths obtained on June 2012. It contains ToRs for 119,924 AS couples. 76781 (64%) relationships are customer/provider relationships, 40,900 (34%) are peering relationships and 2243 (2%) are sibling relationships. We compare our results with a newer CAIDA dataset, published September 2012.

## 6.4 Results

### 6.4.1 Preprocessing Results and ToR augmentation

The preprocessing stage of the algorithm takes the 29 million IP level traceroute measurements and turns them into 1.63 million unique PoP level paths, thus reducing the dataset size by an order of magnitude. 1.48 million PoP-level Paths (91%) are valley free.

Out of the 70714 AS-AS links found in the dataset, only 45202 links (64%) were covered by the CAIDA dataset. It is thus necessary to augment the ToR dataset. We complete the missing ToR for 6699 links and fail to complete 18813 AS-AS links, out of them 495 appear only on paths which are not valley free. Links of unknown ToR which appear only on paths which are not valley-free can not be assigned a ToR with a high level of confidence. The augmentation increases the number of customer-provider PoP links but only slightly increase the number of peer-peer links. For the ToR voting, we use a VOTING-THRESHOLD of 80%, which gives a high level of confidence that the inference is correct. We select this value based on experimentation with a range of values and find that the effect on the results is marginal. Further information is omitted due to space limitations.

The ToR augmentation method requires a minimal number of paths in which the inferred AS-AS link is included and that are valley-free, the MIN\_VOTES threshold. This parameter is required as inferring ToRs according to a few paths might introduce errors due to wrong traceroute replies or wrong AS prefix resolution, similarly to the phenomena described by Zhang *et al.* [131]. We tested a range of MIN-VOTES values, in order to select the best threshold and to verify sensitivity. Setting MIN-VOTES= 5 infers 6699 new ToRs, while MIN-VOTES= 3 helps inferring 8594 new ToRs. However, for a large number of AS-AS links there is only a single applicable path (regardless of the valley free rule), which makes the augmentation difficult. Under such conditions we do not attempt to infer

---

<sup>2</sup><http://www.caida.org/data/active/asrelationships/>



the ToR. For lack of space we omit further discussion of MIN-VOTES sensitivity. We eventually set MIN-VOTES= 5 which is a high confidence threshold, and allows to infer 26% of the missing ToRs.

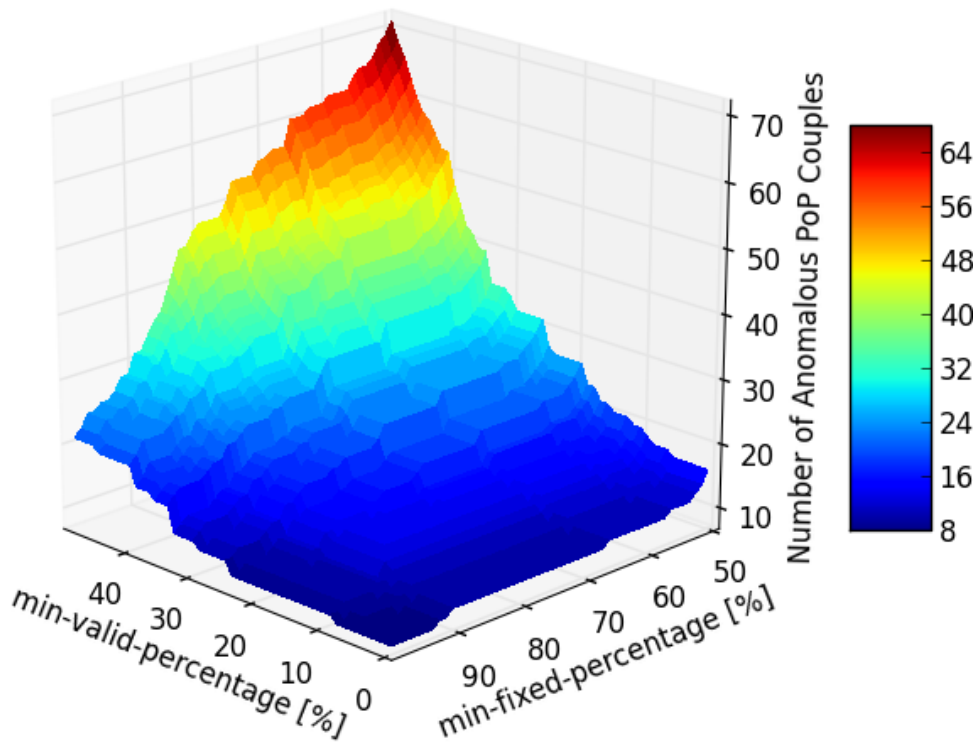
#### 6.4.2 Sensitivity analysis

Two parameters affect the anomaly detection method. The first, *min-valid-percentage*, determines the minimal percentage of valley-free paths required to consider the PoP-PoP ToR correct (as detailed in Section 6.2.3). The second parameter, *min-fixed-percentage*, determines the minimal percentage of valley-free paths after the ToR was replaced required to consider the new PoP-PoP ToR correct. We evaluate the effect these two parameters have on our anomaly detection method's results.

*min-valid-percentage* and *min-fixed-percentage* capture the amount of confidence we wish to achieve in determining whether a specific PoP-PoP link is anomalous. A larger *min-valid-percentage* may cause non-anomalous links to appear as anomalous, but can also lead to the discovery of anomalous links that by chance did not consistently cause path invalidity. A low *min-fixed-percentage* threshold marks PoP-PoP links that even after changing their ToR appear as candidates to be anomalous due to non valley-free paths. This may happen when some of the paths contain other errors, such as traceroute measurement errors resulting from wrong AS resolution or ToR errors on other AS-AS links.

To study the sensitivity to thresholds, we omit anomalies that turn out to be errors in the original AS ToR database or that are caused by IXPs. This is done as these are one-time corrections and do not affect the algorithm in later runs. Figure 6.1 shows the effect of changing the two parameters on the number of discovered anomalous PoPs, with *min-nodes* set to 10 nodes. For the purpose of sensitivity study, we consider as anomalous PoPs only PoPs that fall under the categories of complex AS relationships and odd academic ToRs (see below). Clearly for a large range, between 0% and 35% for the *min-valid-percentage* threshold and between 70% and 100% of the *min-fixed-percentage* threshold, there is little change in the number of discovered anomalies. Thus, We select the thresholds from the non-sensitive region: *min-valid-percentage*= 20% and *min-fixed-percentage*= 75%.

An interesting observation is that eight PoP couples are "perfect anomalies": they appear in no valid PoP paths, but when changing their ToR all the paths in which they appear become valley-free.



**Figure 6.1: Anomaly Detection vs. Thresholds Values.**

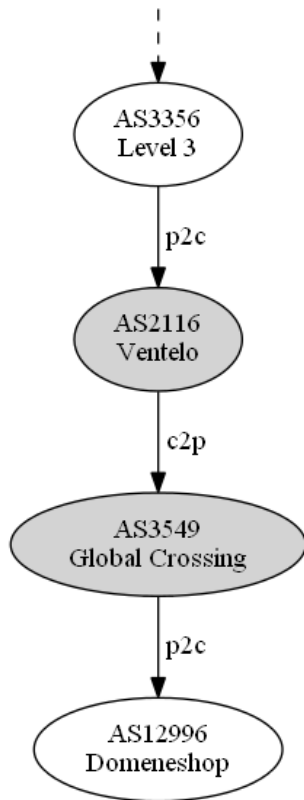
### 6.4.3 Anomaly detection

After the first execution of the anomaly detection algorithm, we detect a couple of dozens anomalies. We classify these anomalies into seven categories and highlight specific cases that exemplify the anomaly type:

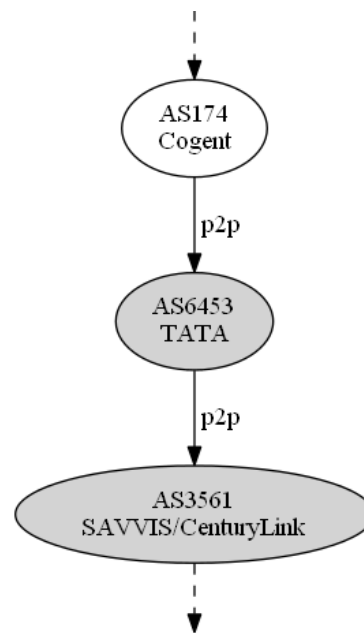
#### 6.4.3.1 AS Prefix Resolution Errors

Our anomaly detection method detected three cases that were attributed to AS prefix resolution errors. In these cases, the corresponding AS for a specific IP address in a traceroute measurement was incorrectly resolved by the RouteViews dataset. This caused a large percentage of the paths which contained this address to contain a valley, as the ToR between the wrongly assigned AS and its neighbors was incorrect. AS prefix resolution errors might occur when the BGP blocks that were announced to RouteViews were incorrect or not updated. Closer inspection, using other tools including WHOIS, revealed the true owner of the IP address. Assessing the accuracy of multiple IP to AS resolution databases is outside the scope of this paper.

Figure 6.2 demonstrates this phenomenon. In this case, an anomalous link is detected between AS2116 (Ventelo) and AS3549 (Global Crossing), AS3549 is the provider according to CAIDA. In



**Figure 6.2: AS Prefix Resolution Error - Example**



**Figure 6.3: Complex AS Relationship - Example**

all the paths that contained this link, it appeared after a link between AS3356 (Level 3) and AS2116 (Ventelo), which is a p2c link, creating a valley in these paths. However, using WHOIS it was discovered that the IP prefix to AS mapping was wrong, and that the PoP first associated with AS3549 actually belongs to Domeneshop (AS12996), which is a customer of Ventelo.

#### 6.4.3.2 IXP and sibling detection

Usually, when IXPs appear in traceroute paths it is as an additional IP hop. In ToR analysis they should be removed or else introduce errors since they are not part of the AS hierarchy, which we did in our preprocessing stage using lists of known IXPs. However, we have found six IXPs that appeared as anomalies in our PoP level traceroutes. Finding IXPs and consequently other anomalies is an incremental process, as each detected IXP allows more paths to become valley-free (due to their omission).

Similarly, we detected wrongly inferred siblings relationships. These are often cases of one ISP taking over a second ISP, which was previously its customer. This change of ToR is not always updated in the ToR dataset. Thus, when checking valley free routing, some of the paths between the pair of

ASes will remain valid as c2p, while others will only be valid as s2s. Since in many routes a s2s ToR is interchangeable with c2p ToR, the change of ToR between the two ASes may be hard to detect. We manage to find 6 wrongly inferred s2s relationships, e.g., between TelePacific (AS14265) which acquired Mpower (AS18687).

#### 6.4.3.3 *ToR inference errors*

On three cases, a PoP-PoP link was deemed anomalous, but closer inspection revealed that the ToR for the corresponding AS-AS link was wrongly inferred by CAIDA. In general, the method tries to avoid flagging such cases as anomalies. It does so by discarding anomalous candidate links for which the confidence for corresponding ASes' ToR is not high enough. We deem two ASes' ToR as confident if there is a majority of paths containing the AS-AS link which follow the valley-free rule.

Two of the three cases we've discovered were corrected in CAIDA's September 2012 ToR dataset, a few weeks following this analysis. In the third case, CAIDA inferred a peering relationship between two ASes (AS12389 and AS8359), while in our measurements almost half of the paths which contained this AS-AS link were not valley free.

One exemplary case of a wrongly inferred ToR, CAIDA inferred the AS3561-AS4134 (SAVVIS-Chinanet) as a peering relationship. Our algorithm detected specific PoPs belonging to these organizations as anomalous, and suggested a p2c relationship instead. The PoPs were deemed anomalous as there was a small majority of paths containing PoP couples from AS3561 and AS4134 (80 out of 145 paths) which were valley free. In September 2012 CAIDA updated this ToR in their dataset and changed the relationship between the two ASes to p2c, same as suggested by our algorithm.

#### 6.4.3.4 *Complex AS relationships*

An interesting relationship was found between two PoPs of AS3561 (SAVVIS/CenturyLink) and AS6453 (TATA) in Canada. As both ASes are Tier-1 providers, the assumed ToR between them is a peering relationships (also indicated by CAIDA). However, only three out of the sixteen unique PoP paths that include a link between this pair of PoPs are valley-free. Out of the remaining 13 paths, 11 traverse a PoP link between AS174 (Cogent) and AS6453 (TATA), clearly another p2p link between tier-1 ASes (see Figure 6.3). When assuming a c2p relationship (the provider being AS6453's PoP),

all paths are valley-free.

It seems that while CenturyLink and TATA have a peering relationship in most locations, this specific PoP-PoP link is configured differently: TATA's specific PoP provides transit services between other Autonomous Systems (namely, Cogent) and SAVVIS.

We have revalidated this finding a couple of weeks after the original experiment's date, by running a dedicated DIMES experiment that, issuing a large amount of traceroute measurements towards the specific IP addresses of these PoPs from many widely spread vantage points. The phenomenon was also reproducible by issuing traceroutes from Cogent routers, using their Looking Glass service.

#### 6.4.3.5 *Odd Academic ToRs*

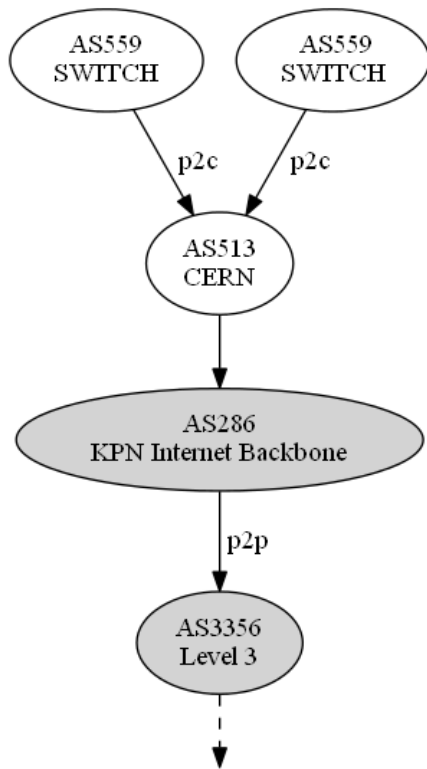
A couple of ToR anomalies are discovered in research institutes' affiliated PoP links. Research institutes are less driven by commercial incentives and tend to be more collaborative in nature, thus setting their ToR criteria differently than most ASes.

Figure 6.4 shows one such case, involving multiple PoPs belonging to research organizations. Traffic flowed from multiple PoPs belonging to SWITCH, the Swiss Education and Research Network (AS559) through CERN (AS513) and then through KPN (AS286), finally reaching the tier-1 provider Level3 (AS3356). According to the ToRs inferred by CAIDA, SWITCH is a provider of CERN, and KPN is a peer of Level3, causing this path to be non valley-free. CAIDA's dataset missed information on the ToR of CERN and KPN, but for any ToR this path violates the valley-free rules.

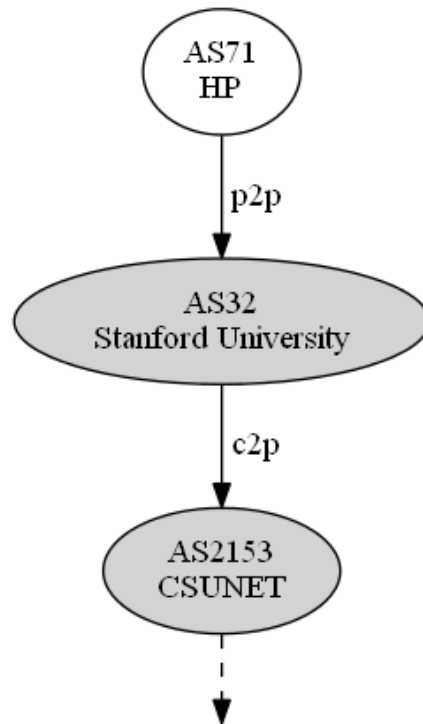
An additional anomaly, shown in Figure 6.5, is a single HP (AS71) PoP that is connected to the Internet via Stanford University (AS32) and CSUNET (AS2153). CAIDA's ToR for the HP-Stanford link was p2p, and the Stanford-CSUNET link was c2p (Stanford is the customer), resulting in a clear anomalous link.

#### 6.4.3.6 *Traceroute errors*

We have found three cases in which we believe detected anomalies were probably caused by wrong router replies. In these cases, a specific IP address which was part of a reported traceroute path was not the actual IP address traversed by traffic on this path. This is caused by ICMP replies that are



**Figure 6.4: Academic ToR Anomaly - First Example**



**Figure 6.5: Academic ToR Anomaly - Second Example**

sent through a different interface than the one the packets actually went through [59]. This error may result in a wrong AS resolution, leading to wrongly assumed non-valley free paths.

#### 6.4.3.7 Unresolved Anomalies

Three of the anomalies we found remain unresolved. While we have assumptions for the nature of these anomalies, we did not manage to corroborate our finding and thus prefer to declare them unclassified.

#### 6.4.3.8 ToR Datasets

AS ToR datasets fail to capture a large amount of AS links, due to their reliance on specific data sources. For example, CAIDA’s AS Relationships Dataset [6] only uses BGP routes in order to infer AS ToRs. Shavitt and Shir [97], and more recently Gregori *et al.* [46], showed that BGP and traceroute measurement sources complement each other. Therefore, our augmentation of an existing AS ToR dataset according to an additional source of traceroute measurements is important by itself.

#### 6.4.3.9 *ToR inference at different layers of aggregation*

To better understand the contribution of PoP level maps to ToR research, the disadvantages in using other levels of aggregation should be discussed in comparison.

For many years, using the AS level graph to infer ToRs seemed to be the right way, as this is seemed to be the level at which ToRs are defined. In addition, the AS graph is relatively small and easy to study. However, the AS level treatment does not allow the inference of complex relationships, where two ASes have different relationships in two different locations. As demonstrated by Dimitropoulos *et al.* [26], ASes might have a more complex relationship in various peering points. In addition, most existing algorithms use specialized methods for sibling relationship detection, which rely on data sources other than BGP and traceroute measurements [6].

Using router or IP level maps for complex relationship detection is also not a good solution as it is hard to identify in them scattered errors and anomalies. IP level paths introduce noise to the measurements and cause anomalies to be dispersed over multiple IP addresses, diminishing their significance and preventing their accurate detection. In addition, router and IP level datasets are very large and require considerable processing resources.

PoP level maps provide an answer to the above issues and propose a better level of aggregation than AS, Router or IP level for anomaly detection. If two ASes have different relationships in two different locations, these will be represented by two distinct PoP-PoP links, and one of them will clearly violate the valley free rules, and thus can be easily flagged. While the same information will also be detectable on the IP/router level, it will be hard to correlate it to a specific location and to discard local errors. Considering the same problem the other way around, when detecting on the IP/router level multiple non valley free routes it is hard to understand the nature of each link's anomaly or error. The aggregation of multiple IP/router level links to a single PoP-PoP link reduces the complexity of this issue considerably, and provides a higher level of confidence to the inferred new ToR.

#### 6.4.3.10 *Dataset Size Dependence*

ToR Errors and PoP-PoP link anomalies are more likely to be found as we increase the number of measurements and diversify the measurement vantage points. When measured from a small number

of sources, an anomaly might not be identified, since a single path might not violate valley-freedom even with the existing error or anomaly. For example, if a ToR is inferred as p2p instead of c2p, it might not be discovered if the link resides between a c2p link and a p2c link. These paths would be valley-free in both cases, and our method - which looks for improvement in the percentage of valley-free paths when assuming a different ToR - would not identify this anomaly.

It is important to note that anomalies detected in a given dataset on the PoP level will remain valid even if the dataset grows considerably, since the threshold to flag an anomalous ToR is based on the number of violating PoP level paths and not their percentage.

#### *6.4.3.11 Validation*

The validation of our method is a hard task. Except for verifying results with ISPs, which are reluctant to cooperate, there is no single ground truth dataset. As we show, many of the datasets that we use as a reference have errors. Some of our results are corroborated by corrections done to the CAIDA dataset shortly after we ran our analysis. Another mean of validation is from ISPs websites and public information. This applies mainly for siblings ToR validation, often caused by one ISP acquiring another.

Another method of validation is using targeted measurements through many scattered vantage points to the point of anomaly. This is intended to eliminate transient routing effects and to confirm the anomaly through as many distinct paths as possible. For some anomalies, such as mistakes in AS resolution, reverse DNS and WHOIS, are useful tool in finding the true IP to AS mapping.

We believe that the level of validation provided in this work is sufficient under the given lack of ground truth conditions and as the results show, it provides a good mean to validate other datasets and sources for ToR information.



## VII Setting the Foundations for PoP-Based Internet Evolution

### Models

#### **7.1 Background**

One of the dreams of mankind has always been to be able to predict the future. In scientific terms, this corresponds to the mathematical description of patterns found in real world data in order to devise models that can be used to predict future events. Researchers have pursued a similar goal over the past decade in the area of Internet modelling and forecasting while using Autonomous System (AS) level maps. Efforts have first been focused on obtaining topological maps of the Internet, principally at the Internet Router (IR) and at the AS granularity levels. In both the IR and AS cases, Internet maps are usually viewed as undirected graphs in which vertices represent routers or ASes and edges represent the physical connections between them. Several large-scale measurement projects have started to go beyond these purely topological characterizations of the Internet's properties, and to tackle the characterization and modeling of the relationship between economic factors and Internet evolution. The promised forecast capabilities however have not yet been achieved due to the lack of sufficient data and the difficulty of integrating Internet data with geographical and economical data at a planetary scale.

In the previous chapters we have described a set of measurement tools and algorithms to obtain PoP-level Internet maps. The PoPs, just like small ISPs, have a strong geographical grip and can better represent the Internet evolution.

In the following section, the PoP topologies of the Internet are annotated with geographical, economical and demographical information to achieve an understanding of the dynamics of the Internet's structure, in order to identify the constitutive laws of Internet evolution. These can be used to develop a realistic topology generator and a reliable forecast framework that can be used to predict the size and growth of the Internet as economies grow, demographics change, and as-yet unattached parts of the world connect.

The combination of the technological infrastructure with monetary aspects can provide an understanding of the forces driving the data-communications industry today. Using tools and methods from the field of complex science (for example, from statistical physics) it is theoretically possible to develop a prediction model. The practical uses of an evolution model are numerous: Internet service providers can leverage the model to decide whether to expand their PoP, upgrade its technology or build a new point of presence. City planners can predict its required infrastructure and assign resources for it in advance. Telecommunication firms and semiconductor corporates can better plan their next generation of product and adapt its schedule and features to the market needs. Last, the growth and strength of developing countries can be assessed and predicted, providing country and world level decision makers with essential information in times of economic crisis and market instability.

In this section, we set the infrastructure for a development of a future evolution model. As discussed next, the information required for the development of such a model is yet out of reach. Thus, the following chapter surveys the relationships between PoPs and economic and demographic aspects, but only over specific time periods. Using this information, a model can be developed once more information is gathered as years go by.

## **7.2 Datasets and Datasets Limitations**

Several types of datasets are used in conjunction in this work. First, we use DIMES's PoPs dataset. Two PoP level maps are selected, one from 2012, and one from 2010. These are the same maps described in Section 4.5. PoP level maps from earlier years lack information, either due to the extent of the measurements, their accuracy or the lack of geolocation data from that time. We note that geolocation data is likely to change over the years, as discussed in section 4.3.4, which may lead to inaccurate PoPs geolocation.

Second, we use the World Bank's World Development Indicators (WDI) [12] from May 2012. This dataset contains a collection of development indicators, compiled from officially-recognized international sources. It presents the most current and accurate global development data available, and includes national, regional and global estimates. The dataset is on country level and it contains indicators such as population and population's growth, GDP, percentage of Internet users and more (total of 1287 parameters per country) on a yearly basis, from 1960 and up till 2012.

Considerable amount of information is gathered from census data. To this end, several census sources are being used. The United States Census Bureau [7] provides several types of USA census information. It collects population and housing information every 10 years, conducts an economic census every 5 years as well as smaller surveys and indicators released annually or several times a year. IPUMS [92] is a project dedicated to collecting and distributing of United States and international census data. It provides harmonized data for free in a manner that eases that analysis process.

In order to study the effect of transportation infrastructure, we focus on the United States and use the Department of Transportation's Bureau of Transportation Statistics [116] to retrieve information on highways infrastructure, busiest airports and more. The bureau provides transportation related economic information as well as connectivity and economic factors. A main source for economic information is the Bureau of Economic Analysis in the US Department of Commerce [115], which provides information on aspects such as GDP and income.

The population of cities is obtained from MaxMind's World Cities database [77], which includes information on the population of most of the world's cities as well as their geolocation. We note that some of the US level databases also include information about the population, but as the size of the population differs from dataset to dataset due to different definitions of a city or a metropolitan area, we stick to the same population dataset across the entire analysis.

### *7.2.1 Datasets Limitations*

As the ultimate goal of the study of PoPs evolution is to come up with a realistic evolution model, the datasets at hand put restrictions and sever limitations on the ability to develop the model over short time periods. First, WDI dataset is only on country level and not on city level, thus it can not be used to the city level modelling intended by this work. International census data is mostly provided on country level and therefore has the same limitation. Census data poses an additional problem, as census is conducted only once every few years (usually five to ten years) and thus does not allow modelling over shorter time periods. As the Internet changes rapidly and technologies emerge and die within a decade, such time frames are not useful. Large portions of the US datasets are provided on state level, and only partial information is available on metropolitan level. The lack of per-city information limits the coverage of PoP's cities in the development of an evolution model over time.

The PoPs dataset limits the development of the model as well. First of all, due to the nature of the

PoPs extraction model, it is not possible to tell whether two PoPs of the same AS in the same city are truly separate or are part of the same PoP that was divided due to a missing measurement of an inner link <sup>1</sup>. For this reason, most of the analysis is based on the number of aggregated PoPs per AS in a given city, with the information on the total number of PoPs and IP addresses within these PoPs observed but largely not used as an indicator. While this may work well in most of the western world, in other regimes the number of competing ISPs is limited or the government controls the communication market (e.g. in Syria [100]). In such countries the number of PoPs per city is limited by these external forces and the study of evolution is inaccurate. Last, and most important, there is no ground-truth dataset of PoPs - neither on country nor on city level. This complicates the validation of this work and mostly limits it to information shared by several specific ISPs.

### **7.3 Analysis**

#### *7.3.1 The Relation between PoPs and Population*

Points of presence are likely to be closely related to economic factors of their area of residence. For example, areas which are densely populated are likely to have more service providers than small towns. We examine here the correlation both at the country level, which was done before (e.g. [128, 69, 23]), and at the city level, which we are the first to do

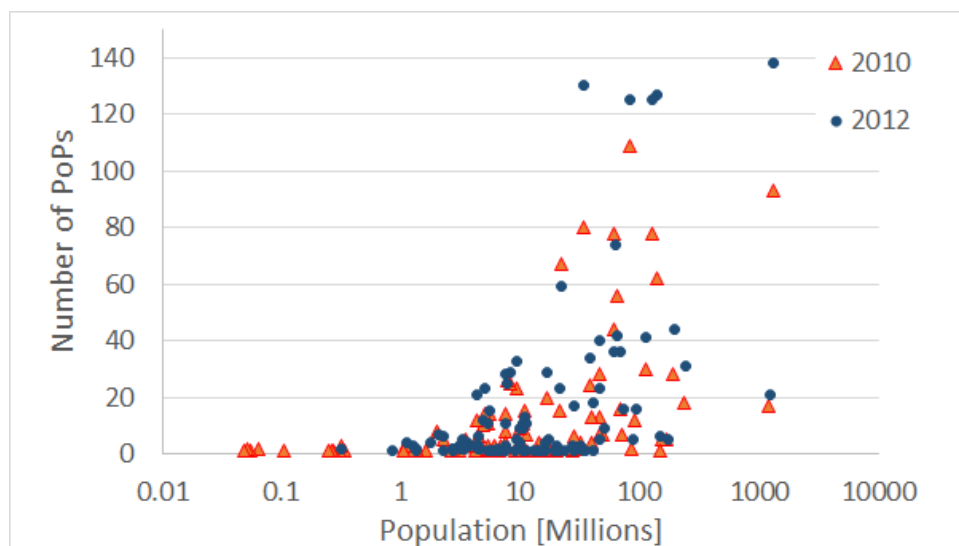
Figure 7.1 shows the number of PoPs discovered on country level compared to the country's population (in millions of people). For clarity, the figure omits the US from the chart, as it is on a different scale. As can be seen, the size of the population is not a strong predictor for the number of PoPs in a country. The correlation coefficient for population to number of PoPs is 0.22-0.23 both in 2010 and 2012. To demonstrate this point further, we present in Table 7.1 the number of PoPs per country compared to its population for a set of selected large countries, both for 2010 and 2012. The country with most PoPs discovered in 2010 is the US, followed by Germany, China, Canada and Japan. In 2012 the list is led by the US, followed by South Korea (Republic of Korea), China, Canada, Russia and Japan. We observe a large growth in the number of PoPs in South Korea and Japan, whereas in countries such as Germany the number of detected PoPs in 2012 is larger than in 2010, yet in overall it is less than in other countries. On the other hand, highly populated countries such as India, Indonesia and

---

<sup>1</sup>Refer to section 3.1 for more details

Country	2010		2012	
	Population	PoPs	Population	PoPs
United States	306.8M	850	309.3M	1203
Germany	81.9M	109	81.8M	125
South Korea	49.4M	46	50M	170
China	1331M	93	1338M	138
Russia	141.9M	62	141.9M	127
Canada	33.7M	80	34.1M	130
Japan	127.5M	78	127.5M	125
United Kingdom	61.8M	78	62.2M	74
Australia	21.9M	67	22.3M	59
Indonesia	237.4M	18	239.9M	31
India	1207M	17	1224M	21
Pakistan	170.5M	5	173.6M	5
Bangladesh	147M	1	148.7M	6

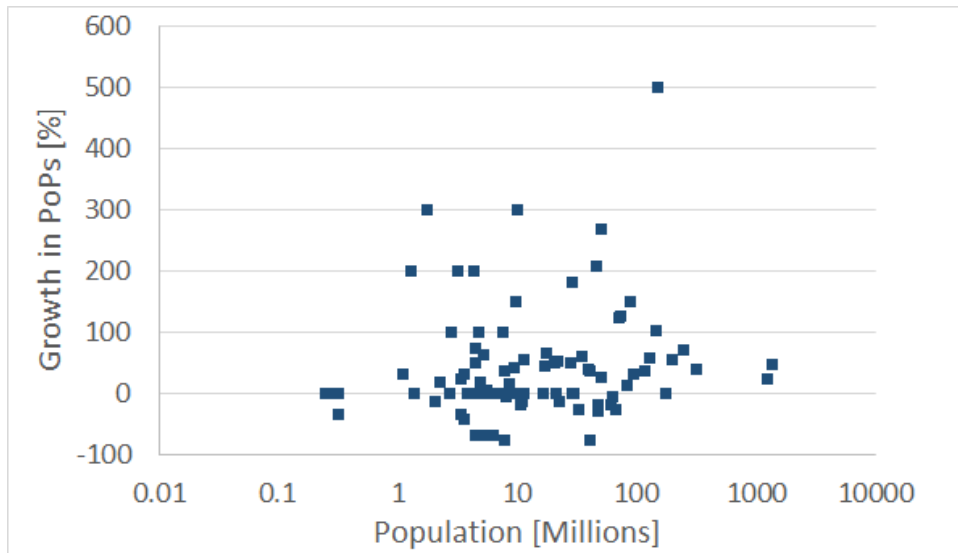
**Table 7.1: Population vs. Number of PoPs on Country Level (Selected Countries)**



**Figure 7.1: Country Level Population vs. Number of PoPs**

Bangladesh have very few PoPs. While the number of PoPs does increase between 2010 and 2012, these countries are still lagging behind other large countries. We note that in Pakistan the number of detected PoPs is not only small (5) but also does not change over time. On the average, the number of PoPs grew by 38% between 2010 and 2012 per country, and in 15% of the countries the number of PoPs doubled itself, as shown in Figure 7.2. We note that in many of these countries only a handful of PoPs was discovered in 2010. One of the exceptions is South Korea, that had 46 PoPs in 2010 and more than tripled this number in 2012.

A different observation is gained by looking at the population versus the number of PoPs on city level. Figure 7.3 shows the number of PoP discovered on city level compared to the city's population



**Figure 7.2: Country Level Population vs. Growth in Number of PoPs**

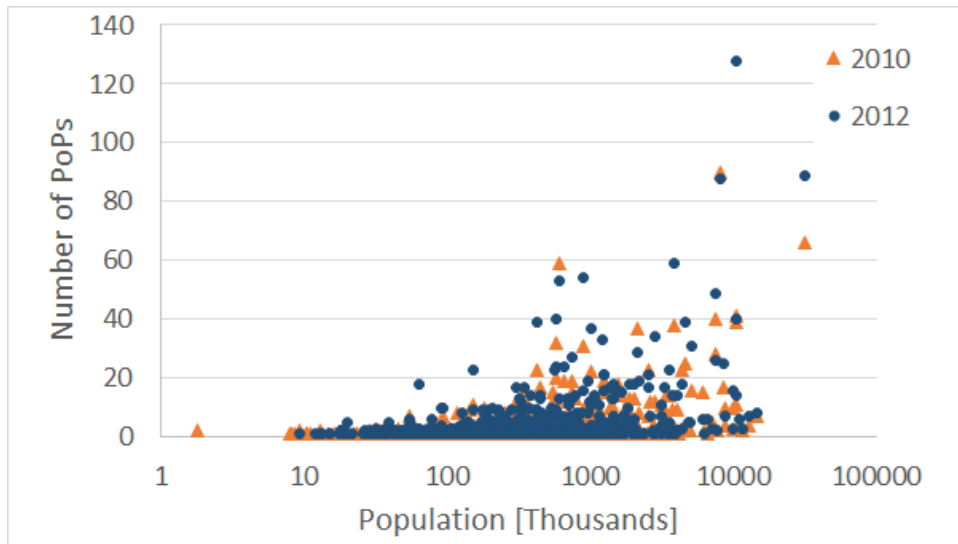
City	Population	PoPs 2010	PoPs 2012
Seoul	10.3M	41	128
Tokyo	31.5M <sup>a</sup>	66	89
New York	8.1M	90	88
Los Angeles	3.9M	38	59
London	7.4M	40	49
Moscow	10.4M	39	40
Paris	2.1M	37	29
Shenzhen	10M	22	37

**Table 7.2: Population vs. Number of PoPs on City Level (Leading Cities in Selected Countries)**

<sup>a</sup>Refers to Tokyo metropolitan area

and a set of leading cities by PoPs in selected countries is shown in Table 7.2. The city with most PoPs discovered in 2010 is New York, followed by Tokyo, Baltimore, Seoul and London. In 2012 Seoul takes the lead with 128 PoPs, followed by Tokyo, New York, Los Angeles and San Jose. In all cases we count PoPs belonging to distinct service providers. In several cases the number of PoPs is decreased between 2010 and 2012, which may be due to lack of measurements, but is also possibly caused by the acquisition or merging of some ISPs. The correlation coefficients for a city's population and the number of PoPs are 0.49 (2010) and 0.51 (2012).

We study the inflation in the number of PoPs in Seoul and find that there are two reasons for that. First, recall that all AS-level PoPs are aggregated on city level, thus we do not count the same autonomous system more than once. When examining the active ASes located in Seoul we find that most of them belong to universities: While in 2010 only three ASes that belonged to universities were detected, in



**Figure 7.3: City Level Population vs. Number of PoPs**

2012 there are 56 such ASes. ASes that belong to other educational and technological institutes are added on top of that, leading to a total growth from 15 to 83 PoPs. While these results raise suspicion regarding the accuracy of the PoPs' geolocation and possible mistakes in geolocation databases, this was found not to be the case, except for a few cases that apply to Seoul's suburbs, and it was manually corroborated that the results are true <sup>2</sup>.

In most of the capitals and large cities of the developed world, tens of PoPs are detected, but in some of the most populated cities of the world, such as Bombay, Manilla and Delhi only a handful of PoPs are detected. While the number of PoPs discovered in these cities grows between 2010 and 2012, it does not significantly change: in Bombay the number of PoPs grew from 4 to 7, in Delhi from 2 to 6 and in Manilla from 9 to 14. In comparison, the number of PoPs in Seoul grew from 41 to 128 and in Los Angeles from 38 to 59. While one may attribute this to the number of Internet users in a country, correlating the number of Internet users or the percentage of Internet users in a country to the number of PoPs is not a good indicator either (see Section 7.3.3). In addition, while the number of PoPs depends on the number of measurements in a target country, in practice this effect is small, as DIMES and iPlane try to reach all possible IP prefixes. PoPs are also detected in small cities, such as Larnaca, Cyprus (less than 50K inhabitants).

Comparing the number of PoPs to the population is somewhat misleading, as countries are considerably different from each other, and one can not compare, for example, the United States to Bangladesh based on population alone. For this reason, we break down city level analysis and conduct it per coun-

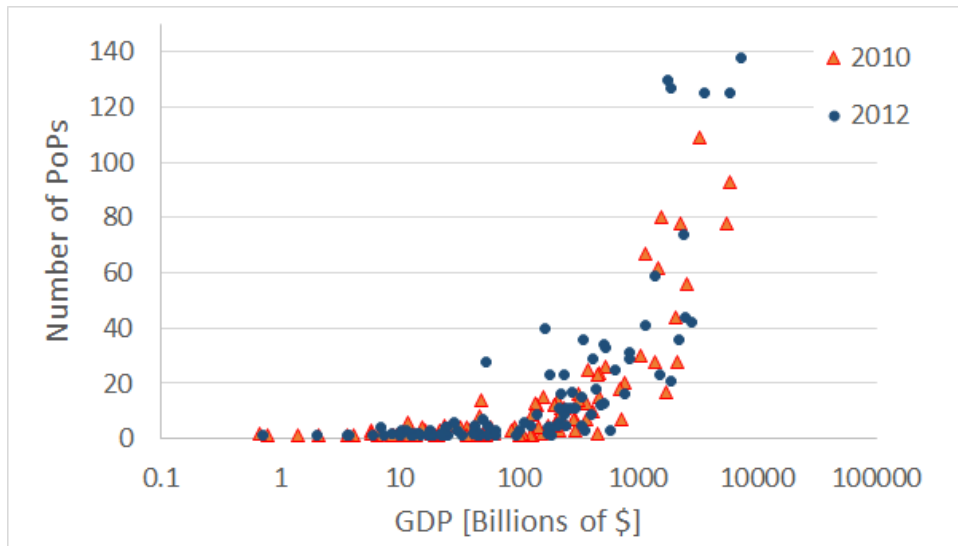
<sup>2</sup>We'd like to thank Dr. Jong Hun Han for his assistance on this subject

try. We include in the analysis only countries where PoPs were detected in at least 5 cities, and check whether the number of PoPs in a city corresponds to the city's size. The dataset, based on 2012 PoP level map, includes 24 such countries and 508 cities. For each country we rank the cities by population and check if the ranking by PoPs' number is identical. We find that for 15 countries the rankings of population and PoPs match, assuming that we allow disregard difference of up to two PoPs difference, since it is negligible. A different view on this aspect is gained by binning. The number of PoPs per city is divided to three bins: 5 PoPs or less, 6-10 PoPs and more than 10 PoPs. Cities are also divided to small cities (100K residents or less), medium cities (100K-1M residents) and large cities (1M residents or more). We find that using this binning, 21 of the 24 countries have a full match between the ranking of PoPs and the population's ranking. This means that on a country level, the size of a city is an excellent indicator to the number of PoPs in it, but the ratio between number of PoPs and population varies between countries. The three countries that do not match this observation are the United States, Italy and Germany. In Italy, Bologna has 6 PoPs, while in larger cities like Turin and Naples only 3-4 PoPs are detected. In Germany, significantly more PoPs are discovered in Frankfurt (24) compared to Berlin (4) and Hamburg (6) despite the later being more than twice its size. In the US many such cases exist, possibly because in many cases the PoP is located in a small town close to a large city. While the anomalies can be explained by other factors, the population is shown not to be the only indicator to determine the number of PoPs per city.

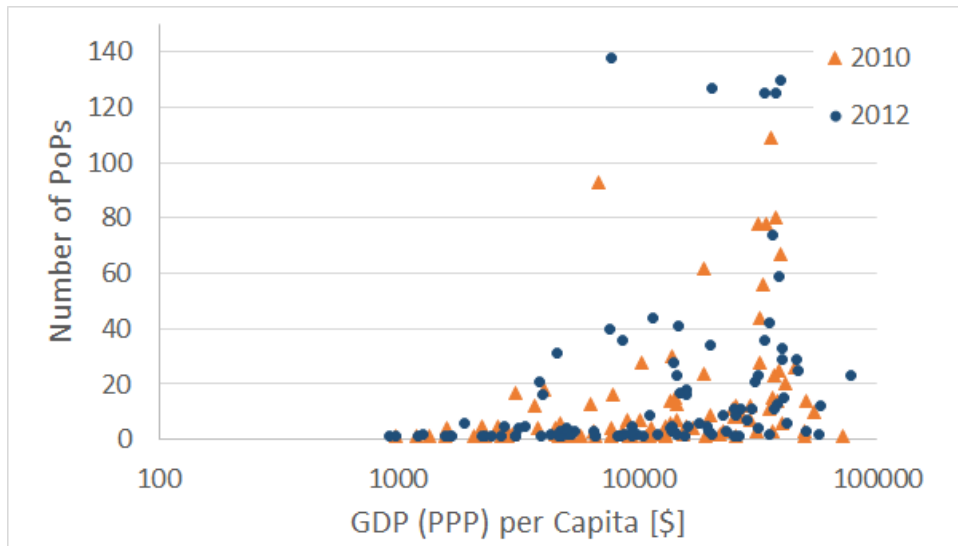
### *7.3.2 The Relation between PoPs and GDP*

The GDP of a country is a good indicator to its number of PoPs. There is a clear relation between the GDP and the number of PoPs, as shown on Figure 7.4. The figure shows on country level the number of PoPs per country compared to its GDP for 2010 and 2012 datasets; for the 2012 dataset we used the GDP reported at the end of 2011 (as published in WDI's May-2012 dataset). As the figure shows, high GDP leads to a high number of PoPs on the country level. The correlation coefficient between the GDP and number of PoPs is very high: 0.92 in 2010 and 0.90 in 2012. For countries with a GDP of 100's of billions of dollars, this is clearly the trend but it is not always the case. For example, Sweden and Saudi Arabia have almost the same GDP (538 and 577 billions of dollars, respectively) yet in Sweden we detect in 2012 thirty three PoPs, while only three PoPs are detected in Saudi Arabia. For this reason, a simple equation that shows the relation between the GDP and number of PoPs can





**Figure 7.4: Country Level GDP vs. Number of PoPs**

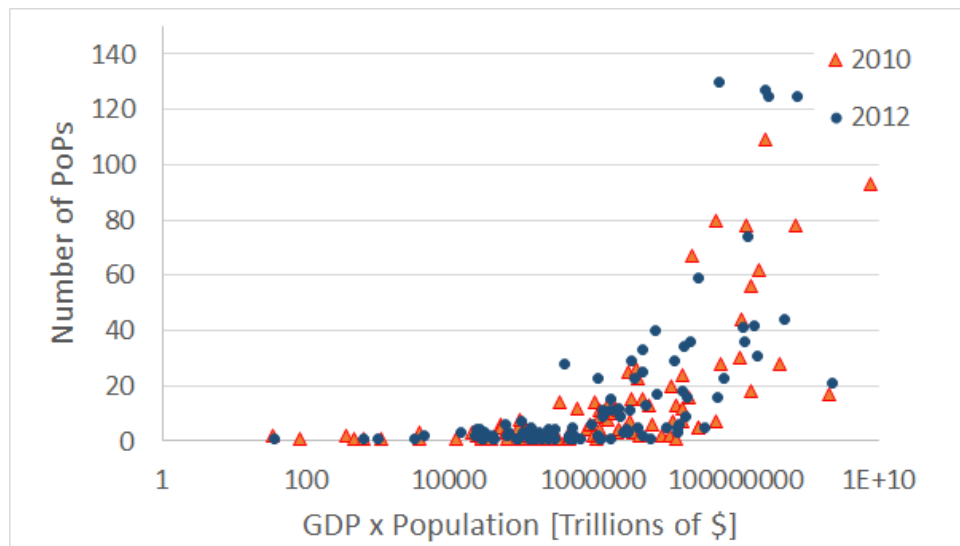


**Figure 7.5: Country Level GDP (PPP) per Capita vs. Number of PoPs**

not be found without a meaningful square root error. Both 2010 and 2012 datasets exhibit a similar pattern and are overlapping in many points.

One may expect that other types of predictors relating to GDP will provide a better indication for the number of PoPs. One such parameter is the GDP (PPP) per capita, meaning the gross domestic product at purchasing power parity per person, which is often considered an indicator of a country's standard of living. However, this turns out not to be a good indicator, as shown in Figure 7.5: some countries have very high GDP per capita but very few PoPs (e.g. Qatar, Kuwait) whereas countries such as China and Russia have considerably lower GDP per capita, but many more PoPs. The correlation coefficient in this case is only 0.25, both in 2010 and 2012.

Another possible predictor for the number of PoPs, complimentary to the previous one, is the multi-



**Figure 7.6: Country Level GDP x Country Level Population vs. Number of PoPs**

plication of GDP in the population, however this turns out to yield results that are slightly less aligned with the best fitted linear line than the dependence on GDP alone, as shown in Figure 7.6, but more aligned than PPP. The correlation coefficient in this case is 0.55 in 2010 and 0.5 in 2012. This indicator may explain why countries with high GDP and small population have the same number of PoPs as countries with a large population but a medium GDP: the GDP is not the only factor, so very large countries with a medium GDP will still need a significant PoPs infrastructure, to provide Internet services its residents.

The growth in GDP is not an indicator to the number of PoPs, and countries with high GDP growth do not have more PoPs than countries with low or negative GDP growth. The correlation coefficient here is neutral: ranging from zero to  $-0.05$ . Similarly, the growth in the number of PoPs between 2010 and 2012 is not correlated with the growth in GDP. An example to this is Japan, that had at the end of 2011 a GDP growth of  $-0.7\%$  whereas its number of PoPs grew by 60%.

### 7.3.3 *The Relation between PoPs and Internet Users*

At a first glance, the number of Internet users per country may seem a good indicator for the number of PoPs: one may expect that the need for PoPs will rise as more Internet users require Internet connectivity. This assumption, however, is not founded. When studying the relationship between the number of PoPs and the number of Internet users per country<sup>3</sup>, there is some weak relation in countries with many Internet users. Meaning, most countries with 20 PoPs or more have tens of

<sup>3</sup>This study is limited to the 2010 dataset only, as the WDI dataset included only the number of Internet users up till 2010, and for consistency we chose not to take this data from later datasets.

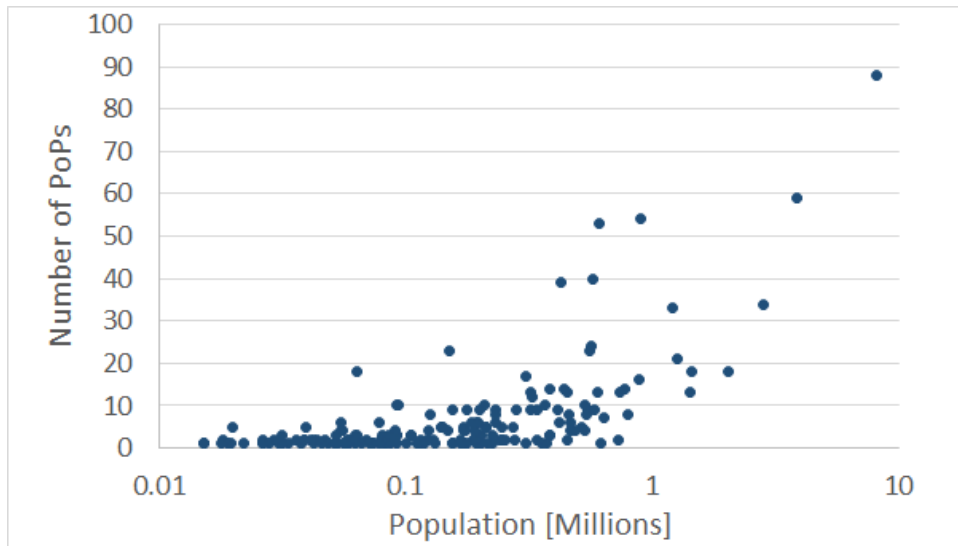
millions of Internet users. Yet this is not always the case: countries such as Austria, Sweden and Switzerland have six to eight million Internet users, but over 25 PoPs in each. Considering the other way around, i.e., whether a large number of Internet users calls for a large number of PoPs, there are some exceptions as well: Nigeria, Turkey and Pakistan all have twenty eight million Internet users or more, but five PoPs or less. The correlation coefficient between the number of Internet users and the number of PoPs is 0.53 in this case.

One explanation may stem from the percentage of Internet users in the population (information taken from the WDI dataset), but our analysis shows no connection between the percentage of Internet users and the number of PoPs. It is not only weaker that the total number of Internet users versus PoPs, it seems to be merely related, with a correlation coefficient of only 0.18. The same applies also for the average bandwidth per user, where the correlation coefficient is 0.19 (based on [14] and covering 49 countries).

A possible explanation to why the number of PoPs does not depend on the number of Internet users, is that service providers not necessarily have to increase the number of PoPs in order to handle increasing demand for Internet access. For example, they can expand existing PoPs, adding more networking equipment and thus exposing more ports towards the end users. The providers can also replace the technology used in their PoP, e.g., using 10GE interfaces instead of 1GE. Last, it is possible that in dense areas, such as crowded cities, we fail to detect multiple PoPs per a single ISP, due to the nature of our algorithm.

#### *7.3.4 A Study of the United States*

The United States is a special case amongst all countries. First of all, the number of PoPs detected in it is extremely high (1203 in 2012). Second, it is a vast country, with a high GDP, considerable population and it is technologically advanced. Last, the large amount of economic and demographic information which is available on city and metropolitan level, enables us to perform more accurate and advanced studies.

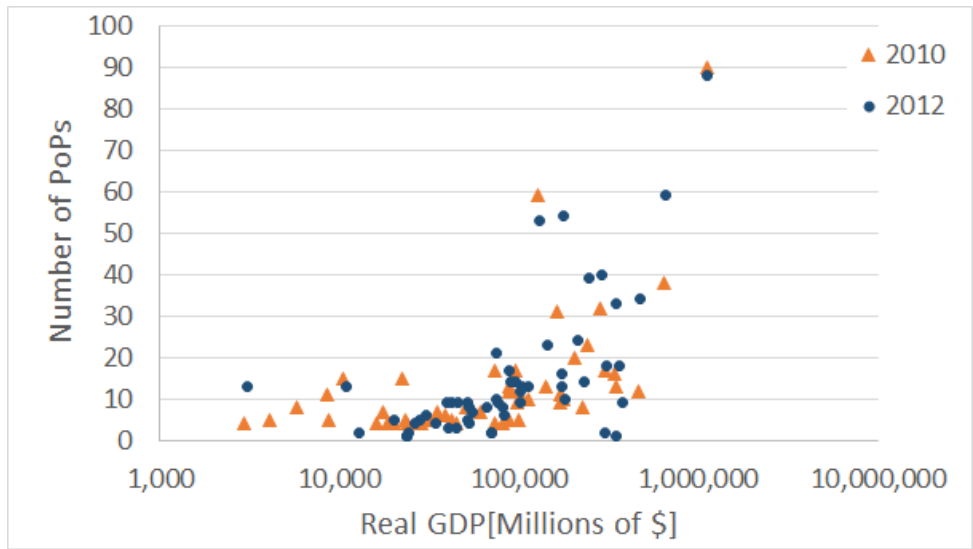


**Figure 7.7: United States City Level Population vs. Number of PoPs**

#### 7.3.4.1 *The Relation between PoPs and Population in the US*

As noted above, the US is one of three countries where no direct relation is observed between city level population and the number of PoPs in that city. Figure 7.7 demonstrates this, with the X-axis being the number of residents in a city (in Millions) and the Y-axis being the number of PoPs (aggregated by AS) in that city. The correlation coefficient, which is 0.79 and 0.78 in 2010 and 2012, correspondingly, does not tell the whole story: While for many cities, like New York and Los Angeles, the rule that more residents mean more PoPs applies, there are many exceptions. Amongst the medium-size cities (less than a million people) one can find cities like Boston or Baltimore with 500K to 600K residents but over 40 PoPs. Many PoPs are sometimes found in small cities as well: 23 PoPs in Springfield, MO (150K residents) or 10 PoPs in Albany, NY (94K residents). Consequently, additional indicators need to be found for the number of PoPs in a city.

The study of additional indicators uses metropolitan level statistics, rather than city level, as this level of aggregation has most information from official US government sources, such as the Bureau of Economic Analysis (BEA), Bureau of Transportation Statistics (BTS) and above all the US Census Bureau. On the PoP level the usage of metropolitans rarely affects the results due to range of convergence applied when assigning PoPs to cities. In a handful of cases where a metropolitan area includes more than a single city, such as Dallas-Port Worth, we aggregate the PoPs' city level information to the metropolitan level.

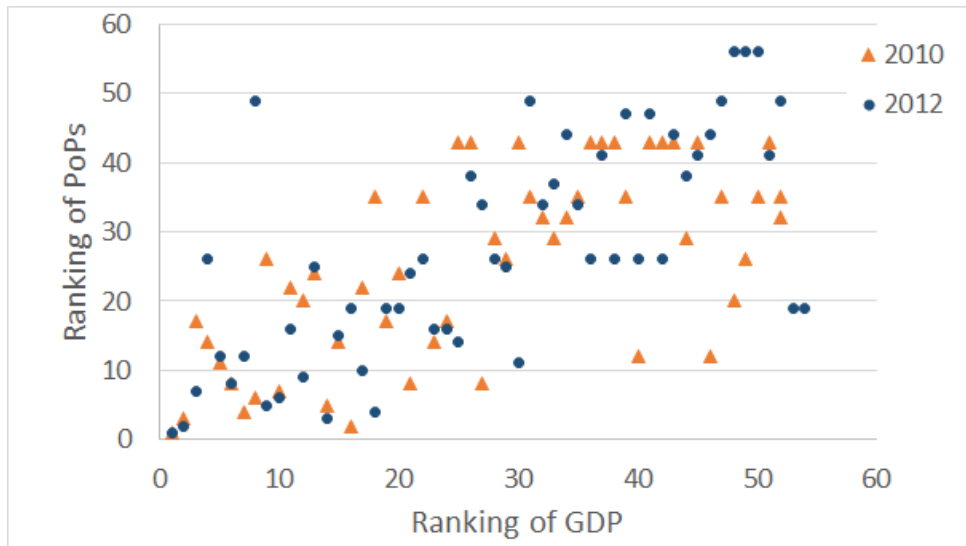


**Figure 7.8: United States Metropolitan Level GDP vs. Number of PoPs**

#### 7.3.4.2 The Relation between PoPs and GDP in the US

Another economical aspect that was studied on country level and can now be observed on metropolitan level is GDP. We study the real GDP (in millions of chained 2005 dollars) as provided by the BEA [119], as a total of all industries. The analysis covers the 50 largest metropolitans (by population) in which we detected PoPs in 2010 and 2012. As opposed to what one may expect, the correlation between GDP and number of PoPs is weaker on metropolitan level: only 0.78 in 2010 and 0.76 in 2012. While the correlation is still evident, it is not as strong as on the country level. This is demonstrated in Figure 7.8. The GDP is shown in millions of dollars on the x-axis, whereas the number of PoPs is shown on the y-axis. While in most metropolitan areas the change in GDP between the years is small, while the number of PoPs rises, there are a few metropolitans where the number of PoPs decreases. Although this may be attributed to lack of measurements, this is also the result of acquisition or merging of some ISPs, which cause a convergence of PoPs in a given area, as we count the PoPs of every AS only once per city.

Another way to consider the relation between GDP and PoPs is using ranking: We rank the metropolitans by the number of PoPs in them, with the highest rank going to the metropolitan with most PoPs and the lowest rank to the one with least PoPs. If two metropolitans have the same number of PoPs, their ranking is similar. Identical ranking is applied to each metropolitan's GDP. This method is selected as much of the US metropolitan area statistics is published using ranking. Figure 7.9 demonstrates the relation between the GDP ranking and the ranking of PoPs: generally speaking, the higher



**Figure 7.9: United States Metropolitan Level Ranking of GDP vs. Ranking of PoPs**

the GDP of a metropolitan, the higher its PoPs' ranking. However, this is not a clear linear relation and there are some exceptions, e.g. San Francisco, ranked 8th by GDP but only 48th by PoPs (in 2012). The correlation coefficient in this case is another evidence: In 2010 it is 0.65 and in 2012 it is 0.71. Both coefficients are weaker than the correlation coefficients between the Real GDP and the number of PoPs.

Another economic factor that is considered is personal income: we study the per capita personal income in the same metropolitan areas, as published by the BEA [118]. The correlation between income and the number of PoPs is weaker than GDP, yet stronger than the country level PPP; It reaches 0.5 in 2012 and 0.63 in 2010. The large gap between the two datasets is another reason not to consider this parameter as a good indicator.

#### 7.3.4.3 *The Relation between PoPs and Transportation in the US*

It is a common assumption that networking infrastructure is tightly related with transportation infrastructure [129], such as railways and highways, and that main transportation hubs also serve as communication hubs. We examine this assumption when considering PoPs and various transportation related statistics in the United States.

The first aspect under study is the US' top freight gateways, in sea, air and land [122]. As these gateways require significant infrastructure in order to transport the cargo, it is interesting to check whether the same locations also serve as networks' landing points and as centers of PoPs. The dataset

is compared to 2012 PoPs dataset. It is found that only twenty of the metropolitans under study are included within the Top 50 freight gateways (the size of the dataset). In this count we include also gateways that are in proximity (up to 150km) of the metropolitan. Four of the metropolitans have more than a single type of a gateway in the list: Houston, Los Angeles, Miami and New Orleans. Ten of these gateways are through water, twelve through air and only three through land. The calculated correlation coefficient between the total value of shipments through a gateway and the number of PoPs is 0.58, and only 0.34 when the ranking of a gateway and the ranking of PoPs is considered. This calculation uses only the small set of twenty metropolitans, and is thus very sensitive and prone to fluctuations.

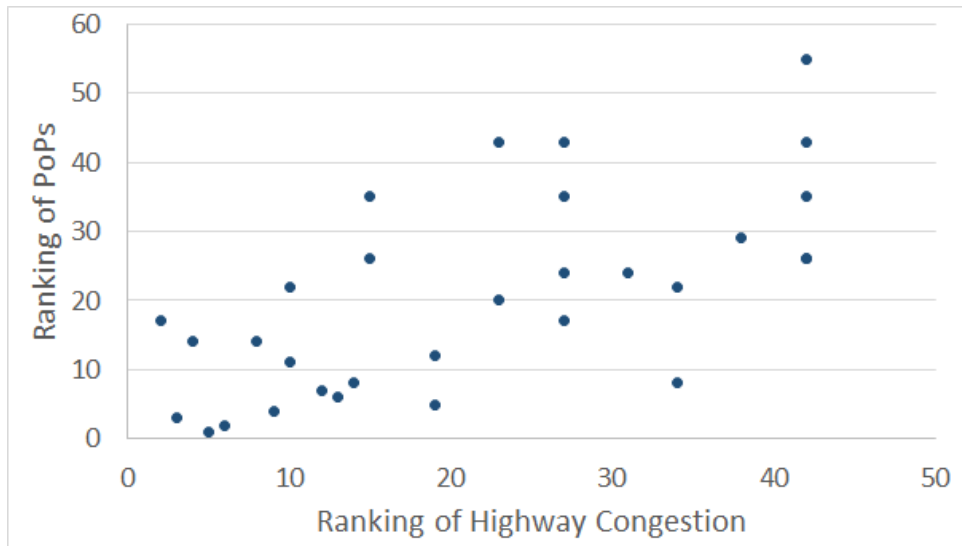
Following freight gateways, we focus on passengers transportation through airports. A database of top 50 US airports is used for this end [121] and is compared with the 2012 PoPs dataset. Out of the top 50 airports, 37 are included in the top 50 metropolitans with PoPs, and a total of 41 in the list of all cities with PoPs. While this indicates that metropolitans with considerable air traffic are likely to have a lot of active ISPs, the relation between the number of PoPs and the amount of passenger is weak: a correlation coefficient of 0.3. The relation between the ranking of a metropolitan by PoPs and by airport's passenger is even weaker, 0.11.

Another type of transportation infrastructure is denoted by railways. While not a direct indicator of railway tracks infrastructure, we examine the top 50 Amtrak stations by number of passengers and compare it to the 2012 PoPs dataset. Just 15 metropolitans are shared between the list of top metropolitans with PoPs and top Amtrak stations. Six more metropolitans appear in the full list of metropolitans with PoPs. However, one needs to note that the Amtrak ranking list includes some cities more than once, thus there are only 43 distinct metropolitans in it. Fifteen overlapping metropolitans can be considered insufficient to calculate the correlation coefficient, but for illustration purposes, we find it to be 0.44 for the relation between number of passenger and number of PoPs and 0.37 between a station's ranking and the PoPs' ranking.

The last case relating to transportation under study is highway congestion in the 50 largest urban areas [120]. This set matches 35 metropolitans in the 2010 dataset<sup>4</sup>. Surprisingly, we find here better correlation to the number and ranking of PoPs compared to the previous cases: the correlation coefficient between the total hours of delay and the number of PoPs is 0.61 and the correlation coefficient

---

<sup>4</sup>We use the latest BTS dataset available.



**Figure 7.10: United States Metropolitan Level Ranking of Highway Congestion vs. Ranking of PoPs**

between the ranking of a metropolitan by a highway congestion delay per commuter and its ranking by PoPs is 0.66. While this is not a strong correlation as with GDP, it is better than for other types of transportation indicators. The correlation between the highway congestion delay hours per commuter and the number of PoPs is weaker, being 0.42. The relationship between a metropolitan ranking by highway congestion per commuter versus ranking by PoPs is shown in Figure 7.10. Note that both ranking lists have several metropolitan with the same ranking, due to identical number of PoPs or identical hours of delay per commuter, which affects the correlation and is reflected in the graph.

#### 7.3.4.4 The Relation between PoPs and Demographic Factors in the US

The United States census information is used to study the relation between two demographic aspects and PoPs: age and race. These two aspects are selected due to their availability, compared to other important aspects that are either not covered on metropolitan level or that were already covered before in this work, such as income.

The 2010 US census information is used in conjunction with the 2010 PoPs dataset to study the relation between different age groups in a metropolitan [113] and the number of PoPs in the area. Table 7.3 shows the correlation coefficient between each age group and the number of PoPs. As can be seen, the correlation coefficient is very similar for all age groups, which may indicate a relation to all age groups. However, as there is a correlation of over 0.99 between the overall size of the population and the size of a specific age group, the results actually reflect the relation between PoPs



Age Group	All	Under 18	18-44	45-64	Over 65
Correlation Coefficient	0.76	0.74	0.76	0.77	0.77

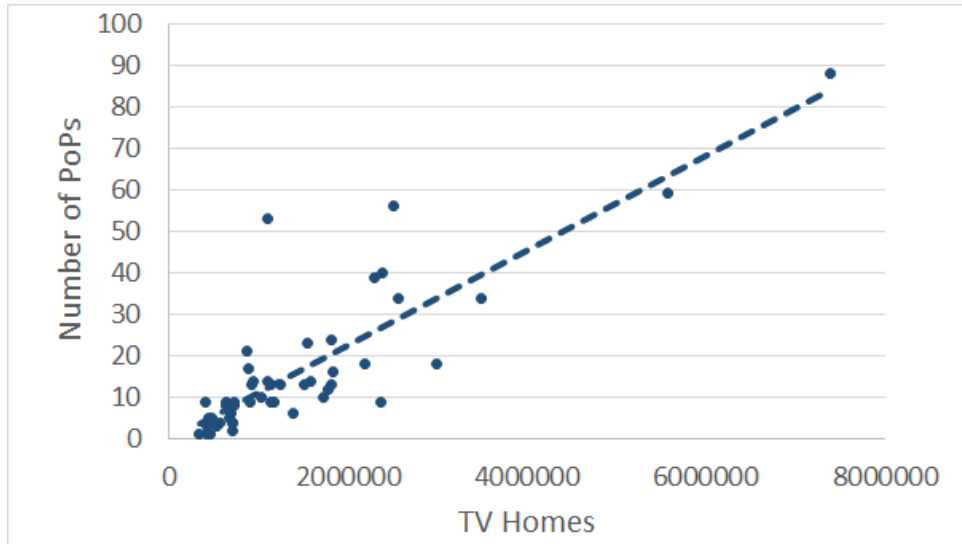
**Table 7.3: Correlation Coefficient Between Size of Different Age Groups and The Number of PoPs**

and population as studied in the beginning of this section.

To study the relation between race and the number of PoPs, we use the information gathered on the same 2010 US census with the 2010 PoPs dataset. The census dataset [114] states for each metropolitan statistical area the number of people by race. Race may be White alone, Black or African American alone, American Indian or Alaska Native alone, or Asian alone. We do not refer to Native Hawaiian and Other Pacific Islander alone as the size of their population is negligible in most metropolitans (a few hundreds of people). In addition, a person may define himself as from two or more races. An exception is people from Latino or Hispanic origin, who may be of any race, and are therefore counted separately (i.e. both under their race and origin). The results portray a complicated story: The correlation coefficient between the size of the population of white, asian or people of two or more races is almost identical to the correlation coefficient for the entire population. For white people, who are about 78% of the population this is understandable, as the size of their population has 0.99 correlation coefficient to the entire population. Asian and people of two or more race, each pose about 2.7% of the population and have 0.91 and 0.96 correlation coefficient to the overall population, which is weaker than for white people but still very high. The size of the American Indian population has a correlation coefficient of only 0.51 to the number of PoPs, yet they are only 1% (on the average) of the overall population with 0.74 correlation coefficient to it, so their case might not be well represented. The African American and Hispanic population are a different case: their share of the population is rather large (each over 10%) and while their correlation coefficient to the overall population is lower than White people or people of two races or more, it is almost the same as that of the Asian population. Yet, the correlation coefficient of these two group is lower by 12%–18% compared to other major races. This may still be attributed to the lower correlation between the size of this population and the overall population, but it may also be driven by other social and demographic factors.

Race	All	White	African American	American Indian	Asian	Two Races	Hispanic
Correlation Coeff to PoPs	0.76	0.76	0.65	0.51	0.76	0.74	0.62
Correlation Coeff to All		0.99	0.87	0.74	0.91	0.96	0.89
Average % of All		78.1%	10.7%	1.0%	2.7%	2.7%	12.4%

**Table 7.4: Correlation Coefficient Between Size of Different Race Groups and The Number of PoPs**



**Figure 7.11: United States Number of TV Homes vs. Number of PoPs**

#### 7.3.4.5 The Relation between PoPs and TV Market in the US

Nielsen Media Research releases every year a rating of Designated Market Areas (DMA) across the US. The size of a market is measured by the size of the television audience in it, where the audience do not need to live within a city to be considered part of its DMA, rather live where its stations are watched the most. For example, the Philadelphia DMA includes southern New Jersey and most of Delaware. We take Nielsen’s 2011-2012 ranking and compare it to the 2012 PoPs dataset.

As the Nielsen dataset includes not only the ranking of the markets, but also their size by the number of TV homes, we first check the correlation between the size of a TV market and the population of the given DMA. A very high correlation may cause the results to mirror the correlation between PoPs and population and thus make the size of the TV market a redundant indicator. The resulting correlation coefficient is 0.88 (compared with the population dataset used in Section 7.3.4.1), which is high but does not mean that the relationship is identical. The correlation coefficient between the number of TV homes and the number of PoPs in a city is found to be 0.85, whereas the correlation

coefficient between the ranking of a TV market and the ranking of its PoPs is 0.82. Both coefficients are higher than 0.78, which is the correlation coefficient for the relation between population and PoPs. Figure 7.11 shows the relationship between the number of TV homes and the number of PoPs, with the dashed line showing the linear relation between the two. The line's coefficient of determination is 0.71. One can expect that the high correlation between the size of the TV market and the number of PoPs will be a result of IPTV penetration, which requires such network infrastructure, however in 2012 IPTV had only 9.66 million subscribers in the United States [85]. On the other hand, the penetration of IPTV to broadband users in the country was almost 40% in 2012 [110]. It thus seems that the right course is to study further the relation between broadband penetration and the number of PoPs. Unfortunately, we did not manage to locate this information on the city or metropolitan level.

#### *7.3.4.6 The Relation between PoPs and Sports Teams in the US*

To continue the line presented in the previous subsection, one possible driver of TV markets is sports events, such as NFL games, that often lead the TV shows ratings lists. It was thus suggested that cities hosting such events require considerable infrastructure in order to support the media, and therefore it may relate to the number of PoPs.

The major sports leagues in the United States and Canada are the MLB (baseball), NBA (basketball), NFL (football) and NHL (hockey). These four leagues are often called "The Big Four". Adding also the MLS (soccer) and CFL (Canadian football) is referred to as "The Big Six". The Big Six sports teams are located in forty metropolitan area in the US, and 9 more in Canada. We focus on the US sports teams and use the current allotment of teams to a metropolitan area. Information is collected from the official websites of the leagues. We note that since there are no CFL teams in the United States, only MLS teams make the difference between the Big Four and Big Six teams' count.

Out of the forty metropolitan areas where teams are located, we detect PoPs in thirty seven places. The average number of PoPs in each of these metropolitans is 21.1, with the median being 14, and the minimum number of PoPs being 4 in Oklahoma (which has only one sports team, in the NBA). The correlation coefficient between the number of PoPs and the number of Big Four sports team is 0.84 and the correlation to the number of Big Six teams is 0.86. This is a high level of correlation, especially as the correlation between the number of sports teams and the size of the population is no more than 0.7. To support this result, the average number of PoPs detected across the entire dataset

is 6.5 PoPs per metropolitan area, with the median being 2.5 PoPs and the minimum a single PoP. Considering the group of metropolitans with no sports teams, the average number of PoPs is 2.9 and the median is two. The large gap in the number of PoPs between the group of metropolitans with and without sport teams point that this may be a valid indicator.

While the results above suggest studying the relation between PoPs and National Collegiate Athletic Association (NCAA) teams, such a case will be complex: there are hundreds of NCAA teams and the games are on multiple broadcast networks as well as on local TV networks. This indicates that the relation will involve not only the PoPs and number of local teams, but also other factors such as the team's division and ranking, and possibly its market size.

#### **7.4 Discussion**

The analysis of the PoP level map versus the various economic and demographic aspects teaches us a few lessons. The most important lesson is that global analysis is too coarse to lead to a model or a set of indicators that will apply to all countries. The difference between countries is too large to expect that if a rule applies to the a country in North America or Europe it will also apply to Africa and Asia. The differences stem not only from the country's level of development or economic status, but also from government policy - for better (South Korea) or worse (Middle Eastern countries [100]). An evolution model that will try and predict PoPs evolution over time will therefore need to apply different metrics to different types of countries. These results corroborate previous works, such as Lakhina *et al.* [69], which showed that the number of router interfaces can't be correlated on a worldwide level, but that there is a correlation within economically homogeneous regions.

The analysis of the PoPs on US metropolitan level is in many ways more fruitful than on the country level. While this is largely due to the availability of information on metropolitan level, the vast size of the country and the large amount of PoPs detected in it, this is also due to derivative cultural aspects. The combination of the advanced technological status of the US with leisure culture make the effect of aspects like size of TV market and sports teams larger than in other places. If one would like to compare these aspects to other comparable areas, e.g., the European Union, he may find that it is hard. For example, in the TV market each country in the EU may have its own policy and use a local language, which is different than the US. Sports teams are also managed differently (e.g., the Football's Champions league and the basketball's Euroleague), as the sports team are included and

excluded vary each year based on local achievements and thus do not form a constant set of teams that requires long-term investment in communication infrastructure (except for a handful of leading clubs).

The indicator that turns out to be the most influential on the number of PoPs is the GDP: strongly on the country level and considerably on US metropolitan level. While analyzing the reason for that compared to other economic factors is outside our field of expertise, there is no doubt in the implications on the number of detected PoPs. On the US city level, the size of the population is a strong indicator to the number of PoPs, and it also has a correlation to other aspects with high level of correlation to the number of PoPs, such as age groups and TV market.

An unexpected result, from our point of view, was the low level of correlation between different aspects of transportation and the number of PoPs on the US metropolitan level. As network infrastructure is considered related to transportation infrastructure (e.g. [90] and [123]), one would expect higher correlation between the two. However, since the parameters that we study were limited due to datasets availability, they may not necessarily reflect the entire scope of transportation infrastructure in an area, which may be the cause for the results.

The lack of information for an evolution study of the PoPs level graph has two contributors. First, the economic, geographic and demographic information that is not always accessible and or not available on the required points in time, as discussed in Section 7.2.1. The second part is the short history of the Internet and the radical changes the network has gone through in the recent decade. As information from other fields of study has a long history and is commonly sampled on a decade and half decade basis, there are not enough overlapping sampling points to devise a reliable model based on measured data. The datasets used in the analysis are only a year and a half apart and thus are not far enough apart to indicate growth or change trends over time. While we do detect more PoPs in 2012 than in 2010, it will be incorrect to deduce anything based on these differences. The use of the two datasets does support the results regarding the correlation between different indicators and the number of PoPs, as the correlation coefficients are very similar across the years.

## VIII Conclusion

This thesis work presented a novel structural approach to automatically generate world-wide PoP maps using the DIMES project infrastructure. The extraction algorithm is based on detection of a network motif, and we discuss at length the theoretical background supporting this scheme. The generated PoP maps have location information for each PoP, deduced from geolocation databases and using a geolocation algorithm which increases the PoP location accuracy. An extensive validation of both PoPs extraction and geolocation algorithms is provided, studying different aspects of the approach.

The thesis also provides a comprehensive study of geolocation databases, comparing a large number of databases of different types. We showed that the information in the databases may be largely biased at the ISP level. Additionally, correlation was found between some databases, while minimal correlation is found between others. The differences between databases is sometimes even in the range of countries.

To mitigate the effect of geolocation services inaccuracies, we presented a new algorithm for PoPs' geolocation that uses PoPs with high level of confidence to geographically locate PoPs with lower confidence using PoP-level link delays. Details of the algorithm's performance, sensitivity to parameters and possible limitations were also provided.

One of the Internet PoP-level maps applications discussed in this work is a method to infer AS relationships using PoP data. The method is useful to detect complex types of relationships as well as anomalies and mistakes in existing ToR datasets. The method leverages PoP-level maps, which reduces the size of the analyzed datasets and highlights anomalies that are otherwise hard to detect on the IP or the router level. Future work on this topic will extend this study, further examining and validating complex AS relationships and anomalies. Additional future work will focus on geography related aspects of ToR, and how they affect the robustness of the network.

In the last part of the thesis, we set the foundations for PoP-level Internet evolution models. We examined different aspects of geographic, economic and demographic factors and checked their correlation

to the number of PoPs in different countries and cities around the world. The results show that GDP is a good indicator of the number of PoPs, on the country level, and that on the US metropolitan level the population may be a good indicator, as well as several other related aspects.

A future evolution model will need to take into account multiple PoP-level maps spread across a longer span of time, in order to achieve a better understanding of the dynamics of evolution over time. The model should look on the city level, where possible, and should distinguish between countries based on their different characteristics (e.g. economic region). Further work should also involve researchers from other disciplines, such as geographic and economic studies, in order to better analyze the data and have a better understanding of the results. Another direction is to perform a focused study of other countries than the United States, and to check similarities and differences.

Another aspect that we hope to study in the future is the evolution of the PoP level map's technological infrastructure. This means that one should look not only at the number of PoPs but also at the technology that is used in them, e.g., 10GbE, 40GbE or 100GbE, and the number of exposed interfaces in each PoP. This kind of study will require collaboration with service providers, as the type of infrastructure used is rarely revealed. This type of study may reflect better some changes in the evolution of the network, due to the dominance of some tier-1 ISPs, who may have greater influence on the network than the introduction of new ones to a city's PoP level map.

## References

- [1] Mfinder - network motifs detection tools. <http://www.weizmann.ac.il/mcb/UriAlon/>.
- [2] Sarangworld project. <http://www.sarangworld.com/TRACEROUTE/>.
- [3] IPInfoDB. <http://ipinfodb.com>, 2010.
- [4] Quova. <http://www.quova.com>, 2010.
- [5] Neustar. <http://www.neustar.biz/>, 2012.
- [6] The CAIDA AS Relationships Dataset. <http://www.caida.org/data/active/as-relationships/>, 2012.
- [7] United States Census Bureau. <http://www.census.gov/>, 2013.
- [8] Akamai. EdgePlatform. <http://www.akamai.com/>, 2010.
- [9] D. G. Andersen, N. Feamster, S. Bauer, and H. Balakrishnan. Topology inference from BGP routing dynamics. In *Internet Measurement Workshop*, pages 243–248, 2002.
- [10] AT&T Global Services. AT&T Global Services global network map. [http://www.corp.att.com/globalnetworking/media/network\\_map.swf](http://www.corp.att.com/globalnetworking/media/network_map.swf).
- [11] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira. Avoiding traceroute anomalies with paris traceroute. In *IMC'06*, pages 153–158, 2006.
- [12] W. Bank. *World Development Indicators 2012*. The World Bank, 2012.
- [13] G. D. Battista, M. Patrignani, and M. Pizzonia. Computing the types of the relationships between autonomous systems. Technical Report RT-DIA-73-2002, Dipartimento di Informatica e Automazione, Universita di Roma Tre, 2002., 2002.



- [14] D. Belson. The State of the Internet, 2nd Quarter, 2012 Report. *Akamai*, 5(2), 2012.
- [15] A. Bender, R. Sherwood, and N. Spring. Fixing ally's growing pains with velocity modeling. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement (IMC'08)*, pages 337–342, 2008.
- [16] BT Global Services. Network maps. <http://www.bt.net/info/europe.shtml>.
- [17] CenturyLink Business. CenturyLink network maps. <http://www.centurylink-business.com/demos/network-maps.html> [Accessed: October 10,2012].
- [18] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman. Planetlab: an overlay testbed for broad-coverage services. *SIGCOMM Comput. Commun. Rev.*, 33(3):3–12, 2003.
- [19] N. Czernich, O. Falck, T. Kretschmer, and L. Woessmann. Broadband infrastructure and economic growth\*. *The Economic Journal*, 121(552):505–532, 2011.
- [20] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: a decentralized network coordinate system. In *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '04, pages 15–26, 2004.
- [21] DB-IP. IP-City Database. <http://db-ip.com/>, 2012.
- [22] A. Dhamdhere and C. Dovrolis. Twelve Years in the Evolution of the Internet Ecosystem. *IEEE/ACM Transactions on Networking*, 19(5):1420–1433, sep 2011.
- [23] A. Dhamdhere and C. Dovrolis. Twelve years in the evolution of the internet ecosystem. *IEEE/ACM Trans. Netw.*, 19(5), Oct. 2011.
- [24] Digital Envoy. NetAcuity Edge. [http : //www.digital - element.com/our\\_technology/edge.html](http://www.digital-element.com/our_technology/edge.html), 2010.
- [25] DIMES. Distributed Internet Measurements and Simulations. <http://www.netdimes.org/>.

- [26] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, kc claffy, and G. Riley. AS relationships: Inference and validation. 37, 2006.
- [27] B. Donnet. Internet topology discovery. In E. Biersack, C. Callegari, and M. Matijasevic, editors, *Data Traffic Monitoring and Analysis*, volume 7754 of *Lecture Notes in Computer Science*, pages 44–81. 2013.
- [28] B. Donnet and T. Friedman. Internet topology discovery: a survey. *Communications Surveys Tutorials, IEEE*, 9(4):56–69, 2007.
- [29] J. C. Doyle, D. L. Alderson, L. Li, S. Low, M. Roughan, S. Shalunov, R. Tanaka, and W. Willinger. The "robust yet fragile" nature of the internet. *Proceedings of the National Academy of Sciences of the United States of America*, 102(41):14497–14502, 2005.
- [30] B. Eriksson, P. Barford, J. Sommers, and R. Nowak. A learning-based approach for IP geolocation. In *Proceedings of the 11th international conference on Passive and Active Measurement (PAM'10)*, pages 171–180, 2010.
- [31] T. Erlebach, A. Hall, and T. Schank. Classifying customer-provider relationships in the internet. In *CCN*, 2002.
- [32] D. Feldman and Y. Shavitt. An optimal median calculation algorithm for estimating Internet link delays from active measurements. In *IEEE Workshop on End-to-End Monitoring Techniques and Services (E2EMON)*, 2007.
- [33] D. Feldman and Y. Shavitt. Automatic large scale generation of Internet PoP level maps. In *Proceedings of the Global Telecommunications Conference (GLOBECOM'08)*, pages 2426–2431, 2008.
- [34] D. Feldman, Y. Shavitt, and N. Zilberman. A structural approach for PoP geolocation. *Computer Networks*, 56(3):1029–1040, 2012.
- [35] L. Gao. On inferring autonomous system relationships in the internet. 9(6), 2001.
- [36] GARR, The Italian academic and research network. <http://www.garr.it/eng/index.php>.

- [37] Geobytes. GeoNetMap. *http://www.geobytes.com/*, 2010.
- [38] F. Georgatos, F. Gruber, D. Karrenberg, M. Santcross, H. Uijterwaal, and R. Wilhelm. Providing active measurements as a regular service for ISPs. In *Proceedings of the Passive and Active Measurements Workshop (PAM'01)*, 2001.
- [39] Global Crossing. Global Crossing network.  
*http://www.globalcrossing.com/html/map062408.html*.
- [40] Gomez Inc. Akamai ip location service performance assessment, akam01, 2009.
- [41] G. Goodell and P. Syverson. The right place at the right time. *Communications of the ACM*, 50(5):113–117, 2007.
- [42] Google. Geolocation API. *http://code.google.com/apis/gears/api\_geolocation.html*, 2010.
- [43] Google Gears Wiki. Geolocation API.  
*http://code.google.com/p/gears/wiki/GeolocationAPI*, 2010.
- [44] R. Govindan and H. Tangmunarunkit. Heuristics for internet map discovery. In *Proceedings of IEEE INFOCOM*, Tel Aviv, Israel, 2000.
- [45] B. R. Greene and P. Smith. *Cisco ISP Essentials*. Cisco Press, 2002.
- [46] E. Gregori, A. Improta, L. Lenzini, and C. Orsini. The impact of ixps on the as-level topology structure of the internet. *Computer Communications*, 34(1):68 – 82, 2011.
- [47] B. Gueye, S. Uhlig, and S. Fdida. Investigating the imprecision of IP block-based geolocation. In *Proceedings of the 8th international conference on Passive and Active Measurement (PAM'07)*, pages 237–240, 2007.
- [48] B. Gueye, S. Uhlig, A. Ziviani, and S. Fdida. Leveraging buffering delay estimation for geolocation of internet hosts. In *Proceedings of the 5th international IFIP-TC6 conference on Networking Technologies, Services, and Protocols (NETWORKING'06)*, pages 319–330, 2006.

- [49] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida. Constraint-based geolocation of Internet hosts. *IEEE/ACM Transactions on Networking*, 14(6), 2006.
- [50] M. A. Hameed, A. Jabbar, E. K. Çetinkaya, and J. P. Sterbenz. Deriving Network Topologies from Real World Constraints. In *IEEE GLOBECOM Workshop on Complex and Communication Networks (CCNet)*, pages 415–419, December 2010.
- [51] Hexsoft Development. IP2Location. <http://www.ip2location.com>, 2010.
- [52] Homeland Security. A roadmap for cyber security research. <http://www.cyber.st.dhs.gov/docs/DHS-Cybersecurity-Roadmap.pdf>, 2009.
- [53] hostip.info. hostip.info. <http://www.hostip.info>, 2010.
- [54] B. Huffaker, A. Dhamdhere, M. Fomenkov, and kc claffy. Toward topology dualism: Improving the accuracy of as annotations for routers. In *Proceedings of the 11th international conference on Passive and Active Measurement (PAM'10)*, pages 101–110, 2010.
- [55] B. Huffaker, D. Plummer, D. Moore, and k. Claffy. Topology discovery by active probing. In *Symposium on Applications and the Internet (SAINT)*, pages 90–96. SAINT, 2002.
- [56] S. M. Huffman and M. H. Reifer. Method for geolocating logical network addresses, 2005.
- [57] G. Huston. Interconnection, peering, and settlements. In *INET*, San Jose, CA, USA, June 1999.
- [58] Y. Hyun. Archipelago measurement infrastructure. <http://www.caida.org/projects/ark/>.
- [59] Y. Hyun, A. Broido, and k. Claffy. On Third-party Addresses in Traceroute Paths. In *PAM'03*, San Diego, CA, 2003.
- [60] IETF. Geopriv workgroup. <http://tools.ietf.org/wg/geopriv/>.
- [61] Internet2. Internet2 gigapop list. <http://eng.internet2.edu/gigapoplist.html> [Accessed: October 10,2012].
- [62] IPLigence. IPLigence Max. <http://www.ipligence.com>, 2010.

- [63] E. Katz-Bassett, J. P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards IP geolocation using delay and topology measurements. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement (IMC'06)*, pages 71–84, 2006.
- [64] K. Keys, Y. Hyun, M. Luckie, and k. Claffy. Internet-Scale IPv4 Alias Resolution with MIDAR. *IEEE/ACM Transactions on Networking*, 2012.
- [65] K. F. King. Geolocation and federalism on the Internet: Cutting Internet gambling's gordian knot. *Columbia Science and Technology Law Review*, Forthcoming, 2009.
- [66] S. Knight, H. Nguyen, N. Falkner, R. Bowden, and M. Roughan. The internet topology zoo. *Selected Areas in Communications, IEEE Journal on*, 29(9):1765–1775, 2011.
- [67] J. Kolko. Broadband and local growth. *Journal of Urban Economics*, 71(1):100–113, 2012.
- [68] P. Koutroumpis. The economic impact of broadband on growth: A simultaneous approach. *Telecommunications Policy*, 33(9):471–485, 2009.
- [69] A. Lakhina, J. W. Byers, M. Crovella, and I. Matta. On the geographic location of Internet resources. *Selected Areas in Communications, IEEE Journal on*, 21(6):934–948, Aug. 2003.
- [70] S. Laki, P. Mátray, P. Haga, T. Sebok, I. Csabai, and G. Vattay. Spotter: A model based active geolocation service. In *Proceedings of IEEE INFOCOM*, Shanghai, China, 2011.
- [71] D. Lee, K. Jang, C. Lee, G. Iannaccone, and S. Moon. Scalable and systematic internet-wide path and delay estimation from existing measurements. *Computer Networks*, 55(3), 2011.
- [72] H. V. Madhyastha. An information plane for internet applications. Thesis, University of Washington, 2008.
- [73] H. V. Madhyastha, T. Anderson, A. Krishnamurthy, N. Spring, and A. Venkataramani. A structural approach to latency prediction. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement (IMC'06)*, pages 99–104, 2006.
- [74] P. Mátray, P. Haga, S. Laki, G. Vattay, and I. Csabai. On the spatial properties of internet routes. *Comput. Netw.*, 56(9):2237–2248, June 2012.

- [75] MaxMind LLC. GeoIP City Accuracy for Selected Countries. *http : //www.maxmind.com/app/city\_accuracy*, 2008.
- [76] MaxMind LLC. GeoIP. *http://www.maxmind.com*, 2010.
- [77] MaxMind LLC. Free World Cities Database. *http : //www.maxmind.com/en/worldcities*, 2012.
- [78] A. E. McElligott. Method and software product for identifying network devices having a common geographical locale, 2007.
- [79] Merit Network. Internet routing registries. *http://www.irr.net/*.
- [80] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.
- [81] J. A. Muir and P. C. V. Oorschot. Internet geolocation: Evasion and counterevasion. *ACM Computing Surveys*, 42(1):1–23, 2009.
- [82] V. N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for Internet hosts. In *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '01)*, pages 173–185, 2001.
- [83] R. Pastor-Satorras and A. Vespignani. *Evolution and Structure of the Internet: A Statistical Physics Approach*. Cambridge University Press, 2004.
- [84] I. Poesse, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye. IP geolocation databases: unreliable? *ACM SIGCOMM Computer Communication Review*, 41(2), 2011.
- [85] Point Topic LTD. IPTV Statistics - Market Analysis. *http : //point – topic.com/wp – content/uploads/2013/02/Sample – Report – Global – IPTV – Statistics – Q2 – 2012.pdf*, 2012.
- [86] Pricewaterhouse Coopers LLP. Quova - report of independent accountants. *http : //www.quova.com/documents/PricewaterhouseCoopers\_Audit.pdf*, 2008.

- [87] QWest Business. Network maps. <http://shop.centurylink.com/largebusiness/enterprisesolutions/networkMaps/> [Accessed: October 10,2012].
- [88] P. Radoslavov, H. Tangmunarunkit, H. Yu, R. Govindan, S. Shenker, and D. Estrin. On characterizing network topologies and analyzing their impact on protocol design. Technical report, Computer Science Department, University of Southern California, 2000.
- [89] A. H. Rasti, N. Magharei, R. Rejaie, and W. Willinger. Eyeball ASes: from geography to connectivity. In *Proceedings of the 10th annual conference on Internet measurement (IMC'10)*, pages 192–198, 2010.
- [90] S. Rinaldi, J. Peerenboom, and T. Kelly. Identifying, understanding, and analyzing critical infrastructure interdependencies. *Control Systems, IEEE*, 21(6):11–25, 2001.
- [91] L.-H. Roller and L. Waverman. Telecommunications infrastructure and economic development: A simultaneous approach. *American Economic Review*, 91(4):909–923, 2001.
- [92] S. Ruggles, J. T. Alexander, K. Genadek, R. Goeken, M. B. Schroeder, and M. Sobek. *Integrated Public Use Microdata Series: Version 5.0 [Machine-readable database]*. Minneapolis: University of Minnesota, 2010.
- [93] A. Sardella. Building next-gen points of presence, cost-effective PoP consolidation with juniper routers. White paper, Juniper Networks, 2006.
- [94] C. M. Schneider, A. A. Moreira, J. S. Andrade, S. Havlin, and H. J. Herrmann. Mitigation of malicious attacks on networks. *Proceedings of the National Academy of Sciences*, 108(10):3838–3841, 2011.
- [95] Y. Schwartz, Y. Shavitt, and U. Weinsberg. A measurement study of the origins of end-to-end delay variations. In *Proceedings of the 11th international conference on Passive and Active Measurement (PAM'10)*, pages 21–30, 2010.
- [96] S. Shakkottai, M. Fomenkov, R. Koga, D. Krioukov, and K. Claffy. Evolution of the internet as-level ecosystem. In *Complex Sciences*, volume 5 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 1605–1616. 2009.

- [97] Y. Shavitt and E. Shir. DIMES: Let the Internet measure itself. In *ACM SIGCOMM Computer Communication Review*, volume 35, 2005.
- [98] Y. Shavitt, E. Shir, and U. Weinsberg. Near-deterministic inference of AS relationships. In *ConTEL 2009*, pages 191–198, June 2009.
- [99] Y. Shavitt and N. Zilberman. A geolocation databases study. *IEEE Journal on Selected Areas in Communications*, 29(9), 2011.
- [100] Y. Shavitt and N. Zilberman. Arabian nights: measuring the arab internet during the 2011 events. *Network, IEEE*, 2012.
- [101] Y. Shavitt and N. Zilbermani. Geographical internet pop level maps. In *Proceedings of the 4th international conference on Traffic Monitoring and Analysis (TMA'12)*, 2012.
- [102] R. Siamwalla, R. Sharma, and S. Keshav. Discovering Internet Topology. Technical report, Cornell University, 1998.
- [103] S. S. Siwipersad, B. Gueye, and S. Uhlig. Assessing the geographic resolution of exhaustive tabulation for geolocating Internet hosts. In *Proceedings of the 9th international conference on Passive and Active Measurement (PAM'08)*, volume 4979, pages 11–20, 2008.
- [104] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '02)*, pages 133–145, 2002.
- [105] N. T. Spring, R. Mahajan, D. Wetherall, and T. E. Anderson. Measuring ISP topologies with Rocketfuel. *IEEE/ACM Transactions on Networking*, 12(1):2–16, 2004.
- [106] Sprint. Global IP network. [https://www.sprint.net/network\\_maps.php](https://www.sprint.net/network_maps.php).
- [107] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz. Characterizing the internet hierarchy from multiple vantage points. In *IEEE INFOCOM*, New York, NY, USA, Apr. 2002.
- [108] D. Svantesson. E-commerce tax: How the taxman brought geography to the “borderless” Internet. *Revenue Law Journal*, 17.1, 2007.



- [109] R. Teixeira, K. Marzullo, S. Savage, and G. M. Voelker. In search of path diversity in ISP networks. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement (IMC'03)*, pages 313–318, 2003.
- [110] TeleGeography. IPTV Broadband Penetration Reaches 15 Percent, Growth Prospects are Patchy.  
*http://www.telegeography.com/press/marketing-emails/2012/06/20/iptv-broadband-penetration-reaches-15-percent-growth-prospects-are-patchy/index.html*, 2012.
- [111] TeliaSonera International Carrier. Network map. <http://www.teliasoneraic.com/NetworkFlash/index.htm>  
[Accessed: October 10,2012].
- [112] M. Thomson and J. Winterbottom. Discovering the local location information server (LIS).  
<http://tools.ietf.org/html/draft-ietf-geopriv-lis-discovery-15>, 2010.
- [113] United States Census Bureau. Metropolitan Statistical Areas–Population by Age: 2010.
- [114] United States Census Bureau. Metropolitan Statistical Areas–Population by Race: 2010.
- [115] United States Department of Commerce. Bureau of Economic Analysis. <http://www.bea.gov/>, 2013.
- [116] United States Department of Transportation. Bureau of Transportation Statistics.  
<http://www.rita.dot.gov/bts/>, 2013.
- [117] University of Oregon Advanced Network Technology Center. Route views project.  
<http://www.routeviews.org/>.
- [118] US Department of Commerce, Bureau of Economic Analysis. Personal Income and Per Capita Personal Income by Metropolitan Area, 2009 - 2011).  
*http://www.bea.gov/newsreleases/regional/lapi/lapi\_newsrelease.htm*, 2012.

- [119] US Department of Commerce, Bureau of Economic Analysis. Real GDP by Metropolitan Area (millions of chained 2005 dollars). <http://www.bea.gov/iTable/iTable.cfm?reqid=70&step=1&isuri=1&acrdn=2#>, 2013.
- [120] US Department of Transportation, Bureau of Transportation Statistics. Highway Congestion in the 50 Largest Urban Areas: 2010. [http://apps.bts.gov/publications/state\\_transportation\\_statistics/state\\_transportation\\_statistics\\_2011/html/table\\_05\\_05.html](http://apps.bts.gov/publications/state_transportation_statistics/state_transportation_statistics_2011/html/table_05_05.html), 2011.
- [121] US Department of Transportation, Bureau of Transportation Statistics. Passengers Boarded at the Top 50 U.S. Airports. [http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national\\_transportation\\_statistics/html/table\\_01\\_44.html](http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national_transportation_statistics/html/table_01_44.html), 2013.
- [122] US Department of Transportation, Bureau of Transportation Statistics. Top U.S. Foreign Trade Freight Gateways by Value of Shipments. [http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national\\_transportation\\_statistics/html/table\\_01\\_51.html](http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national_transportation_statistics/html/table_01_51.html), 2013.
- [123] S. Vinciguerra, K. Frenken, and M. Valente. The geography of internet infrastructure: An evolutionary simulation approach based on preferential attachment. *Urban Studies*, 47(9):1969–1984, 2010.
- [124] X. Wang and D. Loguinov. Understanding and modeling the internet topology: economics and evolution perspective. *IEEE/ACM Trans. Netw.*, 18(1):257–270, Feb. 2010.
- [125] W. Willinger and M. Roughan. Internet topology research redux. *ACM SIGCOMM eBook: Recent Advances in Networking*, 2013.
- [126] B. Wong, I. Stoyanov, and E. G. Sirer. Octant: A comprehensive framework for the geolocalization of Internet hosts. In *Proceedings of the 4th USENIX Symposium on Networked Systems Design and Implementation (NSDI'07)*, 2007.

- [127] XO. Complete network assets. [http://www.xo.com/SiteCollectionImages/about-xo/xo-network/maps/map\\_complete\\_1600.gif](http://www.xo.com/SiteCollectionImages/about-xo/xo-network/maps/map_complete_1600.gif) [Accessed: October 10,2012].
- [128] S.-H. Yook, H. Jeong, and A.-L. Barabási. Modeling the Internet's large-scale topology. *Proceedings of the National Academy of Sciences*, 99(21):13382–13386, Oct. 2002.
- [129] K. Yoshida, Y. Kikuchi, M. Yamamoto, Y. Fujii, K. Nagami, I. Nakagawa, and H. Esaki. Inferring PoP-level ISP topology through end-to-end delay measurement. In *Proceedings of the 10th international conference on Passive and Active Measurement (PAM'09)*, volume 5448, pages 35–44, 2009.
- [130] M. Zhang, Y. Ruan, V. Pai, and J. Rexford. How DNS misnaming distorts Internet topology mapping. In *Proceedings of the annual conference on USENIX '06 Annual Technical Conference (ATEC'06)*, pages 34–34, 2006.
- [131] Y. Zhang, R. Oliveira, H. Zhang, and L. Zhang. Quantifying the pitfalls of traceroute in as connectivity inference. In *Passive and Active Measurement*, volume 6032 of *Lecture Notes in Computer Science*. 2010.
- [132] J. Zupan. *Clustering of large data sets*. Chemometrics research studies series. Research Studies Press, 1982.





## תקציר

ספקי שירות נוטים למקם נתבי תקשורת רבים בנקודה אחת הנקראת (PoP) Point of Presence, והמשמשת לתקשורת באזור גיאוגרפי מסוים. בניית מפות ברמת ה-PoP הוא נושא בעל ענין רב בשל חשיבותו לתחומים רבים, לדוגמה, למעקב אחר התפתחות האינטרנט ולימוד מאפייניו. בתזה זו מוצגת גישה מובנה לזיהוי לבניה אוטומטית של מפות ברמת ה-PoP בקנה מידה גדול באמצעות מדידות traceroute ממיקומים מרובים.

בניית מפת PoPs מתחילה בזיהוי מבני שלהם, ולאחר מכן מוקצה מיקום באמצעות מידע ממספר מאגרי מידע של מיקום גיאוגרפי. העבודה דנה בפשרות שבגישה זו ומספקת ולידציה נרחבת של האלגוריתם. המפות הנוצרות יכולות להיות בשימוש נרחב למחקר, וכמה כיוונים אפשריים ניתנים בעבודה.

המיקום הגיאוגרפי של כתובות IP באינטרנט חשוב לצרכי מחקר אקדמי, מסחרי ויישומי אבטחה שונים. בשל כך קיימים מאגרי מידע וכלים מסחריים ואקדמיים רבים הממפים כתובות IP למיקומים גיאוגרפיים. הערכת הדיוק של שירותי מיפוי אלה הינה מורכבת שכן קשה מאד להשיג מידע מבוסס על נכונות הנתונים. בחלק השני של עבודה זו נבחנת מידת הדיוק של שירותי מיפוי שונים תוך שימוש באלגוריתם שהוצג בחלק הראשון. בדרך זו ניתן לבחון קבוצה של כ-100,000 כתובות ה-IP ברחבי העולם מתוך ידיעה כי כתובות השייכות לאותו ה-PoP בביטחון גבוהה חולקות את אותו מיקום גיאוגרפי. אני מספקת תובנות על נקודות החוזק והחולשה של שירותי המיקום השונים ודנה במידת הדיוק שלהם ובאנומליות שונות. המחקר מראה כי בעוד שמסדי נתונים מסחריים טוענים לרמה מאד גבוהה של דיוק, נכונותם של מסדי הנתונים שלהם מוטלת בספק. כלים אקדמיים, המבוססים על מדידות עיכוב, הוכחו כבעלי טווח רחב של שגיאה גם כן.

העבודה מציגה אלגוריתם מיקום חדשני המשתמש בגרף האינטרנט ברמת ה-PoP על מנת לשפר את רמת הדיוק של מיקום גיאוגרפי, תוך שילוב מידע ממספר מאגרי מיקום IP ומדידות השהיה. האלגוריתם משתמש ב-PoPs עם רמת בטחון גבוהה במיקום (כהגדרתה בחלק הראשון של התזה) על מנת לשפר את מיקומם של PoPs עם רמת ביטחון נמוכה יותר, בצורה איטרטיבית, ולאחר מכן למקם גיאוגרפית כתובות IP בדידות. העבודה מראה כי במקרים רבים התוצאות הניתנות על ידי האלגוריתם מדויקות יותר מאשר ממאגרי מידע מיקום גיאוגרפי, תוך הימנעות ממלכודות הקיימות בשל שימוש במדידות השהיה.

מחקר רב נעשה על מנת להסיק את היחסים המסחריים שלא נחשפו בין מערכות אוטונומיות (ASes). מערכות יחסים אלה, שמבחינות לרוב בין ארבעה סוגי קשרים (ToRs), מכתיבות את מדיניות הניתוב בין ASes. החלק הבא של עבודה זו ממנף את המפות ברמת ה-PoP על מנת לשפר היקש ToR. היא מציעה שיטה המשתמשת במפות ברמת ה-PoP למציאת קשרים מורכבים וזיהוי אנומליות ביחסים בין מערכות אוטונומיות. אני מציגה תוצאות ניסוי המשתמש בשיטה על גבי נתוני ToR שדווחו על ידי CAIDA ומתארת סוגי חריגות וטעויות שזוהו. התוצאות מדגימות את היתרונות של שימוש במפות ברמת ה-PoP עבור היקש ה-ToR וכי דרושים משאבים ניכרים פחות מאשר בשיטות אחרות המסוגלות באופן תיאורטי לאתר תופעות דומות.

החלק האחרון בעבודה זו מניח את היסודות לפיתוח מודל אבולוציה של האינטרנט המבוסס על רמת ה-PoP. לטופולוגיות PoP של האינטרנט מצורף מידע גיאוגרפי, כלכלי ודמוגרפי על מנת להשיג הבנה של מבנה הרשת בתקופות זמן שונות, על מנת לזהות את חוקי היסוד של אבולוצית האינטרנט. נתונים אלה יכולים לשמש לפיתוח גנרטור טופולוגית רשת מציאותי ומסגרת תחזית אמינה שיכולים לשמש כדי לחזות את הצמיחה של רשת האינטרנט ככל שכלכלות ממשיכות לצמוח, נתונים דמוגרפיים משתנים, וכאשר מקומות שונים בעולם מתחברים לראשונה לרשת.



עבודה זו נעשתה בהדרכת

פרופ' יובל שביט





הפקולטה להנדסה ע"ש איבי ואלדר פליישמן

בית הספר לתארים מתקדמים ע"ש זנדמן-סליינר

## גרף האינטרנט ברמת ה-PoP

### נעה זילברמן

חיבור לשם קבלת התואר "דוקטור לפילוסופיה"

הוגש לסנאט של אוניברסיטת תל-אביב

עבודה זו נעשתה באוניברסיטת ת"א בפקולטה להנדסה

בהדרכת פרופ' יובל שביט



הפקולטה להנדסה ע"ש איבי ואלדר פליישמן

בית הספר לתארים מתקדמים ע"ש זנדמן-סליינר

## גרף האינטרנט ברמת ה-PoP

נעה זילברמן

חיבור לשם קבלת התואר "דוקטור לפילוסופיה"

הוגש לסנאט של אוניברסיטת תל-אביב