A Geolocation Databases Study

Yuval Shavitt
Tel-Aviv University, Israel

Noa Zilberman
Tel-Aviv University, Israel

***Abstract*—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. In this work we evaluate mapping services using an algorithm that groups IP addresses to PoPs, based on structure and delay. This way we are able to group close to 100,000 IP addresses world wide into groups that are known to share a geo-location with high confidence. We provide insight into the strength and weaknesses of IP geolocation databases, and discuss their accuracy and encountered anomalies.**

I. Introduction

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 every day 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 havock 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 [19] 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 [22] notes that geolocation can be used to monitor remote access and prevent login using stolen passwords or login ID. It can only be assumed 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. The geolocation information provides means to reduce the risk, for example by blocking users from certain high-risk countries and cross-referencing user expected and actual location.

The IETF has also commenced in defining standards for geolocation and emergency calling through IETF GEOPRIV working group [23], 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 [42]. 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 ([41], [12], [26] and more). However, not many works have focused on the accuracy of geolocation databases. In 2008, Siwpersad *et al.* [38] examined the accuracy of Maxmind [30] and IP2Location [18]. 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.* [16] 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 [32] conducted a survey of geolocation techniques used by geolocation databases and examined means for evasion/circumvention from a security standpoint.

Improving location accuracy by measurements has been addressed by several works in the recent years. IP2Geo [33] 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 [17], which uses several delay constraints to infer the location of a network host by a triangulation-like method. Later works, such as Octant [43] use a geometric approach to localize a node within 22 mile radii. Katz-Bassett *et al.* [25] suggested topology based geolocation using link delay to improve the location of nodes. Yoshida *et al.* [44] used end-to-end communication delay measurements to infer PoP level topology between thirteen cities in Japan. Laki *et al.* [27] increased geolocation accuracy by decomposing the overall path-wise packet delay to link-wise components and were thus able to approximate the overall propagation delay along the measurement path.

Eriksson *et al.* [6] apply a learning based approach to improve geolocation. They reduce IP geolocation to a machine learning classification problem and use a Naive Bayes framework to increase geolocation accuracy.

In this paper, 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 the geolocation databases against. We avoid this need using a different approach, we use an algorithm, whose main features are summarized in Section III-A, for mapping IP addresses to PoPs (Points of Presence). The algorithm, based both on delay measurements and graph structure, has a very small probability to map 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 IV-A. We take a step further and compare multiple databases results for the same PoP (Section IV-C) and study their spread.

II. 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.* [39] 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 [45]. In addition, rules for inferring the locations of all DNS names do not exist, and require some manual adjustments. Google Gears provides a set of geolocation API [13] that allows 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 devices has access to (nearby cell sites, WiFi nodes, etc.) to a third-party location service provider, who resolves the signals into a location estimate [14]. Thus, the service granularity is based on a single IP address granularity and not on address blocks. HostIP.Info [20] 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, entirely or on top of other methodologies, in order to improve geolocation data quality. While many of the measurement based geolocation services that we discussed in Section I do not provide the ability to query specific IP addresses [25], [43], [44], one online geolocation service that does allow it is Spotter, which is based on a work by Laki *et al.* [28]. 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 landmarks 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 determine the targets 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 [30] 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 netspeed can be retrieved as well. In addition to all the above, MaxMind suggests 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 [1] is a free geolocation service that uses MaxMind GeoIP lite database and adds on top of it reserved addresses and optional timezone.

IPligence [24] is a geolocation service provider, existing since 2006. ItsThe 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 [18], a geolocation database with a wider 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 [3], Digital Element's Netacuity Edge and Geobytes. Each of these companies praise themselves with large tier-1 customers from different fields, who use their services for targeted advertising, fraud prevention, and more.

Quova [2], 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 [3] was founded in 1998 and launched its commercial service in 1999. It provides through Edge Platform product IP location information. 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 [4], founded in 2005, publishes under the products NetAcuity and NetAcuity Edge two levels of geolocation information, 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 [9] launched in 2002 its GeoSelect product, for geolocation information. The wealth 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 include also gathering information from websites that require users to enter their location information and then processing this data onto Geobytes' infrastructure map of the Internet [31]. 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.

A. Databases Accuracy

The geolocation service provider is, in many cases, the sole source for database accuracy information. Some vendors do not publish such figures at all, such as IPligence, 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 I presents a summary of accuracy figures, as given by the geolocation service providers on their websites [2], [3], [4], [9], [18], [30]. The table includes information on country level, city level world wide level and the USA city level accuracy.

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 [29]. 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 Pricewaterhouse Coopers [34], which used 3 reference third party databases.

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% †

TABLE I

GEOLOCATION DATABASE ACCURACY AS REPORTED BY VENDOR

†US State level accuracy

The accuracy of the figures in Table I 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 [11], 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 debateable, as described in Sec. I.

III. THE EVALUATION MODEL

A. Building PoP Maps

A PoP is a group of routers which belong to a single AS and are physically located at the same building or campus. In most cases [15], [36] 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 core routers are connected to the core network of the ISP. The algorithm we use for PoP extraction was first suggested by Feldman and Shavitt [7] and later improved by Shavitt and Zilberman [37]. The algorithm looks for bi-partite subgraphs with certain weight constraints in the IP interface graph of an AS; no aliasing to routers is needed. The bi-partites serve as cores of the PoPs and are extended with other close by interfaces.

The initial partitioning removes all edges with delay higher than PD_{max_th}, PoP maximal diameter threshold, and edges with number of measurements below PM_{min_th}, the PoP measurements threshold. PM_{min_th} is introduced in order to consider only links with a high reliable delay estimation to avoid false indication of PoPs. The result non-connected graph G' contains induced sub graphs, each is a candidate to become one or more PoPs. There are two reasons for a connected group to include more than a single PoP. The first and most obvious reason is geographically adjacent PoPs, e.g., New York, NY and Newark, NJ. The other is caused by wrong delay estimation of a small amount 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.

Next, the algorithm checks if each connected group can be partitioned to more than one PoP, using parent-child classification according to the measurement direction in the bipartite graph. Further localization is achieved by dividing the parents and children groups into physical collocations using the high connectivity of the bipartite graph. If *parent pair* and *child pair* 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 the pair of groups is declared as part of the same PoP. Last, a unification of loosely connected parts of the PoP is conducted. For this end, the algorithm looks for connected components (PoP candidates) that are connected by links whose median distance is very short (below PD_{max_th}). In the original algorithm [7], an additional step was implemented, called Singleton Treatment, in which nodes with only one or two links are assigned to PoPs based on their median distance. This step may add to the PoP IP addresses that are not necessarily part of it. Thus, in this work, two PoP level maps were generated: one map without any singletons, which is considered to be accurate looking at the PoP IP addresses only, and a second map that includes singletons. The aim of the second map is to improve location estimation where PoP location is undetermined based on the first map only. As the singletons are necessarily in the vicinity of the PoP, using them does not harm the locations estimation.

One of the motivations for this work is the lack of publicly available ground truths for validating both PoPs structure and IP geolocation. Thus, validating such algorithms was always a difficult task. A previous work [37] focused on the algorithm validation and correctness using stability over time, sensitivity to parameters, and other indirect means. In addition, it reported DNS based validation of fifty PoPs. For this work we additionally validated 23 PoPs against publicly available IP addresses in GEANT, the pan-European academic network, and Proxad, a French ISP. and found no errors.

For this paper's purposes, the thresholds sensitivity should be mentioned, as they may affect the geolocation accuracy. Figure 1 explores the PoP extraction algorithm's sensitivity to PD_{max_th}. In the figure five ISPs are explored: Level 3, AT&T, Comcast, MCI, and Deutsche Telekom. The figure presents the number of IPs included in PoPs when changing PD_{max_th}. 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 selected to be $5ms$, as it presents a good tradeoff between delay measurement's error and location accuracy. The number of IPs included in PoPs decreases as the minimal number of required measurements, PM_{min_th}, increases, as can be expected (see [37]). In our extracted PoP maps, PM_{min_th} was selected to be 5.

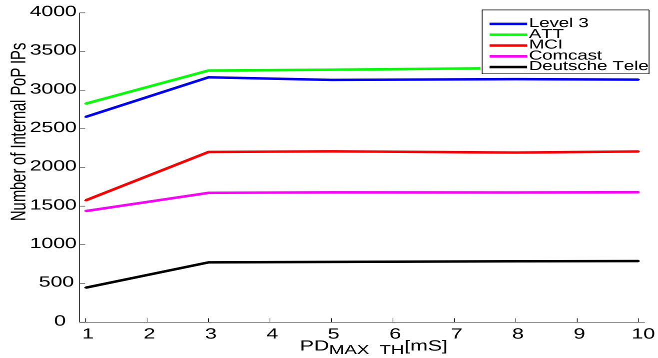

Fig. 1. Number of IPs in PoPs vs. Maximal Delay

B. Data Evaluation Method

The geolocation databases evaluation is conducted using the classification of IP addresses into PoPs as described above. 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.

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, introduced in [37], operates as follows:

Initial Location Each of the evaluated geolocation databases is queried for the location (longitude, latitude) of each IP included in the PoP. Next, the center 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, median provides a better suited starting point. Median does not guarantee good results, for example, if there is complete disagreement between geolocation databases as for the location of a PoP (see Figure 14) . However, since geolocation databases are typically reliable in country-level assignment, such examples are rare.

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, and 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 the convergence range.

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 [7] 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 consequent 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.

The PoP geolocation generated maps were validated for correctness by comparing them against PoP maps published by the ISPs, such as Sprint [40], Qwest [35], Global Crossing [10], and others. The location was validated against listed PoPs by cities, when available, or eyeballed otherwise. We also validated the PoPs list with collaborating ISPs such as GARR [8]. In addition, we reported [37] a small scale testing of the geolocation accuracy based on 50 known university locations. The test was based only on three databases: Maxmind, IPligence and HostIP.info. For 49 out of 50 universities, the location was accurate within a $10km$ radius. The last PoP, belonging to the University of Pisa, was located by the algorithm in Rome, due to an inaccuracy in the MaxMind and IPligence databases. Only Hostip.Info provided the right coordinates for this PoP. PoP locations were also validated against their DNS name, whenever a DNS name was available.

C. Dataset

The collected dataset for PoP level maps is taken from DIMES [5]. We use all traceroute measurements taken during March 2010, totaling 126.7 million, namely 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 2. 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 PoP level maps were generated by the PoP extraction algorithm,

Fig. 2. Map Of DIMES Agents, March-2010

Fig. 3. Map Of Discovered PoPs, March-2010

with and without singletons added. A total of 3800 PoPs where discovered, containing $52K$ IP addresses from the first run, and $104K$ IP addresses from the second run, meaning 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 3 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.

IV. Results

A. Basic Tests

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

	Core PoP IP		With Singletons	
Database	**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 II
NULL IP ADDRESS INFORMATION

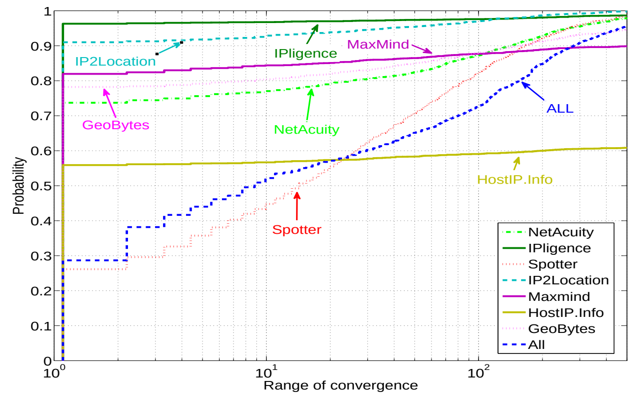

Fig. 4. Range of Convergence Within Databases

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 II 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 come to 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 does 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 be either 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, where 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 does return longitude and latitude information, which are the center of the country where the IP is located. DNS NULL replies are 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.

2) 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 the assumption: in all the PoPs evaluated, with no exception, there are always databases that support the PoP vicinity requirement.

We run the algorithm separately per database. Figure 4 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. IPligence 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 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 that databases with some NULL replies. We further explore this question in Section IV-C.

Figures 5 and 6 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 5 we set a radius of 100km and in Figure 6 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.

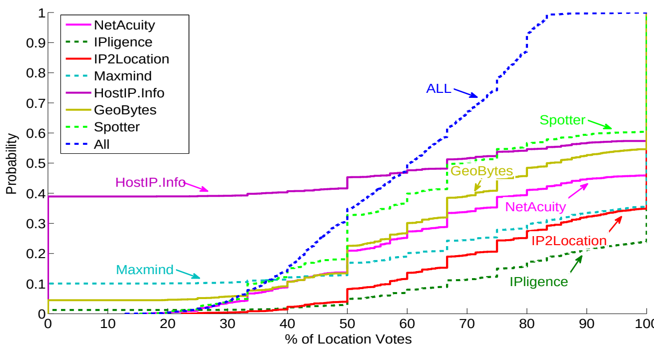

Fig. 5. CDF of Location Votes Percentage Within 100km From PoP Center

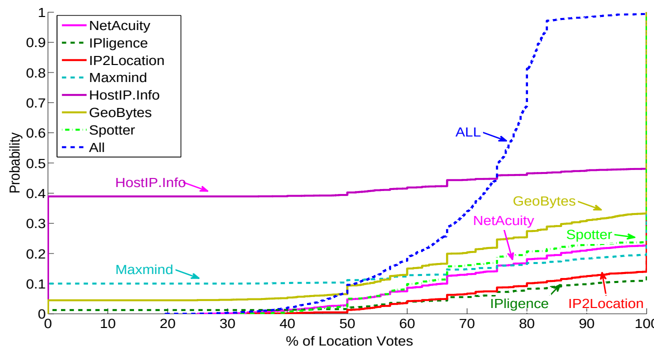

Fig. 6. CDF of Location Votes Percentage Within 500km From PoP Center

For all databases there are PoPs with no majority vote, meaning that less than 50% of the location votes were within the tested radius. IPligence 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.

B. 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 [21], 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. 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

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 III
COMPARISON WITH CAIDA'S GROUND TRUTH DATABASE

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, thus may bias the results.

Table III 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 except for NetAcuity, none of the databases is close to its acclaimed accuracy on 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 equivalent 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 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.

C. 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

†US State level accuracy

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 7 and 8 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 7) are explained by Figure 9. The figure presents for several selected database pairs the CDF of distance vectors. 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. 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 10 depicts for each database the CDF of the deviation of each IP from the PoP median location. The interesting observations here are at the $40km$ range, which is a city range, and $500km$ range, which can be referred to as a region. IPligence, MaxMind and IP2Location

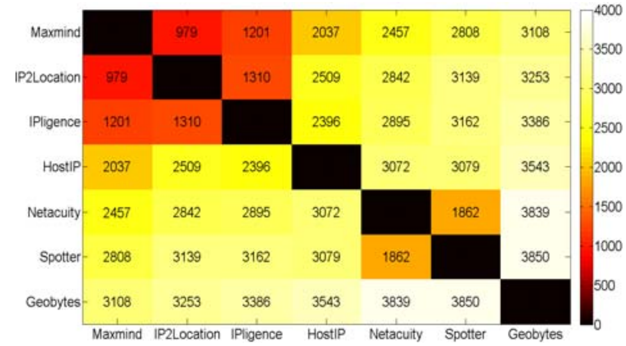

Fig. 7. RMS Distance[km] Between Databases - Heatmap

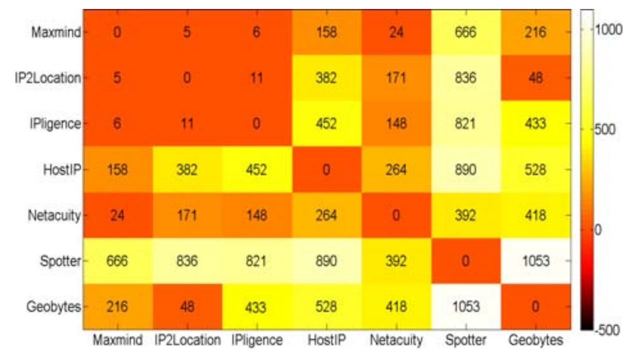

Fig. 8. Median Distance[km] Between Databases - Heatmap

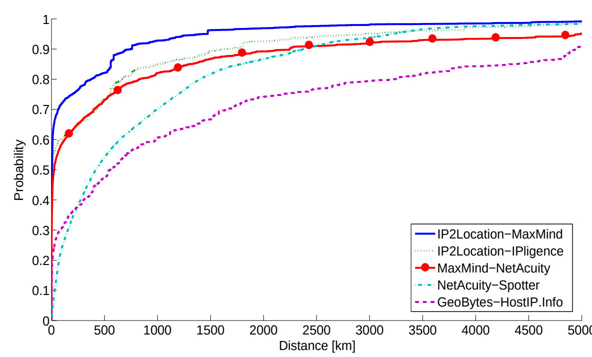

Fig. 9. CDF of Distances[km] Between Databases

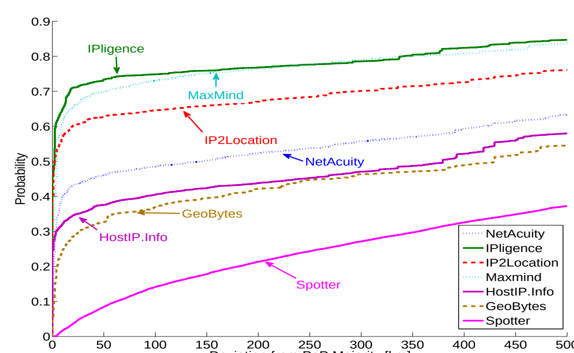

Fig. 10. CDF of Database Location Deviation From PoP Median - 500km Range

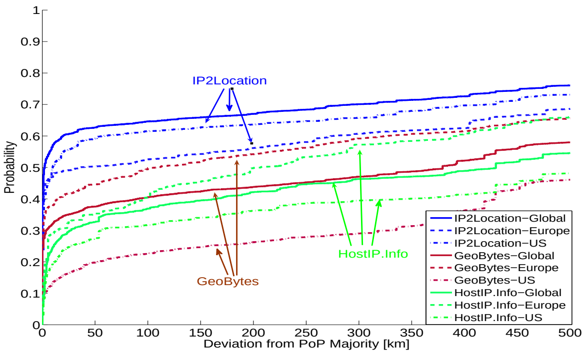

Fig. 11. Breakdown of deviation from PoP majority CDF By region - 500km Range

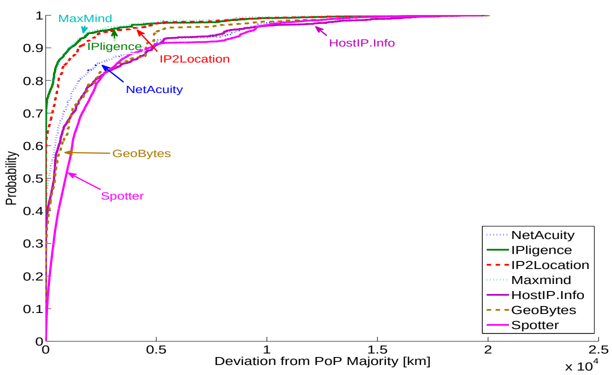

Fig. 12. CDF of Database location deviation from PoP majority

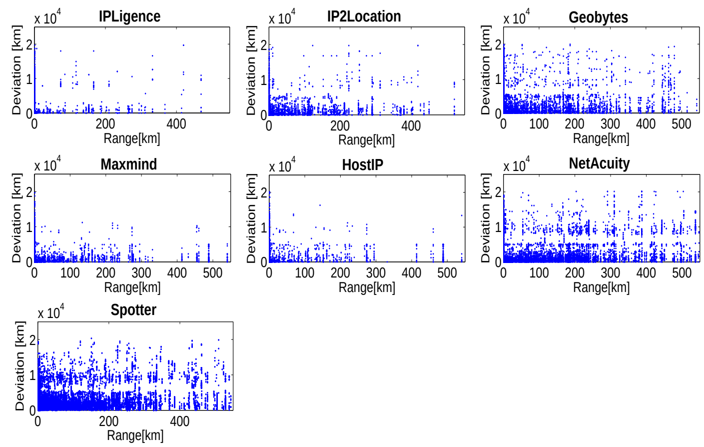

Fig. 13. Database Location Deviation From PoP Median vs. Range of Convergence

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 10 with the median heatmap of Figure 8 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 10) 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 8. Spotter values are above 500km, and indeed in Figure 10 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 11. Other databases, as IP2Location, have greater deviation in Europe than the rest of the world. For clarity, only three of the databases are shown in Figure 11. 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 12 shows that in some databases this tail can hold 15% of the IP addresses. Although the majority vote may be incorrect, this points that at least one of the databases is significantly far off from the real IP address location.

Figure 13 depicts for each database a scatter plot of the range of convergence (X axis) versus the deviation of the IP location from the PoP median location (Y axis). 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 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 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.

D. 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 where 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 [11] 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 14 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.

The mismatch between databases is not uncommon. Some examples exist inside the United States, too: in Figure 15 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).

Fig. 14. Mismatch Between Databases - UUNET

Fig. 15. Mismatch Between Databases - Global Crossing

The 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.

E. Database Changes

One of the motivations to update geolocation databases is the claim that IP geolocation changes significantly over time. Maxmind [30] claim that it looses accuracy at a rate of approximately 1.5% per month. IP2Location [18] states that on average, there are 5%-10% of the records being 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 IPligence, 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.

V. Discussion

Before we discuss our results, it is important to note that the paper is based on a PoP extraction algorithm, and thus relies on its accuracy. The validation that we described here and in previous papers 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 8 or 14), 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 certainty on their indicated location (as shown in Figure 15).

A. Active Measurement Accuracy

Active measurements are used by many geolocation services [25], [43], [28] and by other projects for different localization tasks, most notably for assigning IP addresses to PoPs [39]. Spotter geolocation is based solely on active measurements, thus we selected to study its performance to a greater depth due to the importance of understanding the limitations of this approach.

Figures 16 and 17 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* 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) 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).

*We consulted Peter Haga and Peter Matray from the Spotter project on this aspect.

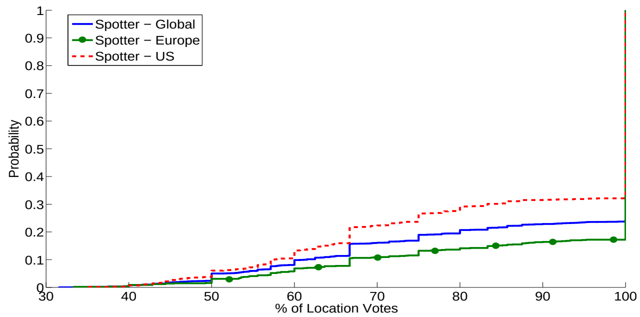

Fig. 16. Breakdown of location votes percentage CDF for Spotter by region.

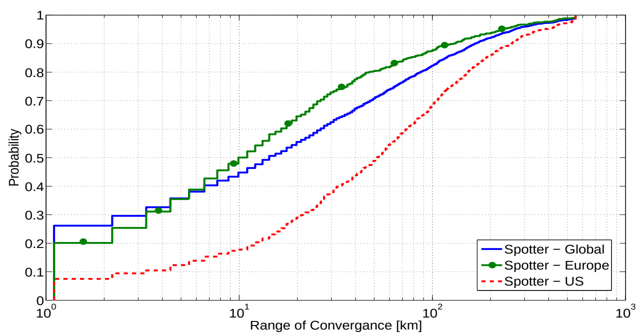

Fig. 17. Breakdown of convergence range CDF for Spotter by region.

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

VI. Conclusion

This paper 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 provides 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. Future research in this field should focus on means to decide on ground truth when there is a disagreement between the databases.

VII. Acknowledgements

We would like to thank Peter Mátray and Peter Hága, as well as other members of the Spotter project [28] for providing us with their dataset and helping this work. We would also like to thank Frank Bobo from Digital Envoy, Adrian McElligott from Geobytes and Edward Lin from Maxmind for helping us in obtaining their databases and answering our questions. This work was partially funded by the OneLab II and the MOMENT consortia, which are partly financed by the European Commission; and by the Israeli Science Foundation, center of knowledge on communication networks (grant #1685/07)

References

[1] IPInfoDB. http://ipinfodb.com, 2010.
[2] Quova. http://www.quova.com, 2010.
[3] Akamai. EdgePlatform. $http://www.akamai.com/$, 2010.
[4] Digital Envoy. NetAcuity Edge. $http://www.digital-element.com/our\_technology/edge.html$, 2010.
[5] DIMES. Distributed Internet Measurement and Simulations. http://www.netdimes.org/.
[6] B. Eriksson, P. Barford, J. Sommers, and R. Nowak. A learning-based approach for IP geolocation. In *Passive and Active Measurement*, pages 171–180, 2010.
[7] D. Feldman and Y. Shavitt. Automatic large scale generation of internet PoP level maps. In *GLOBECOM*, pages 2426–2431, 2008.
[8] GARR, The Italian Academic and Research Network. http://www.garr.it/eng/index.php.
[9] Geobytes. GeoNetMap. $http://www.geobytes.com/$, 2010.
[10] Global-Crossing. Global Crossing Network. http://www.globalcrossing.com/html/ map062408.html.
[11] Gomez Inc. Akamai ip location service performance assessment, akam01, September 2009.
[12] G. Goodell and P. Syverson. The right place at the right time. *Commun. ACM*, 50(5):113–117, 2007.
[13] Google. Geolocation API. $http://code.google.com/apis/gears/api\_geolocation.html$, 2010.
[14] Google Gears Wiki. Geolocation API. $http://code.google.com/p/gears/wiki/GeolocationAPI$, 2010.
[15] B. R. Greene and P. Smith. *Cisco ISP Essentials*. Cisco Press, 2002.
[16] B. Gueye, S. Uhlig, and S. Fdida. Investigating the imprecision of IP block-based geolocation. In *PAM'07: Proceedings of the 8th international conference on Passive and active network measurement*, pages 237–240, 2007.
[17] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida. Constraint-based geolocation of internet hosts. *IEEE/ACM Trans. Netw.*, 14(6), 2006.
[18] Hexsoft Development. IP2Location. http://www.ip2location.com, 2010.
[19] Homeland Security. A roadmap for cyber security research. http://www.cyber.st.dhs.gov/docs/DHS-Cybersecurity-Roadmap.pdf, November 2009.
[20] hostip.Info. hostip.info. http://www.hostip.info, 2010.
[21] B. Huffaker, A. Dhamdhere, M. Fomenkov, and kc claffy. Toward topology dualism: Improving the accuracy of as annotations for routers. In *PAM*, pages 101–110, 2010.
[22] S. M. Huffman and M. H. Reifer. Method for geolocating logical network addresses, September 2005. US Patent 6947978.
[23] IETF. Geopriv workgroup. http://tools.ietf.org/wg/geopriv/.
[24] IPligence. IPligence Max. http://www.ipligence.com, 2010.
[25] E. Katz-Bassett, J. P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards IP geolocation using delay and topology measurements. In *The 6th ACM SIGCOMM conference on Internet measurement (IMC'06)*, pages 71–84, 2006.
[26] K. F. King. Geolocation and federalism on the internet: Cutting internet gamblings gordian knot. *Columbia Science and Technology Law Review*, Forthcoming, 2009.
[27] S. Laki, P. Mátray, P. Hága, I. Csabai, and G. Vattay. A model based approach for improving router geolocation. *Computer Networks*, 54(9):1490–1501, 2010.
[28] S. Laki, P. Mátray, P. Hága, and T. Seb˙Spotter: A model based active geolocation service. In *IEEE INFOCOM 2011*, 2011.
[29] MaxMind LLC. GeoIP City Accuracy for Selected Countries. $http://www.maxmind.com/app/city\_accuracy$, 2008.
[30] MaxMind LLC. GeoIP. http://www.maxmind.com, 2010.
[31] A. E. McElligott. Method and software product for identifying network devices having a common geographical locale, May 2007. US Patent Application 20070115998.
[32] J. A. Muir and P. C. V. Oorschot. Internet geolocation: Evasion and counterevasion. *ACM Comput. Surv.*, 42(1):1–23, 2009.

[33] V. N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for internet hosts. In *SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 173–185, 2001.
[34] Pricewaterhouse Coopers LLP. Quova - report of independant accountants. $http://www.quova.com/documents/PricewaterhouseCoopers\_Audit.pdf$, October 2008.
[35] Qwest. IP Network Statistics. http://66.77.32.148/index_flash.html.
[36] A. Sardella. Building next-gen points of presence, cost-effective PoP consolidation with juniper routers. White paper, Juniper Networks, June 2006.
[37] Y. Shavitt and N. Zilberman. A structural approach for pop geo-location. In *Infocom Workshop on Network Science for Communications (NetSciCom)*, March 2010.
[38] S. S. Siwpersad, B. Gueye, and S. Uhlig. Assessing the geographic resolution of exhaustive tabulation for geolocating internet hosts. In *Passive and Active Measurement*, volume 4979, pages 11–20, 2008.
[39] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with rocketfuel. In *ACM SIGCOMM*, pages 133–145, 2002.
[40] Sprint. Global IP Network. https://www.sprint.net/network_maps.php.
[41] D. Svantesson. E-commerce tax: How the taxman brought geography to the borderless internet. *Revenue Law Journal*, 17.1, 2007.
[42] M. Thomson and J. Winterbottom. Discovering the local location information server (LIS). http://tools.ietf.org/html/draft-ietf-geopriv-lis-discovery-15, March 2010.
[43] B. Wong, I. Stoyanov, and E. G. Sirer. Octant: A comprehensive framework for the geolocalization of internet hosts. In *NSDI*, 2007.
[44] 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 *PAM*, volume 5448, pages 35–44, 2009.
[45] M. Zhang, Y. Ruan, V. Pai, and J. Rexford. How DNS misnaming distorts internet topology mapping. In *ATEC '06: Proceedings of the annual conference on USENIX '06 Annual Technical Conference*, pages 34–34, 2006.