Volume 5, Issue 10, October-2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Web Host Geolocation Based on Probability Model for Latency Measurements and utilizing Maximum Likelihood Estimation Technique

Sk. Sameerunnisa
Asst Prof, C.S.E,
VJIT College of Engg,
Hyderabad, India

K. Raghunadh
H.R, B.Tech, M.B.A,
Advansoft Inc,
Hyderabad, India

Abstract— Accurately locating the geographical position of Internet hosts has many useful applications. Existing approaches for host Geolocation use Internet latency measurements, IP-to-location mapping and also geographical and demographical hints. Existing latency measurement-based Geolocation techniques use the observed latencies from multiple landmarks to the target host to determine maximum bound or both the maximum and minimum bounds of the geographical region where the target host is located. Due to the large variance of Internet latency measurements, the region constrained based on such maximum-minimum bounds tends to be relatively large resulting in large estimation errors. We propose a Geolocation algorithm, GeoWeight and Maximum Likelihood Estimation (MLE), GeoWeight improves the Geolocation accuracy by further limiting the possible target region by dividing the constrained region to sub regions of different weights. The weight assigned to a sub region indicates the probability of the target being in that sub region; a higher weight indicating a more probable region. By considering latency measurements from multiple landmarks and computing the resultant weights of overlapping regions a better con-strained target region can be obtained. MLE uses latency measurements from multiple hosts of known location to the host to be geolocated, to estimate the target location. This paper presents the GeoWeight algorithm and MLE, GeoWeight algorithm evaluates its performance using both synthetic and real data by geolocating target hosts in North America.MLE approach developed by analyzing a large set of data collected on the Planet Lab network test bed . MLE approach uses latency measurements from multiple hosts of known location to the host to be geolocated, to estimate the target location. We compare GeoWeight with two popular Geolocation techniques, Octant and CBG, by geolocating the same set of targets. The results for geolocating Internet hosts in North America confirms the validity of using GeoWeight algorithm and MLE with certainty as its accuracy is found to be better in comparison to existing techniques that are based on Internet latency outperforms existing techniques.

Keyword: GeoWeight,MLE, likelihood, Planet Lab, Octant, CBG.

I. INTRODUCTION

The problem of accurately locating the geographical location of an Internet host, referred to as host geolocation, has many useful practical applications. Internet location information can be lever-aged to improve the user experience and determine business strategy. Some examples of such location-aware applications include showing geographically targeted advertisements on web sites, automatic selection of a language to display web site content, web content delivery based on region, credit card fraud detection, and load balancing and resource allocation between Internet hosts. Our goal is to develop a scalable, reliable and robust IP Geolocation technique that is applicable to locate host on the Internet. Although intuitively it may appear that geolocation may be carried out using the IP addresses, it is not straightforward to do so because there is no direct mapping or one-to-one mapping between between IP addresses and geographical locations. The dynamic nature of IP address assignment makes the host geolocation in an IP environment even harder. Unlike the well studied related problem of wireless localization, Inter-net host localization is more challenging because transmission characteristics in the Internet are more complex than that in air. Transmission delay between the Internet hosts violate triangulation principle.

Current approaches for Geolocation are mainly based on end-to-end packet latency measurements from a set of nodes with known location to the node to be geolocated. Here in after we refer to such nodes of known location as *landmarks* and the node to be geolocated as the *target*. Based on the observed positive correlation between latency and distance travelled by data packets, these latency-based Geolocation techniques use latency measurements from landmarks to the target to constrain the estimated location of the target [1], [2], [3]. These approaches confine the region where the target is estimated to reside to within a maximum distance around each landmark. The use of a constraint for the minimum distance from the landmark, in addition to the maximum possible distance, is shown to improve the Geolocation accuracy. Such positive and negative distance factors are developed based on the maximum and minimum bounds of the distance to latency relationship. However, the variability of latency measurements between Internet hosts yields a significant disparity between the positive and negative distance bounds. As a result, area to which the target is

constrained using these methods is relatively large. Refining location information using additional geographical hints [18, 23] has shown to improve the geolocation accuracy. The location estimates are further refined using additional geographical information [1], [3]. The integration of the under-lying network topology information has been another method considered for improving Internet host Geolocation [4].

In this paper, we investigate the applicability of Maximum Likelihood Estimation (MLE) [5] technique and a novel Geolocation algorithm, GeoWeight, MLE and Geoweight for target Geolocation on the Internet, based on a statistical model for Internet latency. The application of MLE for target Geolocation was motivated by the probabilistic patterns observed in latencies measured between a set of landmarks over a period of time. We used the Internet Control Message Protocol (ICMP) ping, which measures the round trip delay of data packets, as the measure of latency. From latency measurements gathered over a period of time we derived the conditional probability density function (PDF) for latency given the distance travelled by packets. Using this PDF as the likelihood function we computed the MLE for a set of latency measurements from landmarks to the target to estimate its most likely position. We analyzed the applicability of MLE using both synthetic and real data and compared the results to the accuracy of two well accepted Geolocation techniques that are based on latency measurements [2], [3]. We geolocated the same set of targets using the three techniques. The results show that using MLE we are able to produce better accuracy compared to the other two techniques. GeoWeight algorithm ac-counts for the possible variability of distance (between the minimum and maximum possible distances) for a given latency by assigning weights to sub-regions within the region constrained by minimum and maxi-mum bounds. The weights are assigned to sub-regions to reflect the probability that the target could be located in the respective sub region; a higher weight indicating more probable regions. Latency measurements from multiple landmarks to the target result in intersecting regions. GeoWieght algorithm computes a weight for an intersecting region as the sum of weights of overlapping regions enclosed in the intersection. The location of the target is chosen as the centred of the intersection region having the highest computed weight. By assigning weights to sub-regions within the larger region, the GeoWeight algorithm is able to constrain the target location to a smaller region than that was possible from previous approaches, hence resulting in better estimation accuracies for geographical location. we present the GeoWeight algorithm and also the technique for computing weights for different regions. We evaluate the performance of the proposed algorithm using simulated and real distance vs. latency data. Through simulation, we specifically investigate two noise models for latency data, a Gamma distribution and a lognormal distribution, to understand the impact of the noise model on the accuracy of the algorithm. We evaluate the performance of the algorithm by geolocating 60 hosts in North America. The weights for different regions in this case are computed based on large set of latency vs. distance data we gathered over a month using 50 landmarks in North America using the PlatnetLab test bed. We compare our results by geolocating the same target hosts using two primarily-latency-based Geolocation techniques [23, 13] and the results show that our technique outperforms both these techniques.

II. RELATED WORK

Host Geolocation on the Internet is an important research problem that has been addressed by a number of research groups in the past. One of the intuitive approaches to host Geolocation is a comprehensive IP tabulation against physical locations which can be used as a lookup table [6]. However, because of the large number of available Internet hosts, such an approach does not scale very well. Also, a lookup table is difficult to maintain and keep up-to-date, especially, as it cannot take into consideration dynamic IP assignment. Three techniques for Geolocation were proposed in IP2Geo [18]: GeoTrack, GeoPing and GeoCluster. GeoTrack uses trace route information from a host to the target, which contains the list of routers encountered along the path. Using location hints from the DNS names of the routers along the path, the locations of the routers are determined. Of the routers whose locations are known, the closest one to the target is selected, and its location is chosen as the target location. The accuracy of the technique depends on the distance from the target to the nearest router of known location. Next, GeoPing works on the assumption that hosts that are geographically close have similar network delays with respect to other fixed hosts. By comparing the ping times to the target from a set of landmarks or probe machines with the ping times to a set of nodes at known locations, GeoPing estimates the target location to be the same as that of the node with known location having the most similar ping values. Thus, the accuracy of GeoPing is limited by the distance to the nearest probe. The third approach, GeoCluster, is a database lookup technique which groups IP addresses to clusters based on geographical proximity. This information is combined with the user registration database from web based services such as e-mail services. This technique suffers from the general problems related to database lookup-based approaches, such as reliability, scalability and maintainability issues and also unavailability of the user registration database for public access. Recent data-mining based approach Structon [7] is similar to GeoCluster except that it uses publicly available web pages instead of proprietary data sources in order to extract Geolocation information. Structon uses a three step approach. First, extracted Geolocation information from web pages are associated with their IP addresses. Then, this mapping information goes through multi-stage inference processes in order to improve the accuracy and coverage of its IP Geolocation repository of different IP segments. Finally, those IP segments that are not covered in the first two steps, are mapped with the location of the access router with the help of trace route tool. The accuracy of Structon implementation on the Internet depends heavily on the accuracy of extracted geographical mapping information. Moreover, with Structon [7], it is harder to get accuracy more than in the granularity of city level. Constraint-Based Geolocation (CBG) [13] uses ping times from landmarks as a measure of latency. For each landmark a maximum distance bound for a given latency is derived using distance-to-ping relationships observed between landmarks. During Geolocation the observed latencies from landmarks to the target are used to draw circles cantered at each land-mark based on the maximum distance bounds derived earlier.

The target is assumed to reside in the convex region resulting from the intersection of circles, and the target location is estimated as the centred of this convex region. This technique requires the target to be geographically well surrounded by landmarks. Similar to CBG, Topology-based Geolocation (TBG) [4] computes the possible location of the target as a convex region. In TBG, the maximum distance bound is obtained based on the maximum transmission speed of packets in fibre which gives a conservative estimate of the possible region. This region is further refined using inter-router latencies along the path from the target to the landmark, obtained from the trace route command. The final target location is obtained through a global optimization that minimizes average position error for the target and the routers. A more recently proposed measurement-based technique for Geolocation is Octant [23]. In contrast to other constraint based approaches that only limit the area where the target may be located, Octant also identifies areas where the target may not be located based on observed latencies (referred to as negative constraint). Octant expresses such information by considering two circles corresponding to the maxi-mum and minimum distances from each landmark to the target which constrains the possible geographical area where the target may be located. Upper and lower facets of the convex hull correspond to the maximum and minimum distance bounds. Different weights are assigned to different geographical areas based on the number of intersections (higher weights assigned to larger numbers of intersections). The final estimated region is the union of all regions, where the weight exceeds a desired weight or the region size exceeds a selected threshold. A Monte-Carlo algorithm is applied to pick the best single point location from the final estimated regions. These estimated regions in Octant often end up being disconnected parts. In contrast, it is highly unlikely with GeoWeight. As in GeoWeight the maximum (positive) and minimum (negative) distance bounds are divided into different weighted regions. Octant uses geographical and demographical constraints to improve the localization accuracy beyond its measurement only solution.

Finally, it used a force-directed algorithm to obtain target location. This approach requires a large number of landmarks thus experiences higher measurement costs. Additionally this technique is not scalable since the force-directed algorithm does not scale up well to large graphs. Our measurement based approach for geolocation differs from other approaches because we use a probability model for latency measurements to estimate the target location using the maximum likelihood approach. GeoWeight differs from other measurement based approaches because it uses a weighted model for latency-to-distance measurements to estimate the tar-get location.

III. PROBLEM FORMULATION

This section presents the problem statement of the Geolocation algorithm and GeoWeight.

A. Problem Statement

The problem considered here is the Geolocation of a target *H*. Let us denote the unknown position P_0 of the target in terms of its latitude and longitude (lat_0, lon_0).suppose that $\{L_1 , L_2 , L_3 \ldots\ldots L_N\}$ be a set of *N* landmarks. Let (lat_i , lon_i) be the latitude and longitude of the i^{th} land mark L_i. We carry out Geolocation using latency measurements from the *N* landmarks to the target. Let $t_i=\{t_{i,j}\}_{j=1}^{n_i}$ be the set of n_i,latency measurements from landmark *i* to the target. We denote the cumulative set of all measurements from landmarks to the host by: $t_{1:N} = \{t_1 , t_2 \ldots\ldots t_N\}$.
Our goal is to estimate the location p_0 of target *H* using measurements $t_{1:N}$. We denote the estimated location of the target by$\widehat{p_o}$.

Then the Geolocation error is defined as,

$$\epsilon = dist\,(p_o , \widehat{p_o}) \qquad (1)$$

where $dist(p_x ; p_y)$ represents the geographical distance between position p_x *and* p_y.

Fig. 1. Location of the chosen landmarks in North America

B. The Latency Model

Considering that Internet data packets in the majority of the cases travel through optical fibres, the minimum latency between two nodes that are a distance *d* apart can be given by,

$$t_{min}\,(\mathrm{d}) = \mathrm{d}/c_{fibre} \qquad (2)$$

Here c_{fibre} is the maximum transmission speed of data through the fiber, which is approximately 2⁄3 the speed of light [2]. As we show later, the latencies observed in real world Internet traffic are significantly higher than this lower bound due to factors such as router congestion. The observed latencies are modelled as,

$$t(d) = t_{min}\,(d) + E(d) \qquad (3)$$

Where $t(d)$ is the observed latency for distance *d* and $E(d)$ is a noise term that accounts for network delays in the real measurement.

Let $p\,(t|d)$ denotes the conditional probability density function of latency *t* given distance *d*.

IV. EXPERIMENTAL INVESTIGATION AND THE GEOLOCATION ALGORITHM

This section presents the results of the experiments we carried out to investigate the distance-to-latency relationship on the Internet. The goal of the experiments was to derive an empirical probability model for Internet latency as a function of distance. In the experiments we collected latencies (ICMP ping times) between landmarks over a period of time. By analyzing this data, we derived a probability distribution for Internet latency as a function of distance. The following subsection describes in detail the experiments, data analysis and the Geolocation algorithm.

A. Experimental Setup

Our experimentation was carried out on Planet Lab (www. planet-lab.org). We chose 50 Planet Lab hosts in North America as our landmarks as shown in figure 1. These landmarks were chosen such that all of them experience approximately similar latency against given distance ranges within our experimental region. North America covers a large geographical area and possesses substantial number of users, hosts and network connectivity of the Internet. Thus, we believe the methodology that we developed in this paper is notable even though we limited our scope to North America. However, the distance-latency relationship may vary based on different geographical location. The full data set we gathered consists of approximately 150,000 distance-latency measurements. Although we used 50 landmarks, measurements were not available for some landmarks due to the ping command not successfully completing on these Planet Lab nodes during the measurement collection period. The inter-landmark distance in the data covers the range 0.5 km - 4331 km. Figure 2(a) shows the relationship between distance and latency measurements of the complete dataset. The straight line plotted below the measured data shows the theoretical minimum latency as a function of distance, as defined by equation 2. In order to ascertain the observation of our Planet Lab dataset we also analyzed the dataset collected by iPlane [9] during same period. This dataset is based on latency measurements, shown in figure 2(b), between their 68 landmarks which similar to our landmarks are spread around North America.

B. Initial Observations

Figure 2(a) shows a plot of the distance vs. latency (approximately 150,000 data points) gathered on the Planet Lab test bed using 50 Planet Lab nodes in North America as landmarks. The Internet Control Message Protocol (ICMP) [3] ping delay between landmarks was used as the measure of latency. The solid line below the data points in figure 1(a) shows $t_{min}\,(d)$ given by equation 2. Figure 2 shows the histogram of observed distances for a given latency range (90.05 ms-100.00 ms). In order to ascertain the observation of our Planet Lab dataset we also analyzed the dataset collected by iPlane [1] during same period. This dataset is based on latency measurements, shown in figure 2(b), between their 68 landmarks which similar to our landmarks are spread around North America.

Following are five characteristics observed from this data (figure 2(a) and 2(b)):

- The minimum latency observed is higher than the theoretical minimum given by the equation 2.
- There is a positive correlation between latency and distance.
- A simple linear or non-linear relationship is not apparent in the data set - the data is noisy as described by equation 3.
- Although an upper bound on distance for a given latency is apparent, a lower bound is not apparent. This is the case even for the data analyzed on a per landmark basis.

For a given latency (or a latency range), some distances are more probable than other distances (Figure 2)

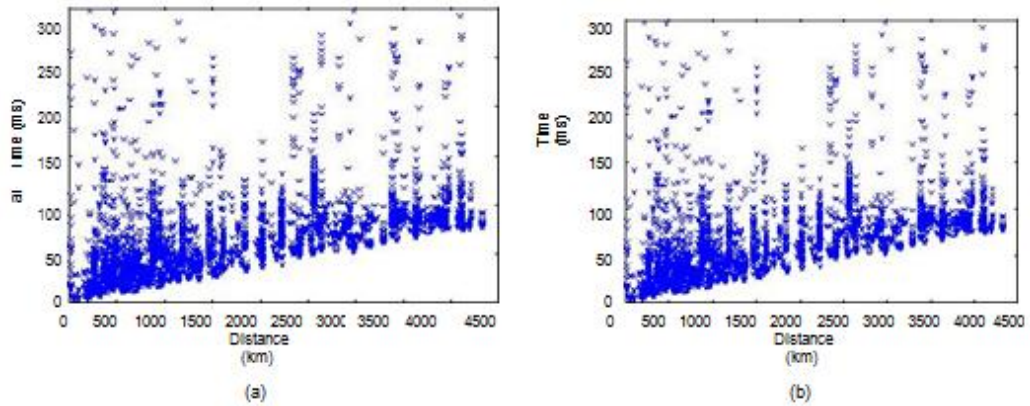

Figure 2: Distance-latency relationship (a) between 50 Planet Lab Landmarks (b)between iPlane dataset landmarks. The straight line below the data points shows delay to distance relationship according to equation

C. Modeling Internet Latency

Our goal is to derive a model for Internet latency as a function of distance between hosts. Let $L_{p,q,}$ be the latency measurements between originating landmark p and destination landmark q which are a distance $d_{p,q,}$ apart. Figure 2(a) consists of latency measurements of all such landmark pairs. We uniformly quantized the distances for all latency measurements of figure 2(a) to a discrete number of equal sized distance bins M.Let $d_{p,q,}$ belong to the m_{th} bin, $m = 1; :$

: : M; i.e. $d_m^{min} \leq d_{p,q} < d_m^{max}$,where d_m^{min} and d_m^{max} are the minimum and the maximum distances for the m_{th} bin. We considered 40 equal sized distance bins in our data analysis. The number of bins was chosen as 40 after careful observation of our total data set. If the total number of bin is higher, total number of points in each bin goes down to draw useful statistical conclusion for these bins. On the contrary, if we have fewer bins, each bin covers a larger distance range. Thus more distance values observed for MLE approach fall in similar bin and produce equal likelihood for more points. We now present our approach to modelling Internet latency using the data. Our experimental data shows an increasing trend of latency with distance, which is intuitive and also agrees with findings from past research [2], [3]. However, it can also be seen that for a given distance, there is a considerable variance in observed latency measures (Figure 2(a)), which establishes the need for modelling delay using a probability distribution. We believe that such an approach will allow more robust estimation of distance given a set of latency measurements.

First, we obtained a linear relationship between the mean distance and latency. To obtain this relationship we considered distance bins as described above and computed a mean latency for a distance bin by averaging all the latency measurements in the distance range of the bin. We then performed a least squares linear fit in the form of,

$$t = A * d + B$$

Where *d* is the mean distance for the bin, *t* is the mean latency for the bin and *A* and *B* are constants estimated from the least squares fit. Figure 3 shows the measured mean and variance of the data and the resulting linear fit. Constants *A* and *B* based on our experimental data were estimated to be 0.0186 and 12.8035 respectively. Next, we used measured data to estimate the conditional probability density function $p\ (t \mid d)$ we attempted to fit the latency data to a number of continuous, non-negative probability distribution functions using the distribution fitting function in MATLAB^{TM} Statistical Tool Box. In our experiment, lognormal distribution was chosen because of its skewed and non-negative properties [10] which have similarities to our real latency dataset. Also, the lognormal distribution provided the best fit for data in most distance ranges. Moreover, previous study [11] found lognormal characteristics in the Internet latency distribution.

It can be seen that the latency *t* for a given distance bin roughly follows a log normal distribution given by,

$$P\ (t \mid \mu, \sigma) = \frac{1}{\sigma t \sqrt{2\pi}} e^{\frac{-(ln(t)-\mu)^2}{2\sigma^2}}$$

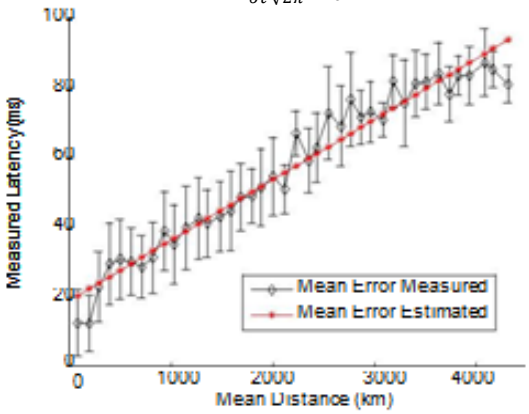

Fig. 3. Mean distance-to-latency relationship and the linear fit.

Where μ *and* σ are the mean and the standard deviation of the corresponding normal distribution. The values of where μ *and* σ in terms of the sample mean *m* and the variance *v* are given by,

$$\mu = \ln (m^2/\text{sqrt}\ (v+m^2)) \quad (4)$$

$$\sigma = \text{sqrt}\ (\ln\ (v/m^2+1)) \quad (5)$$

We computed the values of μ *and* σ using Equations 4 and 5, respectively, in two different approaches. In the first approach, the value of *m* used in the computation was the mean for the given distance obtained from the linear fit. Since the variance *v* was relatively independent of distance, it was taken as a constant and evaluated as the average variance over all distances. In the second approach, the values of *m* and *v* were the actual sample mean and the variance for the distance bins. The first and second graphs in Figures 4(a) and (b) show the computed values of μ *and* σ using approaches 1 and 2. We also estimated the parameters μ *and* σ for each distance bin using the lognormal distribution *log fit* provided by MATLAB for this purpose. The third graph in each figure shows the estimated values of μ *and* σ. It can be seen that the computed values closely match the estimated. Except for the first three distance bins. Although the mismatch for shorter distances can result in estimation errors, for the present work this simplistic model will be retained. However, similar observation, regarding shorter distances, is found in [11].We use the MLE technique as our geolocation algorithm. MLE is widely used for parameter estimation because if an asymptotically unbiased and minimum variance estimator exists for large sample sizes, it is guaranteed to be the MLE [12]. It is a block algorithm, as it operates on the accumulated set of all measurements. Let us define the parameter vector x as,

$$X=[lat_0 \quad lon_0]^T$$

Where *T* is the matrix transpose. The MLE is determined as the vector $\hat{x}_K^{ML}$ which maximizes the likelihood function $p\ (t_{1:N} \mid x)$,

$$\hat{x}_K^{ML} = \arg\max p\ (t_{1:N} \mid x)$$

X

Where k is, K=$\sum_{i=1}^{N} n_i$ is the total number of measurements.

Assuming the latency measurements to be conditionally independent, the likelihood function can be given by,

$$P\ (t_{1:N} \mid x) = \prod_{k=1}^{k} p(p_k \mid d_k(X)) \qquad (9)$$

Where $d_k(x)$ is the geographical distance between the land-mark corresponding to the *k*th measurement and the potential target location x.

More conveniently, the MLE can be obtained by maximizing the logarithm of the likelihood function as below.

$$\hat{x}_K^{ML} = \arg\max p\ (t_{1:N} \mid x) \qquad (10)$$

Therefore, the log likelihood function becomes:

$$P\ (t_{1:N} \mid x) = \sum_{k=1}^{k} p(p_k \mid d_k\ (X)) \qquad (11)$$

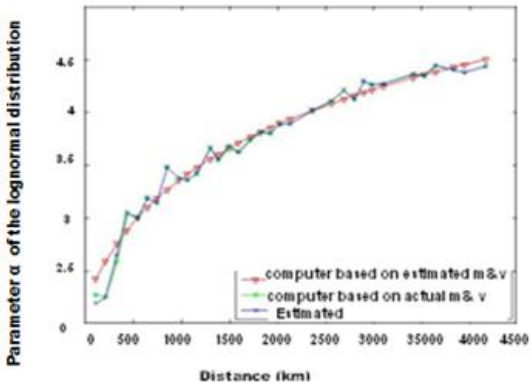

Fig 4(a)

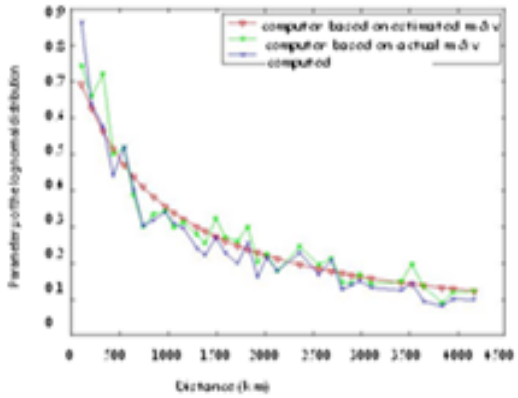

Fig 4(b)

Fig 4 the comparison of estimated and computed values of lognormal parameters (a) μ (b) σ

Where $d_k(x)$ is the geographical distance between the land-mark corresponding to the k_{th} measurement and the potential target location x.

More conveniently, the MLE can be obtained by maximizing the logarithm of the likelihood function as below.

$$\hat{x}_K^{ML} = \arg\max p\ (t_{1:N} \mid x) \qquad (10)$$

Therefore, the log likelihood function becomes:

$$P\ (t_{1:N} \mid x) = \sum_{k=1}^{k} p(p_k \mid d_k\ (X)) \qquad (11)$$

We obtained the MLE by carrying out a search in the 2D search space of (lat_O lon_O) covering the area of interest. In order to provide quick geolocation outcome the search is carried out in multiple steps, where a large area is first searched at a coarse resolution to find the most likely subarea and only this subarea is then searched using a finer resolution search. Further optimization is possible, however, since our goal is to check the applicability of the MLE technique for Geolocation, we consider optimization as not within the scope of this paper. In addition to the ¯first four characteristics of the distance to latency relationship identified in section the GeoWeight algorithm takes into consideration the fifth characteristic of the distance-latency relation-ship; for a given latency, some distances are more probable than other distances. The GeoWeight algorithm uses this characteristic to constrain the possible region of the target to a smaller region than that was possible in the previous approaches as we describe below.

D. GeoWeight Approach:

The GeoWeight algorithm takes into consideration the ¯fth characteristic of the distance-latency relation-ship; for a given latency, some distances are more probable than other distances. The GeoWeight algorithm uses this characteristic to constrain the possible region of the target to a smaller region than that was possible in the previous approaches as we describe below.

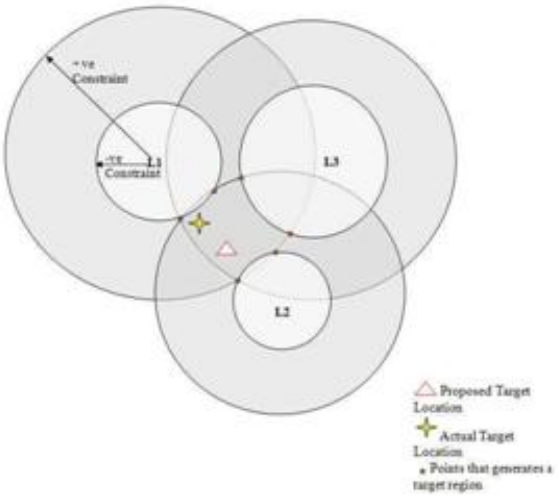

Figure 5: Octant Example

Let t_x be an observed latency from an arbitrary landmark. Based on the distance-latency relationship let d_x^{min} and d_x^{max} be the minimum and maximum possible distances for t_x. Consider the distance range from d_x^{min} to d_x^{max} is divided to $N_{x,d}$ number of equal sized distance bins. The *j*-th bin, (j = 1, 2,.... $N_{x,d}$), covers the distance range from $d_{x,j}^{min}$ to $d_{x,j}^{max}$ given by,

$$d_{x,j}^{min} = d_x^{min} + (j-1)(d_x^{max} - d_x^{min})/N_{x,d}$$

$$d_{x,j}^{max} = d_x^{min} + j(d_x^{max} - d_x^{min})/N_{x,d}$$

Let $w_{x,j}$ be the weight for the *j*-th distance bin corresponding to t_x. The weight $w_{x,j}$ represents the probability of the distance being in the range $d_{x,j}^{min}$ and $d_{x,j}^{max}$ for the observed latency t_x.

For a given latency t_x, the GeoWeight algorithm considers $N_{x,d}$ number of regions around the landmark, with the *j*-th region having distance bounds [$d_{x,j}^{min}$, $d_{x,j}^{max}$] and a weight of $w_{x,j}$ The latency measurements from multiple landmarks will result in intersecting regions, with different numbers of overlap-ping regions in each intersection region. The final weight for each intersection region is computed as the sum of weights of intersecting regions. The region of highest weight is considered as the constrained region of the target and the centred of the region is estimated as the target location. We simplify the generic algorithm presented above by considering the minimum distance, maximum distance and the number of distance bins to be the same for any latency, i.e.,

$$d_x^{min} = D_{min}$$
$$d_x^{max} = D_{max}$$
$$N_{x,d} = N_d$$

where N_d is the number of distance bins for any latency under consideration and D_{min} and D_{max} are the minimum and maximum possible distances for the geolocation scenario. For example, in section 5, where we evaluate our algorithm, the considered

Table 1: An example weight table for GeoWeight

Ping time	0-250kms	250-500kms	500-750kms	750-1000kms
100	**0**	**0**	**0.3**	**0.4**
35	**0.2**	**0.6**	**0.2**	**0**
15	**0.7**	**0.2**	**0.1**	**0**

D_{min} and D_{max} for the geolocation scenario are set such that it covers the whole of the North-American region. The only implication of the above simplification is some distance bins having a weight of 0 due to these distance ranges not being probable for the particular latency.

Figure 6 illustrates an example of the GeoWeight algorithm for a geolocation scenario with latency measurements from three landmarks. In this example the observed latency measures from the three landmarks L1, L2 and L3 are 100, 35 and 15 ms respectively. Table 1 shows the computed weights for the three latencies for different distance regions considering four (N_d = 4) equidistant bins in the distance range 0-1000 km (in this example, D_{min} = 0, D_{max} = 1000) Figure 6 shows the four regions (N_d = 4) with different weights around each landmark in form of circles. These circles overlap with each other and the weight of each intersection region is computed as the sum of weights of the overlapping circles in the inter-section region. For clarity, figure 6 does not show weights of all regions.

In this example, the region of maximum weight has a weight of 2.1 (0.8 + 0.6 + 0.7) and the final target location is selected at the centroid of this region as shown in figure 5.

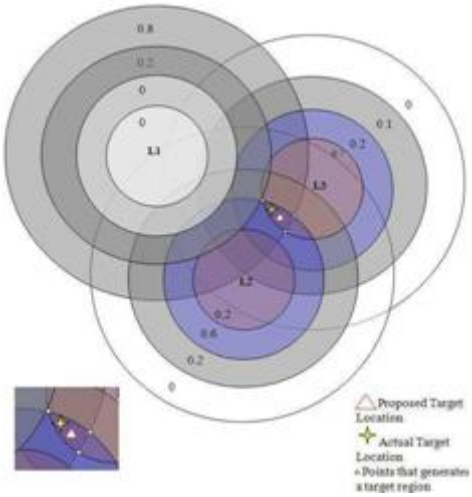

Figure 6: GeoWeight Example

A. Weight Computation

This section describes how the weights for the different distance regions are computed. As mentioned before, the weight for a distance range for a given latency is the probability of the distance being in the range for the given latency. Consider a latency vs. distance data set gathered from Internet measurements covering a distance range D_{min} to D_{max} . Let T_{min} and T_{max} be the minimum and maximum observed latencies. Consider this distance range and the time range divided to N_d and N_t equidistant bins respectively

Table 2: Weight computation example: the number of observed measurements for different time and distance ranges is shown in the table

	(0-500)Kms	(500-1000)kms	(1000-15000)kms	(1500-2000)kms
(0-10)ms	**100**	**100**	**0**	**0**
(10-20)ms	**15**	**15**	**35**	**0**
(20-30)ms	**12**	**12**	**95**	**12**
(30-40)ms	**5**	**5**	**126**	**32**
(40-50)ms	**2**	**2**	**24**	**68**
(50-60)ms	**0**	**0**	**12**	**128**
(60-70)ms	**0**	**0**	**21**	**45**
Total	**134**	**134**	**313**	**285**

Let c_{ij} be the number of data point corresponding to the *i*-th time and *j*-th distance bin.

Table 2 shows an example of delay and distance bins. In this example, T_{min} = 0, T_{max} = 70, D_{min} = 0, D_{max} = 2000, N_d = 4 and N_t = 7.

Each row represents the distance bins corresponding to a single time bin. Each column represents the time bins corresponding to a single distance bin. Each cell of the table shows the number of data points within a given time-distance bin. Since the latency measures are collected for specific distances (i.e. inter landmark distances), even if the same number of latency measurements is gathered for each distance, the total number of distances represented in each distance bin will vary between distance bins as shown in table 2. Therefore, the first step in computing the weights is the normalization across distance bins by dividing the number in each distance-latency cell by the total number of measurements for the particular distance bin. The normalized distance-latency $NR_{i,j}$ are given by,

$$NR_{i,j} = c_{i,j} \,/\, \sum_{i=1}^{N_t} c_{i,j} \qquad (4)$$

The final weight for each cells is computed by normalizing across the latency bin, given by,

$$w_{i,j} = NR_{i,j} / \sum_{j=1}^{N_d} NR_{i,j} \qquad (5)$$

Where $w_{i,j}$ is the weight of *ith* latency bin of j^{th} distance bin which is the probability of the distance region $[d_j^{min}, d_j^{max}]$ given the observed latency is in the range $[T_j^{min}, T_j^{max}]$.

V. EVALUATION

This section presents the results of the experiments we conducted to evaluate our algorithm. We evaluated GeoWeight and MLE using simulated data as well as real Internet data. We also compared GeoWeight with two existing techniques; Octant and CBG.MLE technique by geolocating all of our targets.

We conducted experiments using latency measurements simulated based on the latency model presented in section 3 using two different probability models for noise. The aim of these experiments was to:

- Determine the optimum values of N_d and N_t for the algorithm.
- Evaluate the algorithm for known noise models

Figure 8 shows the cumulative geolocation error distribution of Octant and CBG obtained from this service in dash and dash-dotted line respectively. We geolocated each target three times and the geolocation error shown in figure 8 is the minimum of the three values. This geolocation service uses supplementary constraints such as geographical and demographical hints in addition to latency measurements to refine its estimates whereas our implementation of Octant does not use any such optimization technique. We believe that this is the reason for the observed differences of geolocation error between our implementation of Octant and the implementation provided by the authors. Extra geographical and demographical hints significantly improved Octant's accuracy and this observation is aligned with the description in [3].The proceeding evaluation examined the accuracy of the MLE-based geolocation approach for different number of landmarks. Figure 9 shows the cumulative geolocation error distribution of MLE with 50 and 30 landmarks. The figure shows, accuracy of our approach is relatively independent of number of landmarks or slightly improves with higher number of landmarks. In our experiments one global probability distribution is generated to map distance to latency from the dataset shown in figure 2(a). Mapping distance to latency based on per landmark could leverage the distribution better. However, it requires a rich dataset in order to draw a credible statistical conclusion for a probability based approach. In order to generate such dataset based on per landmark to represent the whole experimental region of North-America.

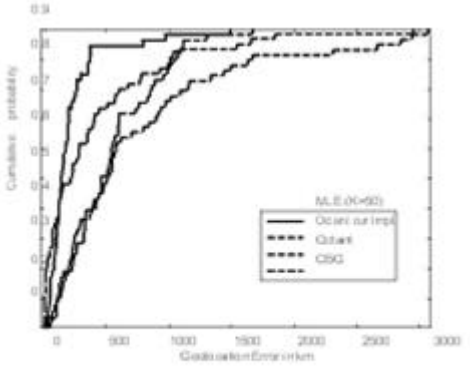

Fig. 8. Comparison of the accuracy of MLE, Octant and CBG

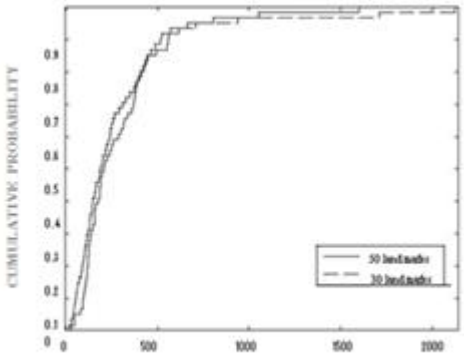

Fig 9 cumulative distribution of geolocation error (€) of MLE based approach of 50 and 30 landmarks

VI. CONCLUSION

This paper investigated the problem of Internet host Geolocation using the maximum likelihood estimation technique and GeoWeight**.** The likelihood function was derived empirically from latency measurements and a probability model for Internet latency computed from observed latencies for different distances. We validated GeoWeight with simulated and real Internet data and we collected over a period of time using 50 hosts on the Planet Lab test bed. The conditional probability of latency for a given distance was approximated by a lognormal distribution to simplify the analysis in this initial work. Using simulated latency measurements, generated based on a lognormal distribution. we tested the theoretical limit of our approach. Geolocation of 50 real targets in the Planet Lab resulted in a median Geolocation error of 134 km.In comparisons to two well known measurement-based approaches for Internet host Geolocation, Octant, CBG and MLE-based approach was found to perform better. It is to be noted that our approach of testing the applicability of MLE for the Geolocation problem of Internet host can be seen more as a statistical model to the problem than as a complete framework such as Octant and CBG. However, it is encouraging that usage of MLE was able to produce good Geolocation outcomes for real targets even with a simple approximation. we are investigating better probability distributions for Internet latency data which can capture the behaviour of noise in latency on a per landmark basis. This outcome establishes the intuition of proposing complete Geolocation framework in future. Future work will focus on a more accurate probability model for latency measurements and MLE approach can be evaluated using a different test bed other than Planet Lab. One initial investigation have focused on the Levy distribution which have shown promising results.

REFERENCES

[1] B.A.Forouzan. *Data Communications and Networking*. Tata McGraw-Hill Publishing Company Limited, 2000.

[2] Basset, E., John, P., Anderson, T., Krishnamurthy, A., Chawathe, Y., and Wetherall, D. towards IP Gelolocation Using Delay and Topology Measurements. *In the Proceedings of IMC 06.* (2006).

[3] C.Guo, Y.Liu, W. H. Q. Y. Mining the Web and the Internet for Accurate IP Address Geolocations. *In Proceedings of INFOCOM 2009. The 28th Conference on Computer Communications. IEEE* (2009).

[4] V. N. Padmanabhan and L. Subramanian., "An investigation of geographic mapping techniques for internet hosts." *In the Proceedings of ACM SIGCOMM Computer Communication Review*, 2001.

[5] Dabek, F., Cox, R., Kaashoek, F., and Morris, R. Vivaldi: A Decentralized Network Coordinate System. *In the*

proceedings of the SIGCOMM 2004 (2004).

[6] E.Limpert, W.A.Stahel, M. Log-normal Distributions across the Sciences: Keys and Clues. *In the Proceedings of Bioscience, Vol. 51 No. 5 pp. 341-352.* (2001).

[7] V. N. Padmanabhan and L. Subramanian., "An investigation of geo-graphic mapping techniques for internet hosts." *In the Proceedings of ACM SIGCOMM Computer Communication Review*, 2001..

[8] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-Based Geolocation of Internet Hosts." *Networking, IEEE/ACM Transactions*, vol. 14, no. 6, pp. 1219–1232, 2006.

[9] B. Wong, I. Stoyanov, and E. Sirer, "Octant: A Comprehensive Frame-work for the Geolocalization of Internet Hosts." *In Proceedings of Sym-posium on Networked System Design and Implementation, Cambridge, Massachusetts*, 2007.

[10] Basset, P. John, T. Anderson, A. Krisnamurthy, Y. Chawathe, and D. Wetherall, "Towards IP Gelolocation Using Delay and Topology Measurements." *In the Proceedings of IMC 06.*, 2006.]

[11] B. Wong, I. Stoyanov, and E. Sirer, "Octant: A Comprehensive Framework for the Geolocalization of Internet Hosts." In Proceedings of Symposium on Networked System Design and Implementation, Cambridge, Massachusetts, 2 007.

[12] D. I.Youn, B.L.Mark, "Statistical Geolocation of Internet Hosts." *In Proceedings of 18th International Conference on Computer Communications and Networks, 2009*.

[13] M. E.Limpert, W.A.Stahel, "Log-normal Distributions across the Sciences: Keys and CLues." *In the Proceedings BioScience, Vol. 51 No. 5 pp. 341-352.*, 2001.

[14] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I*.J Wiley, New York, 1968.

[15] M. J. Arif, S. Karunasekera, S. Kulkarni Internet Host Geolocation Based on a Probability ModelforLatency Measurements CRPIT Volume 102 - Computer Science 2010.

[16] M. J. Arif, S. Karunasekera, S. Kulkarni, A. Gunatilaka, B. Ristic Internet Host Geolocation using Maximum Likelihood Estimation Technique *18th International Conference on Computer Communications and Networks, 2009*.

