Posit: A Lightweight Approach for IP Geolocation

Brian Eriksson
Boston University
and UW-Madison
eriksson@cs.bu.edu

Paul Barford
UW-Madison
pb@cs.wisc.edu

Bruce Maggs
Duke University
and Akamai
Technologies
bmm@cs.duke.edu

Robert Nowak
UW-Madison
nowak@ece.wisc.edu

ABSTRACT

Location-specific Internet services are predicated on the ability to identify the geographic position of IP hosts accurately. Fundamental to current state-of-the-art geolocation techniques is reliance on heavyweight `traceroute`-like probes that put a significant traffic load on networks. In this paper, we introduce a new lightweight approach to IP geolocation that we call *Posit*. This methodology requires only a small number of delay measurements conducted to end host targets in conjunction with a computationally-efficient statistical embedding technique. We demonstrate that Posit performs better than all existing geolocation tools across a wide spectrum of measurement infrastructures with varying geographic densities. Specifically, Posit is shown to geolocate hosts with median error improvements of over 55% with respect to all current measurement-based IP geolocation methodologies.

1. INTRODUCTION

Characterization of the Internet can be performed with respect to the router-level topology (*e.g.,* [1]), autonomous system-level topology (*e.g.,* [2]), end-to-end paths (*e.g.,* [3]), and latencies between hosts (*e.g.,* [4]). One characteristic of Internet structure that has significant implications for advertisers, application developers, network operators, and network security analysts is to identify the geographic location, or *geolocation*, of networked devices, such as routers or end hosts.

The ultimate goal of IP geolocation is to find the precise latitude/longitude coordinates of a target Internet device. There are considerable challenges in finding the geographic location of a given end host in the Internet. First, the size and complexity of the Internet today, coupled with its highly diffuse ownership, means that there is no single authority with this information. Second, no standard protocol provides the geographic position of an Internet device on the globe (although domain names can include a location record). Third, non-mobile Internet devices are not typically equipped with location identification capability (*e.g.,* GPS or other satellite-based techniques [5]), although this may change in the future. However, even these equipped mobile devices may choose not to report location information due to privacy concerns.

IP geolocation methods that are currently used largely fall into two categories. The first is database-specific approaches in which a geolocation database is established by examining network address space allocations and user-entered geographical data. While this can be effective for providers who offer service in a restricted geographic region (*e.g.,* a university or a small town), it will fail for providers with a large geographic footprint unless coupled with additional information (as found in [6, 7]). The second method is to use active probe-based measurements to place the target host within some specified geographic region. The accuracy of probe-based techniques is often dependent on the geographic proximity of target hosts and the measurement infrastructure. The result is geolocation estimates with relatively high median error and high error variability when measurement resources are geographically distant to a target host.

This paper proposes a novel approach to IP geolocation that we call *Posit*. The Posit methodology considers three categories of devices in the network: *Targets* - Hosts with unknown geographic location that we aim to geolocate and respond to probes; *Landmarks* - Infrastructure in the network with known and very accurate geolocation information (sometimes referred to as "passive landmarks" in prior literature); and *Monitors* - Network resources with known geographic location and the ability to send `ping` measurements to both landmarks and targets (sometimes referred to as "active landmarks" in prior literature).

The first goal of our work is to develop a measurement-based IP geolocation framework that provides accurate estimates and reduces estimation error over prior methods. The second goal of our work is to develop a geolocation framework that is lightweight in the sense that it only relies on a small number of low network load measurements in order to establish location estimates. We achieve both of these goals through a statistical framework that extracts geographic distance information between the targets and landmarks using *both* short and long observed latency measurements from the set of monitors.

We examine and validate our Posit geolocation framework using two data sets. Our geolocation targets and monitors are drawn from 431 commercial end hosts, and passive landmarks from 283 domain names with LOC records that have been validated. It is important to note that the exact coordinates of *all* hosts used in this study were known, which gave us a strong foundation for evaluation. Our results show that Posit provides highly accurate geolocation estimates over a broad spectrum of target/monitor/landmark scenarios. Furthermore, comparisons of Posit estimates versus estimates from other geolocation methods show significant improvements in accuracy. These results highlight the necessity of comparing prior geolocation methods over *the same measurement infrastructure*, since the geographic distribution of the measurement infrastructure relative to the target plays

a central role in the variation of geolocation accuracy for all techniques. Specifically, our results show that simply relying on prior published error metrics will not accurately characterize the true performance of geolocation methodologies.

Across a broad spectrum of geographic densities (in terms of the geographically closest monitor/landmark) on the commercial node data set, Posit returns geolocation estimates with median error of only 27 miles. These data sets include a significant subset of targets that are hundreds of miles away from the nearest measurement infrastructure resource, a critical regime when considering non-US geolocation. In comparison with the best competing prior geolocation techniques on the same measurement data sets, we see median error improvements of over 55%. Across various infrastructure regimes, we find Posit outperforming the prior geolocation methodologies with median geolocation error reductions ranging from 40% to 75% over the best prior methodology. In terms of measurement traffic load, Posit uses the same number of `ping`-like probes as competing latency-based geolocation techniques and does not need heavyweight `traceroute` measurements required by recent state-of-the-art approaches (*e.g.,* [8, 9, 10]).

The paper is organized as follows. In Section 2, we review prior studies that inform our work. The datasets used for the experiments in this paper are described in Section 3. The components of the Posit framework are described in Section 4. Finally, the experimental performance of the Posit methodology is explored in Section 5, with the conclusions of the paper in Section 6.

2. RELATED WORK

Considerable prior work has been performed on the subject of IP geolocation (*e.g.,* [8, 9, 10, 11, 12]). We describe the details of these prior methods in Section 5. While we are informed by this work and our motivation for highly accurate estimates is the same, the methodology described in this paper makes new contributions that reduce measurement load on the network and improve estimate accuracy over prior geolocation algorithms. Unlike the Street-Level [8], Octant [9], and Topology-based [10] methodologies, no `traceroute` probes are necessary to use Posit. In addition to significantly decreasing the network load, this avoids the well known problems of interface disambiguation (*e.g.,* [1, 13]) and dependency on unreliable unDNS naming conventions (*e.g.,* [14]).

All previous IP geolocation algorithms use latency as a proxy for distance measurements. While some algorithms use latency measurements solely as an upper bound constraint on possible geographic locations (*e.g.,* [12]), others have tried to directly estimate distance from the latency values (*e.g.,* the spline-based method of [9]). More recent work (*e.g.,* [15, 16, 17]) has used estimation of distance likelihood probability given observed latency. Often, to exploit a landmark with known location, many of these prior methods would require observation of latency between the two resources. Unfortunately, due to the lack of control over either the target or landmarks, direct latency measurements will not be observed. Instead, via inference, the Posit methodology transforms `ping`-like measurements from a set of monitors to estimate the probability of distance between targets and landmarks. This eliminates the need for any direct measurements between the two sets of Internet resources, and allows Posit to exploit location information from a large set of Internet infrastructure using low network load. In addition, while the utility of short latency values between targets and monitors has been previously explored in [10], using a novel distance metric we find the utility of long observed latency measurements for the use of improving geolocation accuracy.

Similar to prior geolocation techniques (*e.g.,* [8, 11]), the Posit framework relies on the procurement of a large set of *passive landmarks* – Internet hosts with known latitude/longitude coordinates that respond to measurement probes. Posit exploits the large collection of existing Internet infrastructure with publicly available geolocation information. Landmarks used by Posit in this paper are hosts with domain names that include location information (via careful analysis of DNS LOC records, using [18]). To the best of our knowledge, this is the first time that DNS has been used for IP geolocation. Posit does not rely exclusively on this resource, and in future empirical studies we expect to add other nodes to our landmark database (such as a set of stratum 0/1 Network Time Protocol (NTP) servers [19]). Indeed, the novel method for identifying additional landmarks described in [8] could be used in parallel with Posit to increase the measurement node density, but we look to that task for future work.

3. DATA SETS

To evaluate Posit, we use a set of measurements collected from 431 commercial hosts with known latitude/longitude coordinates that belong to Akamai Technologies in North America. During the weekend of January 16-17, 2010, pairwise bidirectional hop count and latency measurements were conducted on those nodes. The measurements were collected using standard ICMP ECHO requests via the MTR tool [20]. The servers are used in Akamai's production CDN so during the measurement period, they may have been performing other tasks (such as serving HTTP content), which could have had an effect on latency measurements. In addition, we only consider a single latency measurement between each commercial host in order to minimize network load, which may introduce additional inaccuracies due to queuing delay. These inaccuracies will result in a more faithful representation of time-limited real-world geolocation performance (*e.g.* estimating geolocation concurrently as a web page loads) than studies that pre-process or obtain multiple latency measurements for each target-monitor pair.

We also consider a set of 283 hosts with domain names that include valid DNS LOC records in the continental United States. The locations obtained by the DNS LOC records were verified using both commercial IP geolocation databases and verification that no resource violated latency speed-of-light constraints. Standard ICMP ECHO requests via the MTR tool were performed on January 22, 2011 from the set of 431 commercial hosts to all 283 domain names.[1] While the size of our validation set may appear small compared with prior work (*e.g.,* [8, 15]), in contrast to these larger prior studies we have *ground truth location knowledge of all hosts under consideration.*

[1]The authors would like to thank Rick Weber and KC Ng from Akamai Technologies for supplying us this data.

4. POSIT GEOLOCATION METHODOLOGY

Consider observed latency measurements between a target end host and a set of monitors. From latency measurements, we construct a set of *latency vectors*. For target $i = \{1, 2, ..., N\}$,

$$\mathbf{l}_i^{target} = \left[\begin{array}{cccc} l_{i,1}^{target} & l_{i,2}^{target} & ... & l_{i,M}^{target} \end{array} \right]$$

Where $l_{i,k}^{target}$ is the observed round-trip time between target i and monitor k.

Additionally, consider latency measurements to each landmark $j = \{1, 2, ..., T\}$,

$$\mathbf{l}_j^{land} = \left[\begin{array}{cccc} l_{j,1}^{land} & l_{j,2}^{land} & ... & l_{j,M}^{land} \end{array} \right]$$

Where $l_{j,k}^{land}$ is the observed round-trip time between landmark j and monitor k.

The Posit algorithm will estimate the geographic location of each target end host using only these observed latency measurements vectors from a set of monitors.

4.1 Landmark-to-Target Distance Likelihood Estimation

Consider estimating the geographic distance between a target end host and a single landmark, where we know the geolocation of the landmark but the target location is unknown. Given that we do not control either the target or the landmark, we cannot obtain Internet measurements between the two resources (*e.g.,* latency, hop count, etc.). Instead, we examine the observed latency vectors for a target end host and a landmark, $\mathbf{l}_i^{target}$ and $\mathbf{l}_j^{land}$, respectively. Even without direct measurements between the landmark and target, characteristics of geographic distance can be revealed from these two vectors.

We start by examining various distance metrics between the two latency vectors. Prior work in [21] has shown that by weighting short latency values using the Canberra distance, $d_{i,j}^{canberra} = \sum_{k=1}^{M} \frac{\left|\mathbf{l}_{j,k}^{land} - \mathbf{l}_{i,k}^{target}\right|}{\mathbf{l}_{j,k}^{land} + \mathbf{l}_{i,k}^{target}}$, is a metric that better correlates with geographic distance between the two resources, in contrast with taking the Euclidean norm, $\|\mathbf{l}_j^{land} - \mathbf{l}_i^{target}\|_2$. Further extending this idea, we introduce the concept of thresholding the latency values to only consider the set of short latency indices, $\mathcal{I}_{i,j}$, the indices of the two vectors where at least one of the values is below some specified delay threshold, $\lambda_{lat} > 0$,

$$\mathcal{I}_{i,j} = \{k : \; l_{i,k}^{target} \leq \lambda_{lat} \quad \text{or} \quad l_{j,k}^{land} \leq \lambda_{lat} \; \} \tag{1}$$

The intuition behind this thresholding technique is as follows. Consider a monitor with observed latency to a target and a landmark. If the monitor-target latency is small, this implies that the monitor and target are geographically close (due to speed-of-light constraints). Then if the monitor-landmark latency is also small, this implies that the landmark and target are likely geographically close. If the landmark latency is large, this implies that the landmark is potentially geographically distant from the monitor, and therefore potentially geographically distant from the target. Finally, if both the target and landmark have large latency from the monitor, this is uninformative, as the landmark-target could be either geographically close or distant. Therefore, a particular monitor is informative in terms of target-landmark geographic distance if at least one of the latency values is small. This allows for large latency monitor-target observations to reveal geographic information, as long as monitor-landmark latency (with respect to the same monitor) is short.

We choose the distance transformation for the short latency elements (*i.e.,* the set of monitors indexed by $\mathcal{I}_{i,j}$), such that the observed latency-based distance metric and geographic distance has the closest linear relationship. First, we use the L1-norm for the short latency indices (*i.e.,* the *threshold L1-norm distance*) :

$$v_{i,j}^{L1} = \frac{1}{|\mathcal{I}_{i,j}|} \|\mathbf{l}_i^{target}(\mathcal{I}_{i,j}) - \mathbf{l}_j^{land}(\mathcal{I}_{i,j})\|_1 \tag{2}$$

Where $\mathbf{l}(\mathcal{I}) = \left[\begin{array}{cccc} l_{\mathcal{I}_1} & l_{\mathcal{I}_2} & ... & l_{\mathcal{I}_{|\mathcal{I}|}} \end{array} \right]$ is the subvector of latency with respect to indices $\mathcal{I}$. We also define the *threshold L2 distance* and the *threshold Canberra distance*, where the L1-norm in Equation 2 is replaced with the L2-norm and the Canberra distance, respectively.

To evaluate how well these various latency distance metrics correlate with geographic distance, we use the R^2 *coefficient of determination* metric [22], which measures the quality of the linear relationship between the geographic distance and the distance metric value. By definition, $R^2 = 1$ if there is a perfect linear trend between the geographic distance values and latency distance metric values, and $R^2 = 0$ if the two sets of values are uncorrelated (*i.e.,* no linear trend). Given our commercial node data set and latency threshold of $\lambda_{lat} = 10$, in Figure 1 we show the coefficient of determination, R^2, for the six latency-based distance metrics (from latency vectors to a set of monitors) and for direct observation of pairwise latency between the targets and landmarks. This is shown as the number of monitors under consideration varies and with respect to the true geographic distance between the targets and landmarks.

The figure shows over 60% of the geographic distance information can be explained by the threshold L1-norm distance, a better linear fit compared with all other distance metrics and even direct latency measurements between the target and landmark. This can be explained, as the L1-norm will weight the more important, smaller latency deviations more than either the L2-norm or Canberra distance metrics. The strong linear trend between the threshold L1-norm values and the true geographic distance indicates the potential to obtain accurate geographic distance estimates between targets and landmarks without the need for direct measurements.

While the threshold L1-norm results indicate correlation between our latency vector distance metric and geographic distance, this will return only a single distance estimate between the known landmark location and the unknown target location. To increase accuracy, we consider learning distance likelihood distributions, $\widehat{p}_{land}(d \mid \mathbf{v})$, the probability of a target being d distance away from a landmark given threshold L1-norm values that lie in the range of $\mathbf{v}$. In order to learn these distribution functions, we exploit a small set of training targets with known measurements and geographic locations, and an off-the-shelf kernel density estimator technique [23]. An example of these estimated distributions from a training set of threshold L1-norm values between target/landmark pairs with known pairwise geographic distance information can be found in Figure 2.

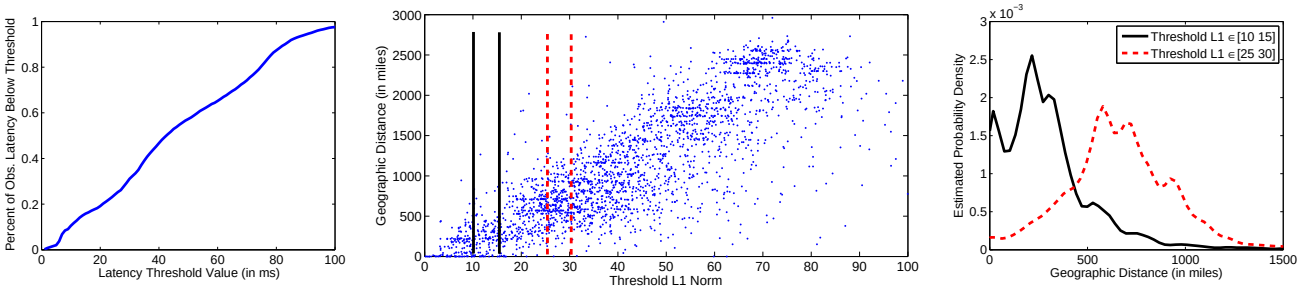

Figure 2: (Left) - Percentage of observed latency values below specified delay threshold. (Center) - An example scatter plot of calculated threshold L1-norm distance values and the ground truth geographic distance (where each point represents a different target and landmark pair) using the Commercial node data set and latency measurements from 50 monitors. (Right) - Kernel density estimates of distance for $\widehat{p}_{land}(d \mid \mathbf{v}_A)$ with threshold L1-norm distance values, $\mathbf{v}_A \in [10, 15]$, and $\widehat{p}_{land}(d \mid \mathbf{v}_B)$ with threshold L1-norm distance values, $\mathbf{v}_B \in [25, 30]$.

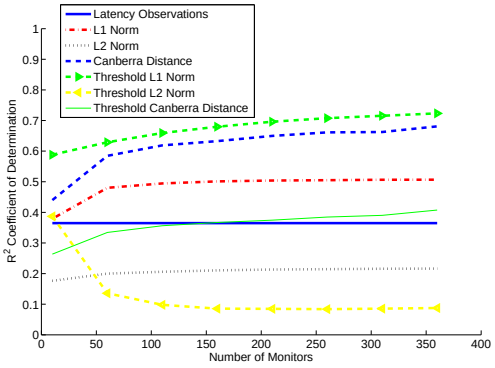

Figure 1: Coefficient of Determination (R^2) - Measure of linear fit quality for various latency-based distance metrics (and observed latency) compared with true geographic distance.

4.2 Statistical Embedding Algorithm

Given the estimated distance likelihood probability for the target to each of the landmarks, $\widehat{p}_{land}(d \mid v_{i,j})$, our goal is to estimate each target's latitude/longitude coordinates. In addition to our estimated likelihood distributions between landmarks and targets, there is additional information we exploit (*i.e.,* the latency observations between the monitors and targets). Similar to recent statistical geolocation methodologies (*e.g.,* [15, 16, 17]), we construct distance likelihood probabilities from observed latencies to the monitors, $\widehat{p}_{monitor}(d \mid l_{i,j})$ (*i.e.,* the probability of being d miles away from monitor j given observed latency $l_{i,j}$), using a training set of targets with known location. Additionally, using Constraint-based Geolocation [12] we obtain the constraint region, $\mathbf{C}_i$, the set of feasible latitude/longitude coordinates given our observed latency values from the monitors.

One of the aspects where our approach differs from prior research on constraint-based and probability likelihood methods is that we assume that the resulting embedding coordinates of the set of targets should be *sparse*. This is the case where we confine geolocation to a small subset of locations (*e.g.,* cities) where we expect the target to be placed. In contrast to previous work (*e.g.,* [9, 15]), which requires explicitly defined population and/or geographic data as input into the algorithm, Posit does not require a priori knowledge of the population density or geographic properties of the region of interest. Instead, to enforce this restriction, we only consider the known locations of landmarks and monitors in the infrastructure that are contained in the geographic constraint region ($\mathbf{C}_i$), areas where we expect to find high population/Internet resource density (*e.g.,* [24]).

Therefore, given the set of feasible latitude/longitude coordinates in the constraint region $\mathbf{C}_i$ found by Constraint-based geolocation, we define the set of constrained resource coordinates, $\mathbf{C}_{\mathbf{R}i}$, as the coordinates of the landmarks ($\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_T\}$) and monitors ($\mathbf{M} = \{\mathbf{m}_1, \mathbf{m}_2, ..., \mathbf{m}_M\}$) found in the constraint region. If none of the monitors or landmarks are found in the region determined by Constraint-based geolocation, we geolocate with respect to the entire constraint region ($\mathbf{C}_i$) by finding the most likely geographic location inside the region.

We aim to find the most probable constrained resource location given our set of trained likelihood distributions and observed measurements to the monitors. In contrast with prior approaches [15, 16, 17], here our sparse approach will only use geographic coordinates of known Internet resources (*i.e.,* locations where we have knowledge of Internet deployment and/or population) and the addition of geographic information from passive landmarks via our threshold L1-norm distances (*i.e.,* the value $v_{i,j}$ for target i and landmark j). This statistical embedding algorithm finds the estimated geographic location coordinates, $\widehat{\mathbf{x}}_i$, by maximizing the log-likelihood given measurements from the monitors to both the target and landmarks.

$$\widehat{\mathbf{x}}_i = \underset{\mathbf{x} \in \mathbf{C}_{\mathbf{R}i}}{\operatorname{argmax}} L_i(\mathbf{x})$$

Where the log-likelihood for target i,

$$L_i(\mathbf{x}) = \left(\sum_{j=1}^{T} \log\left(\widehat{p}_{land}\left(d(\mathbf{x}, \mathbf{t}_j) \mid v_{i,j}\right)\right) + \sum_{k=1}^{M} \log\left(\widehat{p}_{monitor}\left(d(\mathbf{x}, \mathbf{m}_k) \mid l_{i,k}\right)\right) \right) \quad (3)$$

With $d(\mathbf{x}, \mathbf{y})$, the geographic distance between lati-

tude/longitude coordinates $\mathbf{x}$ and $\mathbf{y}$, and the set of monitors and landmarks that lie in the feasible constrain region, $\mathbf{C_R}_i$.

An example of this statistical embedding methodology can be found in Figure 3. This computationally lightweight methodology only requires $O(|\mathbf{C_R}|(T+M))$ computational operations to geolocate each target, as our sparse approach considerably reduces the number of feasible geographic coordinates to optimize over.

4.3 Posit Geolocation Algorithm Summary

We now summarize the Posit geolocation algorithm. By exploiting the distance likelihood distributions (described in Section 4.1), we use the statistical embedding algorithm (described in Section 4.2) to estimate geographic location for our set of test targets. The tuning parameter (λ_{lat}) is found through an efficient bisection search using the small training set consisting of targets with known locations. To prevent overfitting, the best threshold parameter value is found with respect to the training set and that same threshold value is used with respect to every target in the test set. After careful inspection of our data, the likelihood distributions are constructed from the training set for landmark-based distance likelihoods ($\widehat{p}_{land}(d \mid v)$) using the threshold L1-norm distance ranges, $\{(0,5],(5,10],...,(75,80]\}$, and the monitor-based distance likelihoods ($\widehat{p}_{monitor}(d \mid l)$) are constructed for observed latency ranges $\{(0,10)],(10,20],...,(140,150]\}$ (in milliseconds). The complete Posit geolocation methodology is presented in Algorithm 1.

Algorithm 1 - Posit Geolocation Algorithm

Given:

- Latency vectors from the M monitors to our set of N targets, $\mathbf{l}_i^{target}$ for $i = \{1,2,...,N\}$.
- Latency vectors from the M monitors to our set of T landmarks, $\mathbf{l}_j^{land}$ for $j = \{1,2,...,T\}$.
- Training set of targets with known geolocation and latency measurements to the monitors.

Initialize:

- Learn the likelihood distributions, $\widehat{p}_{land}(d \mid v)$ and $\widehat{p}_{monitor}(d \mid l)$ using the training set with known target locations.
- Use the training set to find the value of tuning parameter (λ_{lat}) that minimizes the training set geolocation error rate.

Methodology:
For each target, $i = \{1,2,...,N\}$

- Resolve the threshold L1-norm distances, $v_{i,k}$ for $k = \{1,2,...,T\}$ using Equation 2.
- Using the learned distributions ($\widehat{p}_{land}(d \mid v)$ and $\widehat{p}_{monitor}(d \mid l)$), use the Statistical Embedding methodology (Equation 3) to estimate the target geolocation.

5. EXPERIMENTS

Using both the commercial node data set and domain name data set described in Section 3, we evaluate the performance of the Posit algorithm.

5.1 Comparison Methodologies

To evaluate relative performance of the Posit algorithm, we compare against numerous prior geolocation methodologies[2].

5.1.1 GeoPing and Shortest Ping Algorithms

Some of the first IP geolocation methodologies developed were the Shortest Ping and GeoPing techniques. The *Shortest Ping* technique [10] uses a series of latency measurements from a set of monitors to a target, and then maps that target's geolocation to the location of the monitor with the shortest observed latency value. We expect Shortest Ping to work well in our evaluation for instances where the monitor placement is dense. However, in instances where monitors are not near targets, the Shortest Ping methodology's accuracy should decline and the strength of our Posit methodology will be highlighted.

The *GeoPing* algorithm [11] was the first IP geolocation algorithm proposed that exploited existing Internet infrastructure with known location (*i.e.,* landmarks). Using latency measurements from a set of monitors, the target latency vector is compared with the latency vectors from each of the landmarks to the monitors. The geolocation of the target is the location of the landmark with the smallest Euclidean distance (*i.e.,* L2-norm) between latency vectors. This methodology's accuracy is strongly dependent on the location of the landmarks with respect to the target.

5.1.2 Constraint-Based Geolocation

Using the algorithm described in [12], we implemented the *Constraint-Based Geolocation* (CBG) approach. Using only `ping`-based measurements, the basic intuition behind CBG is that the latency measurements to a set of monitors with known location can be considered a series of geographic constraints. Given speed-of-light in fiber assumptions and self-calibration using a set of training data, we determine a feasible geographic region given each latency measurement where the target must be located in. From a series of latency measurements, the possible geographic placement is considered the intersection of many constraint regions, with the estimated location being the centroid of this intersection region. The size of this final constraint region will be correlated with the smallest individual constraint region size, which we expect is dependent on the shortest observed latency (*i.e.,* likely the geographically closest monitor) to the target.

5.1.3 Octant-Based Geolocation

Building on the Constraint-based Geolocation approach is the *Octant* algorithm [9]. Octant uses both `ping`-based measurements to the targets *and* given geographic information from unDNS [14] information from routers along the path to the targets. Our implementation, which we refer to as

[2]The MATLAB code used in this paper for both our Posit methodology and all comparison methods will be made publicly available.

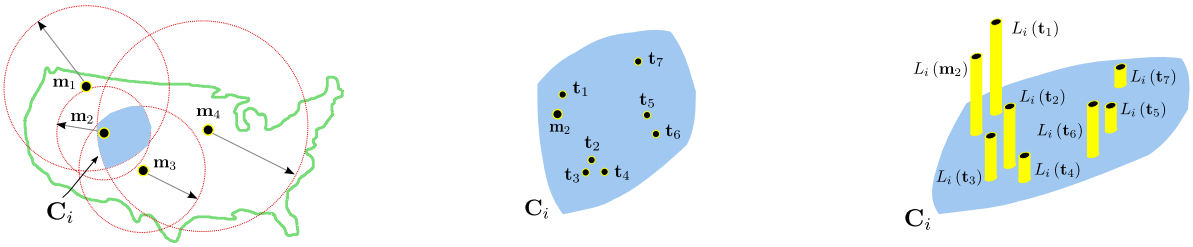

Figure 3: (Left) - Example construction of constrained geographic region $\mathbf{C}_i$ using observed latency from monitors $\{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \mathbf{m}_4\}$ to target i. (Center) - Example set of constrained resources $\mathbf{C}_{\mathbf{R}_i}$, using monitors $(\mathbf{m}_2)$ and landmarks $(\{\mathbf{t}_1, ..., \mathbf{t}_7\})$ located inside the constrained region, $\mathbf{C}_i$. (Right) - The estimated log-likelihood for each constrained resource location using Equation 3, with the estimated geographic location $(\widehat{\mathbf{x}}_i = \mathbf{t}_1)$ associated with the largest calculated log-likelihood value, $L_i(\mathbf{t}_1)$.

Octant-Based Geolocation[3], includes the Octant methodology's use of both "positive" and "negative" geographic constraints from latency measurements, the iterative refinement of the feasible constraint region, unDNS intermediate node information, point selection through Monte Carlo simulation, latency "heights", and spline approximation of latency to distance. Missing from our implementation of Octant is the use of geographic/population information to aid in geolocation, as we feel the lack of process description in [9] could potentially bias our implementation of this component. In our experiments, unDNS-derived geographic information is derived from the last hop router encountered along the path before the target. For our commercial set of 431 nodes, it was found that only 71 nodes had available last hop unDNS information down to the city location. Similar to the CBG approach it is based on, we expect Octant to perform the best when monitors are close to the targets.

5.1.4 Statistical Geolocation

Statistical work in IP geolocation (*e.g.,* [15, 16, 17]) finds the geographic location that maximizes the likelihood probability of geographic location with respect to observed latency measurements. While the construction of the probability distributions varies (nonparametric kernel density estimators in [15, 16], parametric log-normal model in [17]), all three methodologies assume conditional independence between measurements in order to efficiently calculate the geographic location with the maximum likelihood given observed measurements. We compare using the methodology from [15], which relies on a training set of target end hosts with latency measurements and known geolocation to generate kernel density estimates [23] of distance given observed latency.

5.1.5 Shared Path Geolocation

The most recent contributions to geolocation literature focus on using intelligent web-based search to discover a large number of passive landmarks. This approach has been used in the Structon framework [25], the Street-Level geolocation methodology [8], and the Alidade project [26]. While these three projects use a large procurement of passive landmarks as their basis, the underlying geolocation inference mechanisms differ. For example, the Street-Level approach estimates a constraint region from inferred shared path latencies between targets and landmarks via `traceroute` probes, the Structron approach uses clustering from a combination of both IP address subnets and `traceroute` paths, and the Alidade project uses passive landmarks (which they refer to as "beacons") to help localize routers.

For an apples-to-apples comparison with other techniques, we test the `traceroute`-based shared path methodology of the Street-Level geolocation approach on the set of passive landmarks that are used in our evaluation. We call this modified approach, *Shared Path* geolocation. While this measurement methodology is the basis for the Street-Level approach, given our lack of their landmark-discovery data mining infrastructure, we do not claim to be comparing against the specific results of [8]. Given the dependency on CBG of the approach in [8], this methodology should return the most accurate geolocation results when both the monitors and the landmarks are geographically close to the targets.

5.2 Geolocation Probing Complexity

The number of network probes required for each geolocation methodology is seen in Table 1. The most recent geolocation method, Street-Level geolocation, requires a heavyweight `traceroute` probe from every monitor to every target and from every monitor to every landmark, resulting in a very large measurement load on the network. Another methodology that uses data derived from `traceroute` probes is Octant, which resolves router unDNS hints and latency observations to further constrain their target geolocation estimates, requiring at least a single `traceroute` probe to each target. Meanwhile, Posit requires the same number of lightweight latency measurements as the GeoPing methodology, where the only measurements required are latency probes from each monitor to the set of targets and from each monitor to each landmark. In contrast with the recent methods (Street-Level and Octant), Posit requires no `traceroute` measurements.

5.3 Results

Our experiments use the 431 commercial nodes as targets to test the performance of Posit and every competing geolocation algorithm. For each target, we randomly select 25 monitor nodes from the set of 430 remaining commercial nodes. In addition, for each target we randomly select 75 domain names (out of 283 total) with known location as landmarks which aid in geolocation. To assess performance

[3]We were unable to get access to specific Octant code used in [9] to compare it with Posit for this study.

Table 1: Probing complexity for all measurement-based geolocation methodologies (given N targets, M monitors, and T landmarks).

Methodology	`ping`-like Measurements	`traceroute` Measurements
Posit	$O(M(N+T))$	0
Shortest Ping	$O(MN)$	0
GeoPing	$O(M(N+T))$	0
Constraint-Based	$O(MN)$	0
Octant	$O(MN)$	$O(N)$
Shared Path	$O(MN)$	$O(M(N+T))$
Statistical	$O(MN)$	0

Table 2: Geolocation error (in miles) for all geolocation methodologies using number of monitors $M = 25$ and number of landmarks $T = 75$.

Methodology	Mean Error	Median Error	Standard Deviation
Posit	71.38	27.18	102.26
Shortest Ping	172.22	184.80	145.86
GeoPing	241.45	205.55	283.25
Constraint-Based	143.61	116.73	135.02
Octant Based	112.05	64.58	133.73
Shared Path	174.47	160.74	184.22
Statistical	161.06	136.01	141.11

of the Posit, Constraint-Based, Octant, and Statistical geolocation algorithms (all of which require a training set of targets with known geolocation), we perform hold-out Cross Validation [23], where randomly selected 50% of the targets are held out as training data to train the parameters of each geolocation algorithm using known locations of the targets. The geolocation accuracy is reported with respect to performance on the remaining 50% of the targets used as test data.

Using our commercial data set as targets for geolocation and randomly selected landmarks and monitors, the results in Table 2 show the improvements of the Posit methodology over all existing geolocation techniques. On this dataset, we find that Posit returned median error performance of only 27.2 miles, almost 60% less than all other methodologies. The cumulative distribution of the errors can be seen in Figure 4. Clear improvements are shown using the Posit framework over all the competing techniques for over 90% of the targets. The intuition behind why Posit improves on the other methods is that our landmark distance likelihoods exploit information unseen by previous techniques and our sparse embedding methodology enhances accuracy using the natural geographic clustering inherent in the targets.

Of course, these results represent geolocation performance resulting from randomly chosen monitors and landmarks. To further evaluate Posit and all other geolocation methods, we partition our data set to examine performance under a variety of measurement density regimes relative to the geolocation targets. Specifically, we select monitors and landmarks for the set of targets such that each target belongs to one of three different monitor geographic density regimes, where the closest monitor lies 10 to 75 miles, 75 to 150 miles,

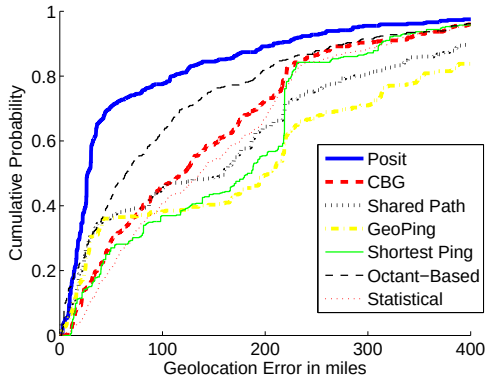

Figure 4: Cumulative distribution of geolocation error for Posit using the commercial dataset (with number of monitors $M = 25$ and number of landmarks $T = 75$).

Table 3: The geolocation error (in miles) for monitor-based methodologies using number of monitors $M = 25$ and number of landmarks $T = 75$ for targets in a monitor dense regime.

Methodology	Mean Error	Median Error
Posit	38.36	22.35
Shortest Ping	81.98	40.31
Constraint-Based	70.18	45.19
Octant Based	79.47	43.44
Statistical	175.55	200.03

or 150 to 250 miles from the target.[4] Also, each target belongs to one of three different landmark geographic density regimes, where the closest landmark lies 0.1 to 5 miles, 5 to 15 miles, or 15 to 30 miles from the target. The landmark density is set considerably closer to the targets due to the significantly larger potential set of landmarks available in the Internet, as recently shown in [8]. To provide a broader perspective on the capabilities of the Posit algorithm, we compare performance against the density regimes that are advantageous to the prior geolocation methodologies.

5.3.1 *Dense Monitor Experiments*

Using the densest monitor regime (where, for each target, at least one monitor is within 10 to 75 miles), we compare performance of Posit with other latency-dependent methodologies (*i.e.,* Shortest Ping, Constraint-based, Octant) which we expect to take the most advantage of the geographically close monitors. The error metrics are seen in Table 3 and the cumulative distribution of errors can be seen in Figure 5-(Left). The results show that Posit outperforms by obtaining median error at least 40% less than all other methodologies.

In terms of Octant-Based geolocation, one concern might be that not all of the targets in our test set have last hop unDNS information, and that is biasing our Octant-Based geolocation results lower. An additional experiment was

[4]The threshold of 250 miles was chosen due to over 90% of the population of the United States residing within 250 miles of the most populous 100 United States cities ([27]).

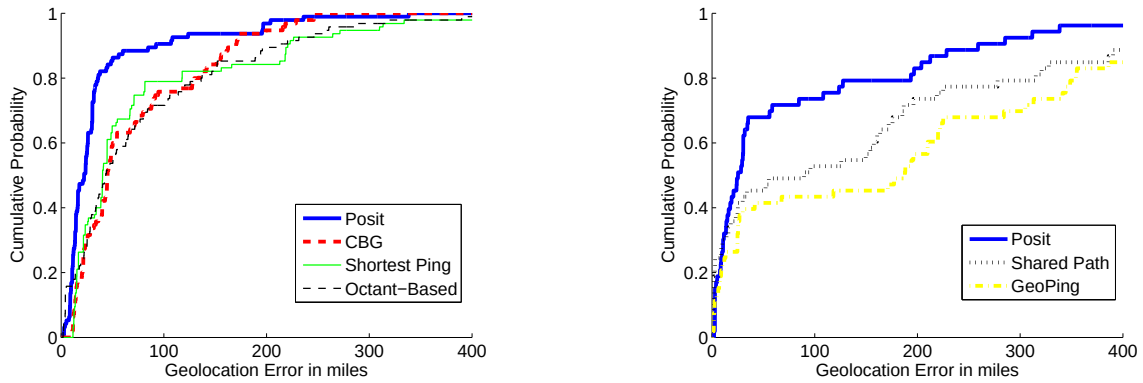

Figure 5: Cumulative distribution of geolocation error for Posit using the commercial dataset with number of monitors $M = 25$ and number of landmarks $T = 75$. (Left) - Compared with monitor-based techniques for targets with close monitor density. (Center) - Compared with landmark-based techniques for targets with close landmark density.

performed restricting geolocation performance only to targets with available last hop unDNS information. The resulting geolocation performance on this subset of targets finds the Octant-Based with a mean error of 123.76 miles and median error of 87.26 miles. Meanwhile for this same set of targets, Posit produces results with a mean error of 76.19 miles and median error of 26.40 miles.

5.3.2 Dense Landmark Experiments

Using the densest landmark regime (where, for each target, at least one landmark is within 0.1 to 5 miles), we compare performance of Posit with other landmark-dependent methodologies (*i.e.,* GeoPing and Shared Path). The error metrics are seen in Table 4 and the cumulative distribution of errors can be seen in Figure 5-(Center). Again, we find that Posit outperforms the other methods in this regime, here with median error at least 65% less than both competing methodologies.

In terms of the Shared Path geolocation methodology, one concern might be that our resolved accuracy deviates significantly from the Street-Level published results. In addition to the reduced number of landmarks considered here, inspection of the `traceroute` derived shared path latency estimates reveal that due to the many disadvantages of `traceroute`-based probing (*e.g.* aliased router interfaces, routers that block ICMP, and potential invisible MPLS routing) the `traceroute`-derived shared path estimates can be wildly inaccurate. We validated our Shared Path implementation by performing their landmark constraint-based methodology using directly observed pairwise latency between the landmarks and the targets, as opposed to their methodology of estimating this data from `traceroute` probes. In these highly idealized tests, our Shared Path implementation returned a mean error of 105.21 miles and median error of 31.71 miles, which is still less accurate than the Posit methodology. This clearly indicates the need for a very dense landmark infrastructure when using a Shared Path-based geolocation methodology.

5.3.3 Sparse Infrastructure Experiments

While we demonstrate that Posit performs well for regimes where either the monitors or the landmarks are geographically dense with respect to the targets, we now examine the performance when the distribution of the measurement infrastructure is *sparse*. We consider a sparse density regime where the monitors are all greater than 150 miles away from our targets and the landmarks are all greater than 15 miles away from our targets. For the results seen in Table 5, we find that the Posit methodology outperforms all competing methodologies with median error 60% less than all other techniques. We additionally note the large degradation in geolocation performance for all methodologies in this regime. This further emphasizes the importance of characterizing geolocation methodologies across multiple measurement infrastructures with varying geographic densities.

Table 4: The geolocation error (in miles) for landmark-based methodologies using number of monitors $M = 25$ and number of landmarks $T = 75$ for targets in a landmark dense regime.

Methodology	Mean Error	Median Error
Posit	80.48	25.34
GeoPing	203.34	187.76
Shared Path	157.99	89.64

Table 5: The geolocation error (in miles) using number of monitors $M = 25$ and number of landmarks $T = 75$ for targets in a monitor sparse and landmark sparse regime.

Methodology	Mean Error	Median Error
Posit	76.10	30.62
GeoPing	270.26	160.74
Shortest Ping	260.48	218.79
Constraint-Based	210.82	203.01
Shared Path	261.49	228.46
Octant Based	126.29	76.51
Statistical	119.52	83.58

5.3.4 Posit Framework Components

Finally, we examine how the separate components of the Posit framework contribute to the accuracy of the method.

Table 6: The geolocation error (in miles) using removing components from the Posit framework.

Posit Modification	Mean Error	Median Error
Full Posit Framework	71.38	27.18
No Landmark Likelihood	77.29	30.19
No Monitor Likelihood	89.96	34.00
No Sparse Embedding	98.22	46.19
No Constraint Regions	102.67	28.81

In Table 6 we leave out each component of the framework and observe the adjusted geolocation accuracy. We find that removing the likelihood information from either landmarks or monitors has a moderate effect on accuracy, while removing the sparse embedding component significantly increases both mean and median error. The removal of the constraint regions (given speed-of-light limitations) has little effect on median error, but causes a large change in mean error as this would allow the Posit framework to geolocate to any monitor/landmark location.

6. CONCLUSIONS

The ability to determine the geographic coordinates of an IP host can be used in many different location-specific applications. Median and worst case errors in predictions made by prior geolocation methods render them ineffective for some classes of location-aware services. The goal of our work is to develop a lightweight IP geolocation methodology that is highly accurate and can compute estimates based on a relatively simple set of measurements.

In this paper, we described a new method for IP geolocation that we call Posit. Our latency-based methodology estimates geographic location using a distance likelihood estimation methodology combined with a new statistical embedding process. This helps mitigates the effects of noisy distance estimation from measurements, and situations where targets lie at distance from the measurement infrastructure.

We assess the capabilities of Posit using a data set of latency measurements collected from hundreds of hosts in the Internet with precisely known geographic coordinates. Our results show that Posit is able to identify the geographic location of target hosts with median error of only 27 miles. We compare this with implementations of the current measurement-based geolocation methodologies, which produces geolocation estimates with median errors of 64 miles or more on the same dataset. These results highlight the powerful capabilities of our approach.

The results of our study motivate future work in a number of areas. First, we plan to expand the scope of the Posit infrastructure to include a larger set of landmarks, which will further improve our estimation accuracy. Also, we plan to begin building an IP geolocation database using Posit that we plan to make available to the community.

7. REFERENCES

[1] R. Sherwood, A. Bender, and N. Spring, "DisCarte: A Disjunctive Internet Cartographer," in *Proceedings of ACM SIGCOMM Conference*, Seattle, WA, August 2008.

[2] D. Magoni and J. Pansiot, "Analysis of the Autonomous System Network Topology," in *ACM SIGCOMM CCR*, vol. 31, no. 3, July 2001.

[3] V. Paxson, "End-to-End Routing Behavior in the Internet," in *IEEE/ACM Transactions on Networking*, vol. 5, no. 5, October 1997.

[4] T. S. E. Ng and H. Zhang, "Predicting Internet Network Distance with Coordinates-Based Approaches," in *Proceedings of IEEE INFOCOM Conference*, New York, NY, June 2002.

[5] "Skyhook Wireless," http://www.skyhookwireless.com/.

[6] B. G. S. Siwpersad and S. Uhlig, "Assessing the Geographic Resolution of Exhaustive Tabulation for Geolocating Internet Hosts," in *Proceedings of PAM Conference*, Cleveland, Ohio, April 2008.

[7] B. Huffaker, M. Fomenkov, and k. claffy, "Geocompare: A Comparison of Public and Commercial Geolocation Databases," in *Cooperative Association for Internet Data Analysis (CAIDA) Technical Report*, 2011.

[8] Y. Wang, D. Burgener, M. Flores, A. Kuzmanovic, and C. Huang, "Towards Street-Level Client-Independent IP Geolocation," in *In Proceedings of USENIX NSDI*, vol. 5, no. 5, Boston, MA, March 2011.

[9] B. Wong, I. Stoyanov, and E. Sirer, "Octant: A Comprehensive Framework for the Geolocation of Internet Hosts," in *USENIX NSDI Conference*, April 2007.

[10] E. Katz-Bassett, J. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe, "Towards IP Geolocation Using Delay and Topology Measurements," in *Proceedings of ACM Internet Measurements Conference*, October 2006.

[11] V. N. Padmanabhan and L. Subramanian, "An Investigation of Geographic Mapping Techniques for Internet Hosts," in *Proceedings of ACM SIGCOMM Conference*, San Diego, CA, August 2001.

[12] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-Based Geolocation of Internet Hosts," in *IEEE/ACM Transactions on Networking*, December 2006.

[13] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP Topologies with Rocketfuel," in *Proceedings of ACM SIGCOMM Conference*, Pittsburgh, PA, August 2002.

[14] M. Zhang, Y. Ruan, V. Pai, and J. Rexford, "How DNS Misnaming Distorts Internet Topology Mapping," in *USENIX Annual Technical Conference*, 2006.

[15] B. Eriksson, P. Barford, J. Sommers, and R. Nowak, "A Learning-based Approach for IP Geolocation," in *Proceedings of PAM Conference*, Zurich, Switzerland, April 2010.

[16] I. Youn, B. Mark, and D. Richards, "Statistical Geolocation of Internet Hosts," in *Proceedings of IEEE ICCCN Conference*, San Francisco, CA, August 2009.

[17] M. Arif, S. Karunasekera, S. Kulkarni, A. Gunatilaka, and B. Ristic, "Internet Host Geolocation Using Maximum Likelihood Estimation Technique," in *Proceedings of IEEE AINA Conference*, Perth, Australia, April 2010.

[18] "A Means for Expressing Location Information in the Domain Name System - RFC 1876."

[19] "Network Time Protocol (Version 3) Specification - RFC 1305."

[20] "The MTR Tool," http://www.bitwizard.nl/mtr.

[21] A. Ziviani, S. Fdida, J. de Rezende, and O. Duarte, "Towards a Measurement-based Geographic Location Service," in *Proceedings of Passive and Active Measurements Conference*, Antibes Juan-les-Pins, France, April 2004.

[22] L. Wasserman, "All of Nonparametric Statistics (Springer Texts in Statistics)." Springer, May 2007.

[23] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001.

[24] A. Lakhina, J. Byers, M. Crovella, and I. Matta, "On the Geographic Location of Internet Resources," in *IEEE Journal on Selected Areas in Communications*, August 2003.

[25] C. Guo, Y. Liu, W. Shen, H. Wang, Q. Yu, and Y. Zhang, "Mining the Web and the Internet for Accurate IP Address Geolocations," in *Proceedings of IEEE Infocom Miniconference*, Rio De Janerio, Brazil, April 2009.

[26] N. L. Caruso, "A Distributed System For Large-Scale Geolocalization Of Internet Hosts," in *Duke University Masters Thesis*, 2011.

[27] "U.S. Census Bureau, http://www.census.gov/."