GEOHYBRID: A HIERARCHICAL APPROACH FOR ACCURATE AND SCALABLE GEOGRAPHIC LOCALIZATION

Ibrahima Niang and Bamba Gueye and Bassirou Kasse

Université Cheikh Anta Diop de Dakar
Senegal

ABSTRACT

Geographic location and Grid computing are two areas that have taken off in recent years, both receiving a lot of attention from research community. The Grid Resource Brokers, which tries to find the best match between the job requirements and the resources available on the Grid, can take benefits by knowing the geographic location of clients, for a considerable improvement of their decision-taking functions. A measurement-based geolocation service estimates host locations from delay measurements taken from landmarks, which are hosts with a known geographic location, toward the host to be located. Nevertheless, active measurement can burden the network. Relying on database-driven geolocation and active measurements, we propose GeoHybrid. GeoHybrid estimates the geographic location of Internet hosts with low overhead as well better accuracy with respect to geolocation databases. Afterwards, we propose a geolocation middleware for grid computing. By defining the architecture and the methods of this service, we show that a promising symbiosis may be envisaged by the use of the proposed middleware service for grid computing.

Index Terms— Geolocation, Measurement, Grid performance optimization, Resource Broker

1. INTRODUCTION

Geographically locating an Internet host from its IP address enables a diversified and interesting new class of location-aware application. Nowadays a lot of services depend on the geographic location of Internet hosts. Examples of such applications comprise targeted advertising on web pages, displaying local events and regional weather, automatic selection of a language to first display the content of web pages, restricted content delivery following regional policies, and authorization of transactions only when performed from pre-established locations.

Multimedia delivery systems, such as Content Distribution Networks (CDNs) that offers a world wide service but has limited servers, can also benefit from knowing the location of their clients [1]. For example, benefits include the indication of nearby servers to clients or the location-based adaptation of the multimedia content. In other words, the nearest geographically located server, which in most cases is likely to have the lowest latency and/or highest bandwidth path.

Active measurement-based IP geolocation techniques have been proposed [2–6], and these may achieve desirable properties, such as accuracy, and robustness. These techniques use a set of reference hosts, called landmarks, to estimate the location of other hosts. However, these properties come at the expense of scalability, high overhead and very high response time ranging from tens of seconds to several minutes to localize a unique IP address. This is several order of magnitude slower that what is achievable with database-driven geolocation, representing the passive approach.

Database-driven geolocation usually consists of a database-engine (e.g., SQL/MySQL) containing records for a range of IP addresses, which are called *blocks* or *prefixes*. Geolocation prefixes may span non-CIDR subsets of the address space, and may even span only a couple of IP addresses. Examples of geolocation databases are *GeoURL* [7], the *Net World Map* project [8], and free [9–11] or commercial tools [12–16]. When coupled with a script embedded in a website and upon a client access to the website being detected, a request can be sent instantly to the database. This request can be to check if the IP address has an exact or longest prefix match (LPM) with a corresponding geographic location and coordinate. Since there is no actual measurement involved but merely a simple lookup, the request can be served in a matter of milliseconds. The expected time for which a website should be fully loaded, without causing any nuisance, is in general within one second.

Nevertheless, exhaustive tabulation is difficult to manage and to keep updated, and the accuracy of the locations is unclear. In practice however, most location-aware applications seem to get a sufficiently good geographic resolution for their purposes. Siwpersad et al. in [17] have shown that the geographic resolution of databases is far coarser than the resolution provided by active measurements, typically several times coarser than the confidence given by active measurements. As most geolocation databases do not give confidence in the accuracy of their location records, they are likely not to be trustworthy sources of geolocation information if precise IP address-level locations are required. Also, the geographic dispersion between results from several databases can span an entire region.

It became clear that solely relying on databases leads to incorrect results or results that have a high geographic dispersion. Furthermore, measurement-based geolocation can burden the network with extra traffic and can therefore trigger intrusion detection systems. We aim at mitigating the number of measurements generated in the network. To overcome theses limitations, we propose an hybrid geolocation service called *GeoHybrid*. Firstly, the technique GeoHybrid uses a database to find the geographic location of the IP block which hosts the IP of the target. Secondly, in order to improve the provided localization, GeoHybrid selects either few landmarks located at the vicinity of the geographic location of the IP block (heuristic choice) or randomly selects few landmarks. Afterwards, we localize target hosts with lower number landmarks compared to [2], and thus, mitigate the impact of measurements. Note that, the measurement tasks are done with the Constraint-Based Geolocation (CBG) technique. Furthermore, we improve the accuracy of geolocation databases. Afterwards, we compare both approaches (*i.e.* heuristic choice and random choice). The obtained results show that the heuristic choice outperforms the random choice.

The geographical distributed computing architectures - the so-called Grid - appear as new trend in supercomputing and distributed computing [18]. The users that perform operations such as submit jobs, control their execution and retrieve their output, demand resources allocation simultaneously. The quality of this service depends directly on the network condition, and the computation capacity of each cluster. Therefore, geolocation tools may contribute in supporting a highly dynamic environment where operational conditions are constantly changing. In fact, job execution may require one or more files and produces output data, thus, given the distribute nature of the databases, the input/output process can produce considerable data traffic across the Grid. Furthermore, if the same amount of resources are available everywhere, GeoHybrid can permit to the *Workload Management System* (WMS) to delegate jobs to the closest cluster. Note that, the WMS has the responsibility of managing the Grid resources. Furthermore, we can do a geographic mapping of different resources available on the Grid, and thus allow users the possibility to send their jobs following geographic constraints.

This paper is organized as follows. Section 2 reviews the related work on this field. Section 3 describes the CBG approach to estimate the geographic location of a given target host. In Section 4, we introduce our hybrid geolocation service and points out the contributions of GeoHybrid in contrast to previous approaches. Following that, we present results for datasets in Section 6. We illustrate the use of geolocation in case of Grid computing in Section 7. Finally, we conclude and present some research perspectives in Section 8.

2. RELATED WORK

A DNS-based approach to provide a geographic location service of Internet hosts is proposed in RFC 1876 [19]. This proposition, however, is not widely adopted since it requires changes in DNS structure and administrators have no motivation to register new location records. Tools such as [20,21] query Whois databases in order to obtain the location information recorded therein to estimate the geographic location of a host. This information, however, may be inaccurate or stale. Moreover, if a large and geographically dispersed block of IP addresses is allocated to a single entity, the Whois databases may contain just a single entry for the entire block.

There are also some geolocation services based on an exhaustive tabulation between IP addresses ranges and their locations. This is the case of some projects [7, 8] or commercial services [12, 15, 16]. Exhaustive tabulation is difficult to manage and to keep updated and unreliable, since the accuracy is hard to determine and it also relies on how truthfully a user has submitted his personal information. Furthermore, the results are usually coarse grained and not suited for applications where accuracy is required. The authors of [22] quantify the extent to which locating all IP addresses within a block leads to an inaccurate geolocation of Internet hosts. With active measurements, they show that the geographic span of block of IP addresses make their location difficult to choose. Therefore, using a unique location for a block of IP addresses as an estimate of the location of its IP addresses leads to significant localization errors, whatever the choice made for the location of the block.

Different techniques [3] estimate the geographic location of an Internet host from DNS names, from clustering the IP address space with BGP prefix information, or from delay measurements. An example of a discrete measurement is the GeoPing [3] approach where the location is based on the nearest landmark, thus having a discrete space of answers. In contrast, the Constraint-Based Geolocation (CBG) [2], where landmarks are used as well, the estimation is based on multilateration providing a continuous space of locations. The authors of [23] present a topology-based geolocation method. They extend multilateration techniques with topology information. In fact, they use traceroute from landmarks to map topology.

Nevertheless, measurement-based approaches burden the network with extra traffic and can therefore trigger intrusion detection systems (IDS). If an IDS is alarmed, it might block future access at some points in the route, which evidently will lead to incorrect measurements as well.

3. BACKGROUND ON CBG APPROACH

In this section, we present a brief background on how CBG provides geolocation estimation for target hosts based on delay measurements.

3.1. Multilateration with geographic distance constraints

The physical position of a given point can be estimated using a sufficient number of distances or angle measurements to some fixed points whose positions are known. When dealing with distances, this process is called multilateration.

Consider a set $\mathcal{L} = \{L_1, L_2, \ldots, L_K\}$ of K landmarks. Landmarks are reference hosts with a well-known geographic location. For the location of Internet hosts using multilateration, CBG [2] tackles the problem of estimating the geographic distance from these landmarks towards the target host to be located, given the delay measurements from the landmarks. From a measurement viewpoint, the end-to-end delay over a fixed path can be split into two components: a deterministic (or fixed) delay and a stochastic delay [24]. The deterministic delay is composed by the minimum processing time at each router, the transmission delay, and the propagation delay. This deterministic delay is fixed for any given path. The stochastic delay comprises the queuing delay at the intermediate routers and the variable processing time at each router that exceeds the minimum processing time. Besides the stochastic delay, the conversion from delay measurements to geographic distance is also distorted by other sources as well, such as circuitous routing and the presence of redundant data. Anyway, it should be noted that no matter the source of distortion, this delay distortion is always additive with respect to the minimum delay of an idealized direct great-circle path.

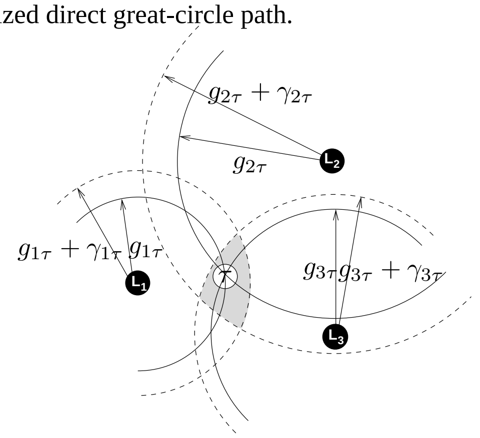


Fig. 1. Multilateration with geographic distance constraints.

Figure 1 illustrates the multilateration in CBG using the set of landmarks $\mathcal{L} = \{L_1, L_2, L_3\}$ in the presence of some additive distance distortion due to imperfect measurements. Each landmark L_i intends to evaluate its geographic distance constraint to a target host τ with unknown geographic location. Nevertheless, the inferred geographic distance constraint is actually given by $\hat{g}_{i\tau} = g_{i\tau} + \gamma_{i\tau}$, *i.e.* the real geographic distance $g_{i\tau}$ plus an additive geographic distance distortion represented by $\gamma_{i\tau}$. This purely additive distance distortion $\gamma_{i\tau}$ results from the possible presence of some additive delay distortion. As a consequence of having additive distance distortion, the location estimation of the target host τ should lie somewhere within the gray area (*cf.* Figure 1) that corresponds to the intersection of the overestimated geographic distance constraints from the landmarks to the target host.

3.2. From delay measurements to distance constraints

Previous work [3, 25] has investigated the correlation between geographic distance and network delay. Figure 2 provides an example of the relation between the distance and the delay for one of the landmarks we used in our measurements towards the remaining landmarks of our dataset (further details on the experimental data used are found in Section 6). The *bestline* shown in Figure 2 for a given landmark L_i is defined as the line that is closest to, but below all data points (x, y), where x expresses the actual great-circle geographic distance between this given landmark and all the other landmarks in the set, while y represents the measured RTT between the same pairs. The equation of the bestline is defined as

$$y = m_i x + b_i. \quad (1)$$

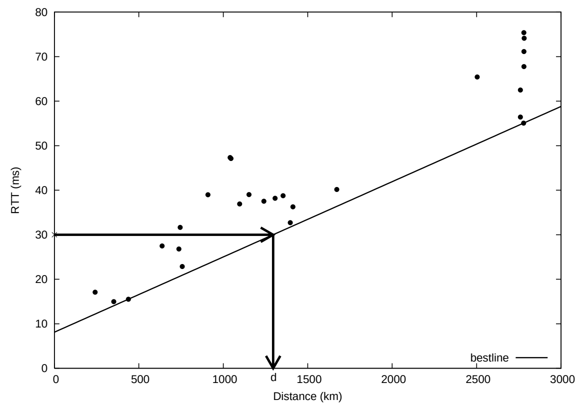


Fig. 2. Sample scatter plot of geographic distance and network delay.

It should be noted that each landmark finds its slope m_i and its positive intercept b_i based only on delay measurements between the available landmarks. For further details about the computation of b_i and m_i, we refer the reader to [2]. The presence of a positive intercept b_i in the bestline reflects the presence of some localized delay. Each landmark uses its own bestline to convert the delay measurement towards the target host into a geographic distance constraint. A delay measurement from the considered landmark of Figure 2 towards a particular target host τ is transformed into a distance constraint by projecting the measured delay on the distance axis using the computed bestline of this landmark. For example, if the measured delay is 30 ms, the distance constraint is d, as illustrated by the thick arrow in Figure 2. This estimated geographic distance constraint $\hat{g}_{i\tau}$ between a landmark L_i and a target host τ is derived from the delay $d_{i\tau}$ using the bestline of the landmark as follows:

$$\hat{g}_{i\tau} = \frac{d_{i\tau} - b_i}{m_i}. \quad (2)$$

Each landmark L_i localizes a given destination τ inside a circle whose radius is the obtained distance constraint $\hat{g}_{i\tau}$. The region formed by the intersection of all these circles from the set of landmarks is called in CBG the *confidence region*. CBG provides the centroid of this confidence region as the location estimation for the target host.

4. GEOHYBRID LOCALIZATION FRAMEWORK

The goal of GeoHybrid are twofold: (*i*) mitigate the number of measurements by reducing the number of landmarks used for geolocating target hosts, and thus enhance scalability; (*ii*) improve the accuracy of geolocation databases. In fact, using the single location for a block of IP addresses as an estimation of the location of its IP addresses leads to significant localization errors, whatever the choice made for the location of the block [22].

4.1. Hybrid geolocation framework

Fig. 3 illustrates the different components of our hybrid geolocation service. The geolocalization framework can be decomposed as follows :

- A database which contains block of IP addresses (entries). In fact, a database entry is composed of a pair of values, corresponding to the integer representation of the minimum and maximum address of a block. Each block is then associated with several informations helpful for localization: country code, city, latitude and longitude, and Zip code.

- A given server where is implemented the heuristic which allows to trigger, if necessary, measurements from landmarks towards a fixed host.

- Afterwards, if measurement task is needed, the server delegates the measurements to few landmarks which are chosen following a fixed rules. It is worth noticing that the selected landmarks use CBG technique 3 to localize target hots.

The process of locating a given target with GeoHybrid host is more explained in Section 5.

4.2. Structure of database used for IP geolocation

According to the GeoHybrid framework, as illustrated in Fig. 3, when a request arrives for geolocation purposes, the server should use a database to geolocate the target host. In the sequel of this paper we restrict our attention to one commercial database called GeoIP. This database, GeoIP by Maxmind [16] is used because of its popularity (see [16] for a listing of some of their customers) and its expected reliability.

In fact, the Maxmind database is split into two parts as depicted in the Table 1: *table 1* and *table 2*. One part contains the IP prefixes and a location identifier (loc id). The other part consisted of the representation of the location identifiers such as country, region, city, zip code and geographic coordinate. Maxmind contains more than 3 millions of block of IP prefixes. It should be noted that “lon.” and “lat.” means longitude and latitude respectively in Table 1.

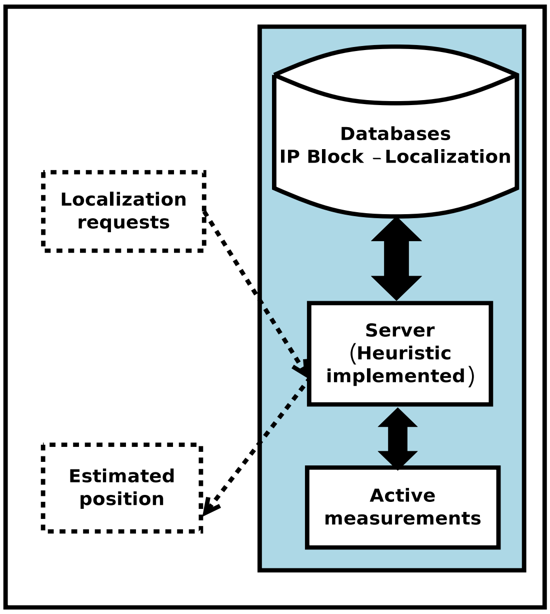


Fig. 3. Hybrid geolocation framework.

Table 1. Database fiels.

table 1	
IP prefix	loc id

table 2						
loc id	country	region	city	zip code	lat.	lon.

Using an exhaustive tabulation as in [7, 8, 16], we find the IP block which owns the IP address of the target host. By knowing the location of this IP block, one can determine from table 2 the geographic location of the target host. It should be noted that the goal of the exhaustive tabulation is to check if the IP address has an exact or longest prefix match with a corresponding geographic location and coordinate. As we know the coarse grained location of the target host, we can select the set of landmarks $\mathcal{L}$, following a given criteria, that should perform measurement task. Otherwise, if the IP target belongs to any database’s block, we should use all landmarks available in our measurement infrastructure to estimate the position of the target.

5. HEURISTIC CHOICE OF LANDMARKS

As shown in the GeoHybrid framework (Fig. 3), the server implements several heuristics simultaneously for the selection of probes (landmarks). The core feature of GeoHybrid is its capability to use only the set of landmarks located at the vicinity of the IP prefix that owns the target host. It is worth noticing that the geographic location of the set of landmarks $\mathcal{L}$ is known. After having the geographic location of the IP prefix that hosts the target, from Maxmind database, we can estimate the distance between the set of landmarks and the

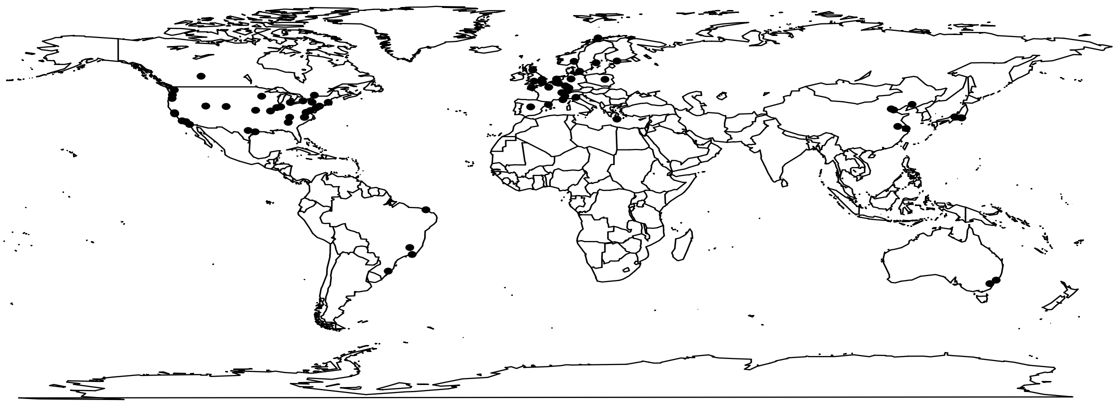


Fig. 4. Geographic location of landmarks.

target host. Based on [26], the geographic distance between each landmark L_i and the target host τ can be estimated as follows:

$$\beta = \sqrt{\left(\sin\left(\frac{lat_i - lat_\tau}{2}\right)\right)^2 + \cos(lat_i) \times \cos(lat_\tau) \times \alpha} \quad (3)$$

$$\alpha = \left(\sin\left(\frac{lon_\tau - lon_i}{2}\right)\right)^2 \quad (4)$$

$$\hat{dist}_{i\tau} = 6371 \times 2 \times \arcsin(\beta) \quad (5)$$

It should be noted that lat_i and lon_i represent the latitude and longitude, expressed in radian, of landmark L_i; lat_τ and lon_τ, also in radian, represent the latitude and longitude of the target host τ. Afterwards, the geographic distance (in km), between landmark L_i and the target τ is obtained from equation 5. The value 6371 used in equation 5 represents the radius of the earth and the product $2 \times \arcsin(\beta)$ gives the geographic distance expressed in radian. It is worth noticing that in section 6, the distance are expressed in km. For the target host τ, we obtained the following distance vector:

$$D_\tau = [\hat{dist}_{1\tau}, \hat{dist}_{2\tau}, \ldots, \hat{dist}_{K\tau}], \quad (6)$$

where K represents the total number of landmarks of $|\mathcal{L}|$, and $\hat{dist}_{i\tau}$ represents the geographic distance (in km), computed between the landmark L_i and the target τ for $1 \leq i \leq K$.

Assume that we would like to choose only n among the K landmarks which form the set of landmarks $\mathcal{L}$ for measurement purposes. The goal of our heuristic is to find the n nearest landmarks towards the target hosts. In other words, we should find the smallest distances $\hat{dist}_{i\tau}$, $1 \leq i \leq n$ with respect to equation 6.

6. EVALUATION

6.1. Datasets

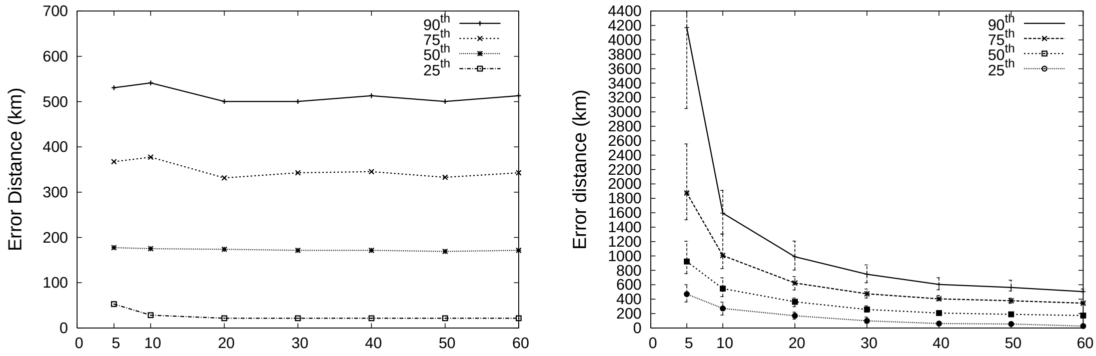
To validate our heuristic, we use two datasets formed by RIPE hosts [27] and AMP hosts [28]. The experimental datasets comprise 127 hosts located in United States and Europe. The main reason for this restriction is that the datasets we have had correspond to hosts located in these regions. Unfortunately, datasets that provide the geolocation of the involved hosts are uncommon. Nevertheless, we indeed believe that the results we report in this paper are interesting and promising in spite of being limited to the U.S. and Europe.

In this paper, we consider *MaxMind* [16] which is a commercial geolocation database. Maxmind database is formed by more than 3 millions IP blocks and each block is associated with several informations helpful for localization: country code, city, latitude and longitude, and Zip code (Table 1). Note that block prefixes are between /8 and /32. Nevertheless, most of IP block from Maxmind correspond to subnet smaller than /25.

In our experiments, for geolocating target hosts, we consider 74 PlanetLab [29] nodes as landmarks. Their geographic distribution is illustrated in Fig. 4. Landmarks perform *ping* measurements towards a given target host to locate it. Each ping is composed by 10 packets sent by interval of 1 second. The inter-packet spacing is due to the fact that we do not want to trigger IDS alarm. Each packet has a size of 1024 Ko. Only the minimum RTT (Round Trip Time) is considered. In order to locate target host we use the CBG methodology described in section 3.

6.2. Results

In this section, we evaluate the impact of the number of adopted landmarks in the performance of GeoHybrid. After inferring the point estimate for each considered target host, we compute the error distance, which is the difference



(a) Landmarks chosen by an heuristic

(b) Landmark chosen at random

Fig. 5. Error distance as a function of the number of landmarks.

between the estimated position and the real location of the target host τ.

Fig. 5 shows different percentile levels of the error distance of GeoHybrid location estimates as a function of the number of adopted landmarks. For example, the 90th percentile curve represents the error distance at which the CDF plot of mean error distance meets the 0.90 probability mark. The x-axis is the number of chosen landmarks among all landmarks, and the y-axis is the difference between the estimated position and the real location of the target host. The number of landmarks varies between 5 and 60.

Fig. 5(a) illustrates the case where landmarks have been chosen according to their vicinity to the location of the IP block which hosts the target (*i.e.* heuristic choice). We remark that a certain number of landmarks, typically about 20, is needed to level off the error distance (Fig. 5(a)). Nevertheless, for curves illustrated the 90th and 75th percentile, we have a slight rise of the estimation error. Probably, the reason is due to the presence of some distortion in our delay measurements caused by the added landmarks, which are far with respect to the target hosts. Nevertheless, the general trend observed in Fig. 5(a) is, more chosen landmarks are the closest towards the target hosts and more the estimation is better. Indeed, by considering only the closest 20 landmarks, 50% of target hosts are located with an error distance lower than 175 km.

Fig. 5(b) illustrates the impact of the number of adopted landmarks in the performance of GeoHybrid. Note that, the choice of landmarks is done randomly. We compute the mean error distance as the average of all error distances corresponding to several random sets of k landmarks chosen out of the total number of available landmarks (74 landmarks). Because the number of possible placement combinations become very large as we increase k, we do not consider all the possible choices of k landmarks. Error bars indicate the 99% confidence interval. These results suggest that a certain number of landmarks, typically about 30, is needed to level off the mean error distance. Nevertheless, the obtained error with random approach is upper than the heuristic choice. Indeed, with 30 landmarks chosen randomly, 50% of target hosts are localized with an error lower than 400 km. In contrast, the heuristic choice has an estimate error lower than 175 km for 50% of target hosts.

By considering a few number of landmarks we reduce the amount of time needed to localize a target host, and thus the response time is widely shortened. Furthermore, we mitigate the number of traffic generated in the network.

7. GEOLOCATION SERVICE FOR GRID COMPUTING MIDDLEWARE

The integration of geolocation information can be extremely useful for the optimization of the decision taking process of a Grid Resource Broker. For instance, the GeoHybrid service can be used for the improvement of data management among different *Storage Elements*: for the selection of the nearest replica of a given file if multiples copies of it are present in different storage elements.

7.1. DataGrid overview

The Workload Management System (WMS) is the component of the Grid that has the responsibility of managing the Grid resources, (*i.e.* in each Site i (Fig.6)), in such a way that applications are conveniently, efficiently and effectively executed. It is formed by the :

- *User Interface (UI)*: it allows a user to interact with the Grid in order to perform operations such as submit jobs, control their execution, and retrieve their output.

- *Resource Broker (RB)*: it is the core component of the WMS. The RB tries to find the best match between the job requirements and the resources available on the Grid whose characteristics are retrieved from *Information System* (Fig. 6). The output of the search is a *Computing Element* where the job, while running, has access to all resources specified in the job description, such as data or storage space.

- *Logging and Bookkeeping (LB) Service*: it is the Grid service responsible to store and manage logging information which concerns the WMS itself. Further-

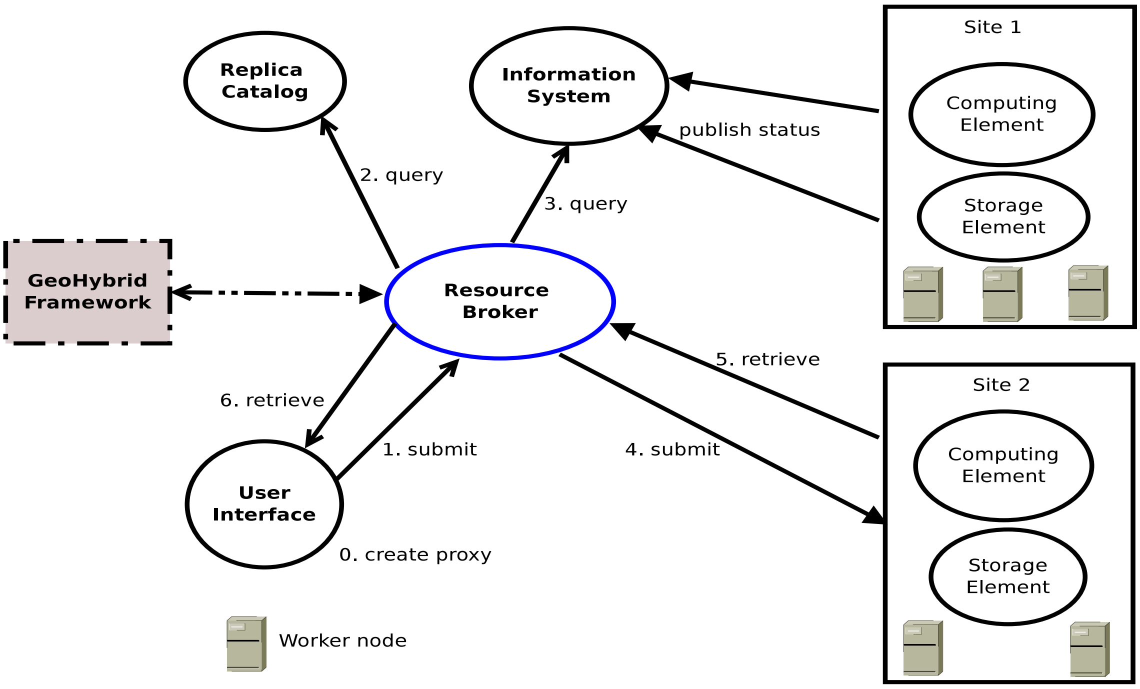


Fig. 6. A logical view of the Geolocation-based Grid Optimizer.

more, the bookkeeping collects information about active jobs, i.e, jobs that are within the WMS. It consists of the job definition, its status, resource consumption. In particular from the events stored in the logging and bookkeeping databases it is possible to reconstruct the status of a job that was previously submitted to a Resource Broker for execution on the Grid. The LB is located inside the Information System shown in Fig. 6.

- *Top-BDII (Information Index)*: it is a LDAP (Lightweight Directory Access Protocol) server which collects the different resources available in the Grid. It is used by the Resource Broker in order to select resources. It should be noted that each site can have its own BDII called Site-BDII. In such case, it collects the available resources in the site, from the Computing Element, and shares this information with the Top-BDII. The Top-BDII is located inside the Information System.

7.2. Optimization scenarios

Fig. 6 illustrates a grid optimization service. Let us assume that a user wants to send a job to a RB. Firstly, it needs to access to User Interface; it obtains a timeout of 24 hours for doing its job ("*create proxy*") (see Fig. 6). Note that, this timeout can be renewed. Afterwards, the User Interface submits the job to the RB, and then the RB sends a request to a *Replica Catalog* (Fig.6) in order to verify if it is possible to realize this task. In such case, the RB queries the *Information System*, and thus receives a list of candidate worker nodes often geographically distributed. Note that, this list contains the best computing element for a given job execution. Afterwards, the RB can send a request to the GeoHybrid server, as illustrated in Fig. 3, in order the find the geographic location of the user and the worker nodes. Following the obtained responses from GeoHybrid, the RB selects the closest worker node towards the user among the list of candidate worker nodes. Therefore, according to this heuristic, we mitigate the amount of traffic exchanged across the Grid.

8. CONCLUSION

In this paper, we proposed the GeoHybrid framework, a scalable measurement-based method to estimate the geographic location of Internet hosts. Relying on geolocation database and active measurement, GeoHybrid estimates the geographic location of Internet hosts with lower overhead by reducing the number of used landmarks. Using active measurement, GeoHybrid provides also better accuracy with respect to geolocation databases by improving their geographic estimation which is coarse grained.

Our experimental results show that the heuristic choice, where we select only the closest landmarks towards a given IP block, outperforms the approach where landmarks are chosen randomly. Indeed, with 30 landmarks chosen randomly, 50% of target hosts are localized with an error lower than 400 km. In contrast, the heuristic choice has an estimate error lower than 175 km for 50% of target hosts and typically about 20 landmarks, is needed to level off the error distance.

The synergy between the areas of grid computing and geographic location points out the importance of a specific measurement middleware service. Based on GeoHybrid, we improve the selection mechanism of worker nodes from the Resource Broker. Indeed, the candidate worker nodes can be sorted following their vicinity to the user which sent the job. Therefore, the amount of traffic generated across the Grid is minimized.

Our future work consists to implement this middleware in the Research Education Network which interconnects different Universities and High schools in Senegal.

9. REFERENCES

[1] Kunwadee Sripanidkulchai, Bruce Maggs, and Hui Zhang, "Efficient content location using interest-based locality in peer-to-peer systems," in *Proc. IEEE INFOCOM*, April 2003.

[2] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-based geolocation of Internet hosts," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1219–1232, December 2006.

[3] V. N. Padmanabhan and L. Subramanian, "An investigation of geographic mapping techniques for Internet hosts," in *Proc. ACM SIGCOMM*, August 2001.

[4] E. Katz-Bassett, J. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe, "Towards IP geolocation using delay and topology measurements," in *Proc. ACM SIGCOMM Internet Measurement Conference (IMC)*, October 2006.

[5] B. Wong, I. Stoyanov, and E. G. Sirer, "Geolocalization on the Internet through constraint satisfaction," in *Proc. USENIX Workshop on Real, Large Distributed Systems (WORLDS)*, Seattle, WA, November 2006.

[6] B. Gueye, S. Uhlig, A. Ziviani, and S. Fdida, "Leveraging buffering delay estimation for geolocation of Internet host," in *Proc. IFIP Networking Conference*, Coimbra, Portugal, May 2006.

[7] "GeoURL," http://www.geourl.org.

[8] "Net World Map," http://www.networldmap.com.

[9] "Host IP," http://www.hostip.info.

[10] IPInfoDB, "Free IP address geolocation tools," http://ipinfodb.com/.

[11] Software 77, "Free IP to country database," http://software77.net/geo-ip/.

[12] GeoBytes Inc., "GeoNetMap - geobytes' IP address to geographic location database," http://www.geobytes.com/GeoNetMap.htm.

[13] Qwerks Inc., "WhereIsIP - IP whois tool," http://www.jufsoft.com/whereisip.

[14] Hexasoft Development Sdn. Bhd, "IP address geolocation to identify website visitor's geographical location," http://www.ip2location.com.

[15] Quova Inc., "GeoPoint - IP geolocation experts," http://www.quova.com.

[16] MaxMind, "Geolocation and online fraud prevention from MaxMind," http://www.maxmind.com/.

[17] S. Siwpersad, B. Gueye, and S. Uhlig, "Assessing the geographic resolution of exhaustive tabulation for geolocating Internet hosts," in *Proc. of PAM*, Cleveland, Ohio, USA, April 2008.

[18] Ian Foster, "The grid: A new infrastructure for the 21st century science," in *Grid Computing: Making the Global Infrastructure a Reality*. 2003, pp. 51–63, John Wiley & Sons.

[19] C. Davis, P. Vixie, T. Goodwin, and I. Dickinson, "A means for expressing location information in the domain name system," *Internet RFC 1876*, Jan. 1996.

[20] University of Illinois at Urbana-Champaign, *IP Address to Latitude/Longitude*, http://cello.cs.uiuc.edu/cgi-bin/slamm/ip2ll/.

[21] D. Moore, R. Periakaruppan, J. Donohoe, and k.c. Claffy, "Where in the world is netgeo.caida.org?," in *Proc. of INET*, Yokohama, Japan, July 2000.

[22] B. Gueye, S. Uhlig, and S. Fdida, "Investigating the imprecision of IP block-based geolocation," in *Proc. Passive and Active Measurement Conference (PAM)*, Louvain-la-neuve, Belgium, April 2007.

[23] Ethan Katz-Bassett, John P. John, Arvind Krishnamurthy, David Wetherall, Thomas Anderson, and Yatin Chawathe, "Towards ip geolocation using delay and topology measurements," in *Proc. of IMC*, New York, NY, USA, 2006, pp. 71–84, ACM.

[24] C. J. Bovy, H. T. Mertodimedjo, G. Hooghiemstra, H. Uijterwaal, and P. van Mieghem, "Analysis of end-to-end delay measurements in Internet," in *Proc. of PAM workshop*, Fort Collins, CO, USA, Mar. 2002.

[25] Artur Ziviani, Serge Fdida, José Ferreira de Rezende, and Otto Carlos Muniz Bandeira Duarte, "Toward a measurement-based geographic location service," in *Proc. of PAM*, Antibes Juan-les-Pins, France, Apr. 2004, Lecture Notes in Computer Science (LNCS) 3015, pp. 43–52.

[26] *Ed Williams*
, http://http://williams.best.vwh.net/avform.html.

[27] *RIPE Test Traffic Measurements, 2000*, http://www.ripe.net/ttm/.

[28] *NLANR Active Measurement Project, 1998*, http://watt.nlanr.net/.

[29] PlanetLab, "An open platform for developing, deploying, and accessing planetary-scale services," 2002, http://www.planet-lab.org.