TULIP Comparing geolocation techniques

Introduction

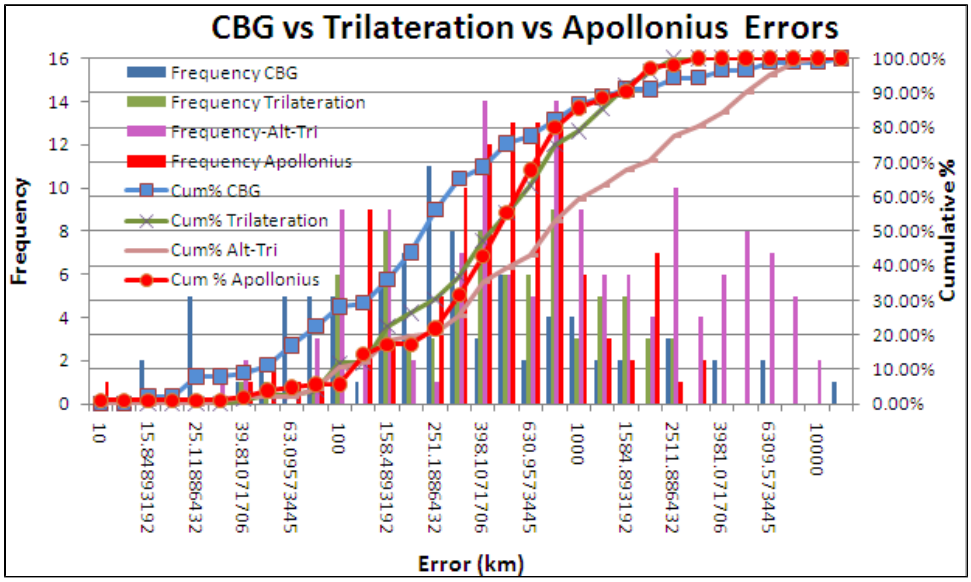
We want to compare the various methods of geolocation using ping RTT measurements to estimate the distance between landmarks and targets. The landmarks are at known lat/longs and the min RTT from the three (Tri-Lateration) or more closest landmarks to the target are used. From each min RTTs the distance to the target is estimated as *distance (km) = alpha * minRTT(ms) * 100 (km/ms)*. In these tests we use each of the other landmarks (at known locations) one at a time as targets (si we know the location of the targets also). Comparing the actual location of the target and the estimated location we were able to calculate the error as the distance between these two values. There is a spreadsheet with more details.

Constraint Based Geolocation (CBG) using Tri-lateration vs Tri-Lateration with no constraints

We started with 174 targets. Of these CBG modified to only use 3 landmarks returned 15 with no useful result. Tri-Lateration was only able to provide results for 76 of the targets. The number of targets that were found with both CBG with tri-lateration and tri-lateration with no constraints (henceforth referred to simply as Tri-lateration) was 73.

For these 73 targets CBG using tri-lateration gave a lower error 63/73 times and Tri-lateration 10/74 times. If we also remove the results where the errors were < 1km (i.e. the target's geolocation was being estimated by nearby landmarks on the same site) then the number of useful results dropped to 41 with CBG using Tri-lateration having a lower error 32/41 times. The distributions (excluding CBG with tri-lateration results where the error was < 1km) are compared in Figure 1 below.

Figure 1: Histograms of the frequency and cumulative distributions for CBG with tri-lateration, Trilateration and Apollonius algorithms for estimating the geolocation



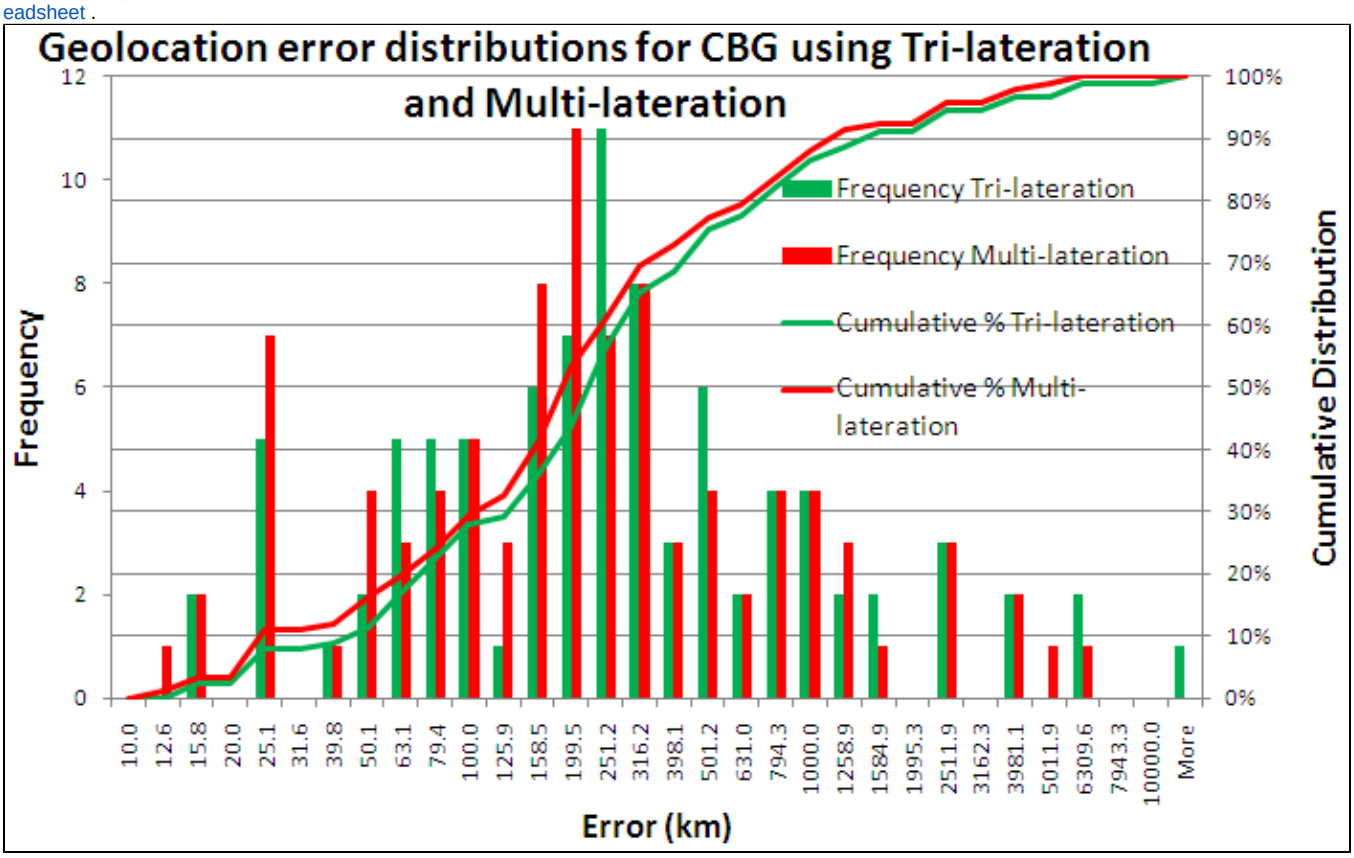
It is seen that Trilateration with no constraints has a median value of ~ 412km, while CBG with Tri-lateration is closer to 250km.

Comparison with the Apollonius Geolocation Algorithm

We also evaluated the performance of the Apollonius geolocation algorithm that uses tangential circles rather than the intersecting circles of tri-lateration. In this case we had successes with 105 targets. The frequency and cumulative distributions are seen in Figure 1. It is seen that Apollonius performs similarly to Tri-lateration with a median error of about 449ms.

Comparison of CBG with Multi-lateration and Tri-lateration

We then compared our modified CBG using tri-lateration with CBG using Multi-lateration. The distributions are shown below, and more details are in the spreadsheet .



It is seen that the two dsitributions are very similar with multi-lateration having a smaller median error. It also has a higher success rate (see the table below).

Metric	CBG with Multi-lateration	CBG with tri-lateration	Trilateration	Apollonius
% success rate	92%	91%	44%	63%
Median	190km	250km	413km	449km

Files

outputDistance.csv	Created by Zafar Gilani, sent by email 6/14/2010. It compares trilateration (3 landmarks) with mulitlateration (3 to 5 landmarks) giving the target hostname, IP address, its actual lat/longs, the targets estimated location, landmark hostnames, landmark lat/longs and error between estimated and actual
acuumulated_results.xlsx	Created by Fida, sent by email 6/10/2010. It is a compendium comparing the several of the geolocation methods including:CBG, SOI, TBG, TBG_Updtaed, Apollonius, Triilateration
cbg_tri_lateration_vs_new_tri_lateration.xlsx	File Sent by Zafar 3.08pm Jun 1 2010. Compares improved trilateration (by Farrah) vs CBG trilateration. - There are a total of 174 targets for CBG out of which 131 are those which ignore values that are either "0<error<1" or "NaN". - Only 74 targets are ones that overlap between CBG tri-lateration and improved tri-lateration. - If I don't ignore CBG's values that have estimate error in the range "0<error<1" then CBG performs 64/74 times better and tri-lateration performs only 10/74 times better. - But even if I ignore values with error estimate "0<error<1" then CBG performs 32/74 times better, improved tri-lateration performs 10/74 times better and rest are unaccounted.
m_vs_t_rtt_new.xlsx	From Zafar by email 6/2/2010 2:06am. So far what I've gathered from doing this: There are so-called "bad" landmark estimate values in target files which causes these. There is also a portion in the code that deliberately ignores such values (see 1 under "Results, observations and explanation" here). By restricting n to 10 and 4 I've managed to remove those "bad" values for 39 and 14 targets respectively.
cbg vs trilateration (zafar) v2.xlsx	From Zafar, direct upload on 6/19/2010 1:41am. This spreadsheet provides a histogram of the errors for 74 overlapping results between CBG trilateration and improved trilateration (by Farrah).
all-analysis.xlsx	From Zafar, direct upload on 7/14/2010 6:46pm. This file contains all sorts of detailed information regarding all studied geolocation techniques. Furthermore the spreadsheet also contains graphs for: - Error frequency and cumulative percentage per technique. - Number of targets detected per geolocation technique.

Procedure to generate analysis for all studied geolocation techniques

We created a partially automated procedure to collect data from multiple spreadsheets into a single detailed spreadsheet for comprehensive analysis. The trouble is that different geolocation techniques give results for different sets of targets (or hosts) and this number varies largely from one spreadsheet to another. Furthermore there are inconsistencies in data and data formatting. However to cope with all of this and generate an analysis follow the guidelines below:

1. The first step requires us to create multiple CSV files. Each CSV file will correspond to an independent geolocation technique. Open a spreadsheet already created and copy **target IPs** and **error distance** columns into a new spreadsheet and save it as a CSV (.csv) file. Name it against the geolocation technique such as **apollonius.csv** for Apollonius. Table 2 below shows geolocation technique against its file name.
    a. It is a possible that for some geolocation techniques, we might not have IP addresses, instead we might have hostnames. To handle such a case we have created a shell script **GetIPFromHostName.sh** to convert a list of hostnames into IP addresses. To do this, copy the hostnames to **HOSTS** variable inside **GetIPFromHostName.sh**. These must be separated by white-space or new line character. Run the script to get the print out of IP address list at the terminal.
2. Put these under a **csv** directory. Put the **csv** directory and **Node_info.txt** file alongside **CreateCSVForComparison.pl** script. Table 3 below provides links to these files.
3. Execute **CreateCSVForComparison.pl** script. This will generate **all-analysis.csv** file containing data in the following format. This will contain all data including **null** value for those targets for which a geolocation technique didn't find any estimate results. The name of each technique represents column of error distance values.
4. Open this **all-analysis.csv** file and convert this to a spreadsheet for analysis.

data format of all-analysis.csv, all values are comma separated

```
serial no, hostname, ip, region, apollonius, cbg_multi, cbg_tri, cbg_with_apollonius, soi, sping, tbg,
tbg_updated, tulip_imp, tulip_old
```

Table 2 below showing list of csv files.

Geolocation technique	File name
Apollonius	apollonius.csv
CBG with multilateration	cbg-multilateration.csv
CBG with trilateration	cbg-trilateration.csv
CBG with Apollonius	cbg-with-apollonius.csv
Speed of Internet (SOI)	soi.csv
SPing	sping.csv
TBG	tbg.csv
TBG Updated	tbg-updated.csv
TULIP trilateration improved	tulip-trilateration-improved.csv
TULIP trilateration old	tulip-trilateration-old.csv

Table 3 below provides links to the files mentioned above.

File	Description
GetIPFromHostName.sh	Script that takes hostnames and converts those to IP addresses. The hostnames list must be copied to HOSTS variable inside the script, each value separated by white-space or new line character. This script outputs a list of IP addresses in the same order as that of hostnames.
csv directory	Contains all the csv files.
Node_info.txt	Contains information such as hostname, IP addresses, Regions, Lat/Longs, etc. for 182 targets.
all-analysis.csv	An amalgamation of all the geolocation techniques and their error distances against the IP addresses and other information. The format is shown above in the box titled "data format of all-analysis.csv".
CreateCSVForComparison.pl	Script that takes csv files from csv directory and Node_info.txt file as inputs and processes out all-analysis.csv file as output.

Known issues

Output file formatting issues:

Once **all-analysis.csv** is generated, don't directly copy it to Windows since there are some formatting issues in such a case. It won't open correctly in Microsoft Excel. So in order to make this right. Open **all-analysis.csv** via vim (on Linux) and copy paste the text into Windows notepad (later save it as **all-analysis.csv**). Once done press **CTRL+H** to find and replace **^M** characters that are read by Linux vim but not by Microsoft Excel and probably therefore causes all sorts of formatting issues.

Input file formatting issues:

Node_info.txt file can have potential formatting issues when copied from Windows to Linux. Instead copy text to clipboard and paste it in a file via vim (on Linux).