

Estimation of Internet Node Location by Latency Measurements – The Underestimation Problem

Dan Komosny

*Brno University of Technology, Brno, Czech Republic
e-mail: komosny@feec.vutbr.cz*

Miroslav Voznak

*VSB - Technical University of Ostrava, Ostrava, Czech Republic
e-mail: miroslav.voznak@vsb.cz*

Kathiravelu Ganeshan, Hira Sathu

*Unitec Institute of Technology, Auckland, New Zealand
e-mail: hsathu,kganeshan@unitec.ac.nz*

crossref <http://dx.doi.org/10.5755/j01.itc.44.3.8353>

Abstract. In this paper we deal with discovering a geographic location of a node in the Internet. Knowledge of location is a fundamental element for many location based applications and web services. We focus on location finding without any assistance of the node being located – client-independent estimation. We estimate a location using communication latency measurements between nodes in the Internet. The latency measured is converted into a geographic distance which is used to derive a location by the multilateration (triangulation) principle. We analyse the latency-to-distance conversion with a consideration of location underestimation which is a product of multilateration failure. We demonstrate that location underestimations do not appear in experimental conditions. However with a real-world scenario, a number of devices cannot be located due to underestimations. Finally, we propose a modification to reduce the number of underestimations in real-world scenarios.

Keywords: geographical; location; geolocation; IP address; Internet; latency; measurement; multilateration; triangulation; PlanetLab.

1. Introduction

Geographic location of Internet nodes is used in many Internet applications, such as on-line credit card fraud detection and prevention [13], web and other services content personalization, spam filtering, distance-based resource allocation [25], emergency VoIP implementation [6], password sharing detection, digital rights management, and any abuse of on-line services [2]. With the Internet of Things that connects small computing devices to the Internet, such as variety of sensor nodes, the knowledge of location of Internet devices becomes even more fundamental [1].

Native or non-native IP resources can be used for client-independent IP Geolocation, i.e. without any assistance of the node being located. Native are based on common features available in IP systems [22]

whereas non-native methods use other resources, such as GPS or radio signal analysis. Non-native approaches are limited in the IP environment¹.

Native approaches are known as passive or active (measurement-based). Passive approaches locate a target (IP node being located) by using various databases which store location data, such as WHOIS or DNSLOC [7]. Other databases map blocks of consecutive IP address spaces to geographic locations. However, databases face a number of limitations such as up-to-datedness problem of the stored location

¹ For example, the client-dependend GPS system can be used for IP Geolocation. However, there is a line-of-sight path requirement from the node being located to four or more satellites. This is not true for the majority of the IP devices.

data due to new IP addresses assignments and relocations.

The active IP Geolocation involves measurement of communication latency. The latency is measured from a set of landmarks with known geographic positions to the target with unknown location². Simple methods map the target's location to the closest landmark in terms of the lowest latency measured, such as ShortestPing [14]. Similarly GeoPing [23] uses the latency values to build the latency-location profiles. It estimates a location of a target by comparing the latency-location profiles. The target's location is mapped to the location of the landmark with the closest location profile.

The latency can be transformed to a geographic distance [18, 20] which is the base for constraint-based Geolocation. It uses the multilateration principle to estimate a location area of a target [21]. When a unique position is required (for example latitude and longitude), the centroid of the specified area is used at the location of the target. Static and dynamic conversions are used for latency-to-distance transformation. The first one is based on the propagation time of digital information. The first assumptions considered that digital information travels in optical cables at $2/3$ of the speed of light in a vacuum [24]. However, the latency is also a product of other factors, such as processing delays or routing policies. Research into this area has led to a definition of a tighter constraint which is $4/9$ of the speed of light in a vacuum. The Speed of Internet (SOI) method [14] uses this constant. In this paper, we use 'SOI $4/9$ ' and 'SOI $2/3$ ' notations for the used fractions of the speed of light in a vacuum.

Dynamic conversion uses calibration. The landmark constructs a list of the measured values as shown in Fig. 1. This figure shows an example of the CBG (Constraint-Based Geolocation) [11]. CBG constructs a line which lies under all the points (latency values from landmarks measured) and touches the closest point at the same time. This line is then used to derive the geographic distances for the latencies measured.

The geographic distances obtained from the described methods are then used as input for the multilateration principle to estimate the target's location. Fig. 2 shows an example of IP Geolocation using the CBG method³.

Other methods such as Octant and Spotter [30, 19] use a similar latency-to-distance calibration based on latency probability models [28, 27], network topology structures, population densities, and city geographic boundaries [3, 5, 19, 4, 15].

² The other way is also possible for client-dependent location: a target can estimate its own location by measuring the latency to the landmarks.

³ In our implementation of IP Geolocation, the great circles with radius over 3500 Km were not used for location and they are not shown in the figure.

In this paper we focus on the location underestimation problem of the active client-independent Geolocation. Location underestimation is a product of multilateration failure when the estimated geographic distances from the landmarks do not intersect to derive a target's location as shown in Fig. 2. This failure is caused by an inaccurate latency-to-distance conversion. We found that around 15 % of the location attempts ended with an underestimation. We identified an underestimation if at least one of the great circles around the landmarks did not delimit the correct area of the target location i.e. the target location was not in the resulted area or the area was null.

The paper is structured as follows: The following section describes the related work with a focus on location underestimations. The section 'Analysis of underestimations' describes our observation that location underestimations happen with dynamic latency-to-distance conversion in a real-world application. We give an example of an underestimation. We also show how underestimations can be fixed. Next we study how the location accuracy is affected by the underestimation fix. Finally, we conclude the paper.

2. Related work

The need for accurate geolocation of IP nodes without client assistance is an important goal in current Internet research. Location accuracy varies a lot depending on the method used. The accuracy of GeoPing, CBG, Learning-Based, and Octant was discussed in the paper [8]. The median of the location error was between 40-160 km. Shavitt and Zilberman [26] evaluated the accuracy and reliability of the Spotter method. At city level (40 km) location accuracy was around 30 %. The accuracy also depended on the position of the landmarks⁴. Considering the country-level granularity, Spotter achieved 85 % accuracy of the correct locations. Eriksson et al. [4] evaluated ShortestPing, GeoPing, CBG, and Octant. The results varied from 30 to 200 km of the median location error.

Location underestimations were analysed in [14]. The authors found that CBG failed to locate 27 of 128 nodes. Their solution was to use 'SOI $2/3$ '. However, this method gives a lower location accuracy. Jayant and Katz-Bassett [12] analysed the underestimation problem of the general landmark-based estimation (LBE). One hundred targets were underestimated due to a wrong calibration using 1411 node pairs. The authors' conclusion and future work was that CBG must be improved to better handle underestimations. They proposed using a different latency-to-distance calibration. Other papers mention the underestimations problem, but they do not present any figures and specific data, such as [11, 10, 9].

⁴ The authors found significant differences for the landmarks in the USA and in Europe.

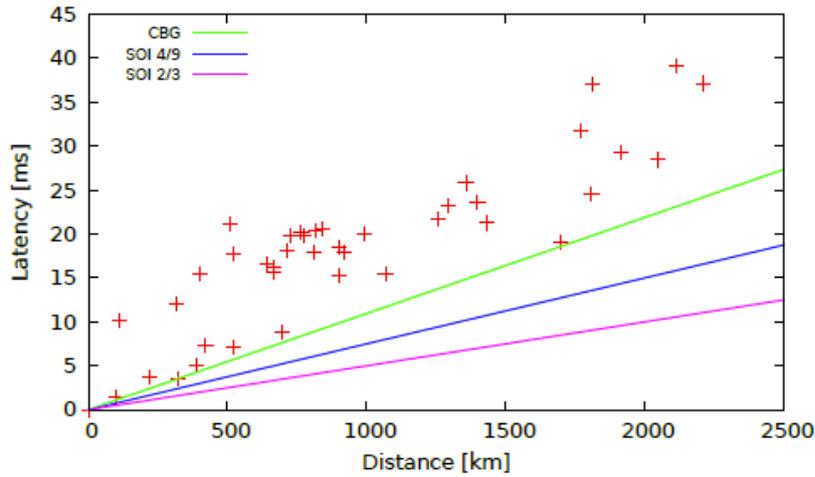


Figure 1. SOI and CBG lines for landmark

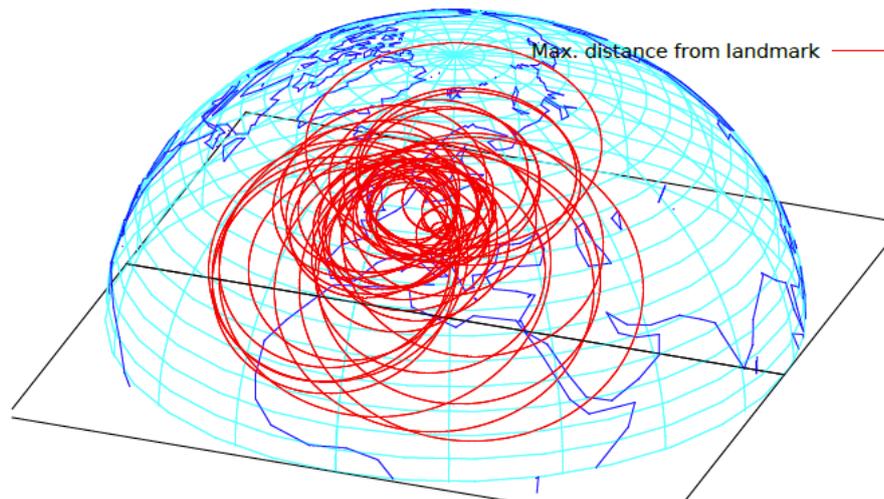


Figure 2. Example location using CBG

3. Analysis of underestimations

In our work we focused on the client-independent geolocation method CBG. CBG is currently recognized as a part of the state-of-the-art IP Geolocation⁵. According to the literature reviewed, the current best active Geolocation approach is described in [29]. It tries to reach street-level location accuracy by employing a three-tier methodology (i) active measurement from geographic landmarks (ii) passive measurement to web servers and (iii) closest node selection. CBG is used in the first-tier to geolocate a target into a specific area.

Our first attempt was to verify the reported accuracy of CBG. For this purpose, we implemented a real-world IP Geolocation system based on the planetary-scale experimental network PlanetLab. PlanetLab is commonly used as the global geographically-distributed testbed [16]. The developed system works with the PlanetLab nodes in Europe. There are over

300 PlanetLab nodes at over 150 sites⁶. We deployed our software developed for this purpose to each of the nodes. Based on our previous research [17], a number of the nodes were not available for the remote access (ssh) which we needed for latency measurement and software deployment. As a result of this, we used 215 nodes in our experiment.

Our research methodology was the following: Firstly, we did not use the PlanetLab nodes as both the landmarks and targets. The reason was not to have the location results negatively influenced by using both the node types from the same network. The location results when using both the landmarks and the targets from the same networks are of significantly better values. Instead, we used targets outside of PlanetLab. For this, we collected 122 nodes acting as targets. We used sources such as the DNSLOC service [7], and NTP (Network Time Protocol) servers. In this way, we created a real-world scenario for IP Geolocation experiments.

⁵ An on-line CBG Geolocation system can be found here – www.wanmon.slab.stanford.edu/cgi-wrappreflex.cgi

⁶ <http://www.planet-lab.eu/>

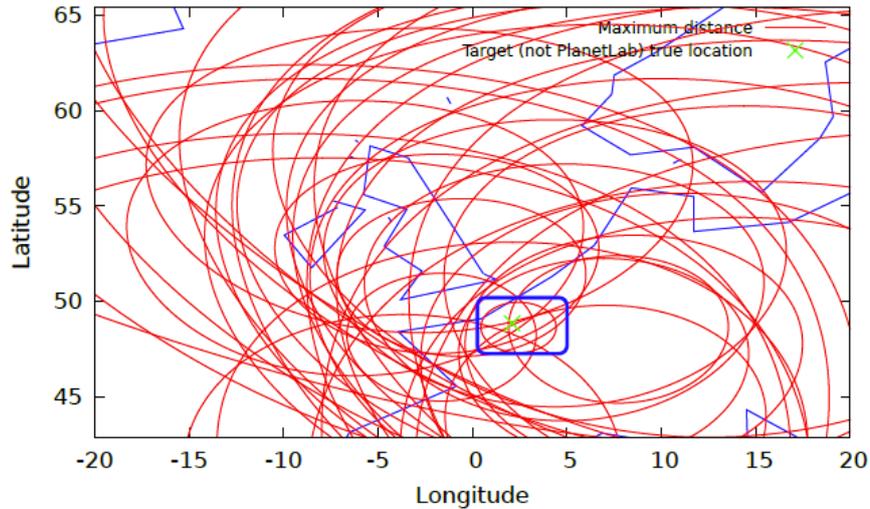


Figure 3. Example of location underestimation using CBG line, TG subset B, target location not estimated

After location of the targets, we noticed that CBG did not work as intended. In a number of cases, about 15 %, we faced a problem of location underestimations. We identified an underestimation if at least one of the great circles around the landmarks did not delimit the correct area of the target location. An example of such underestimation is shown in Fig. 3 for a target in Warsaw, Poland. We analysed this in more detail and tried to prove a hypothesis that if both the landmarks and targets had been from the same network, this issue would not have appeared. To prove this, we created two sets of the targets. The first one, subset A, involved targets belonging to PlanetLab. The second one, subset B, involved nodes not belonging to PlanetLab. Moreover, as our intention was to follow the real-world scenario, we also checked all the positions of the nodes. We left only one PlanetLab

node at a single location. At the same time, we left only one non-PlanetLab node at a single location. The locations of the PlanetLab nodes and non-PlanetLab nodes were distinct. We considered the minimum distance 25 km between all the nodes. By careful selection we got distinct sets of targets and landmarks. This reduced the number of the nodes, but on the other hand, we created as close to real-world scenario as possible since the location results were not influenced by the similar accuracies of the nodes geographically close to each other. We note that without this selection we obtained about a 20 km better accuracy (on average) than using this modified dataset. An overview of the selected nodes (LM – landmark, TG – target) is shown in Table 1. The minimum distances achieved are shown in Table 2.

Table 1. Overview of nodes

Node type	Node count
LM	46
TG all	144
TG set – not same loc, with LM and other TG	59
TG PlanetLab, subset A – not same loc, with LM and other TG	27
TG not PlanetLab, subset B – not same loc, with LM and other	32

Table 2. Minimum distances between nodes

Between nodes	Minimum distance [km]
LM (PlanetLab)	60
TG (PlanetLab)	35
TG (not PlanetLab)	45
LM (PlanetLab) and TG (PlanetLab)	26
LM (PlanetLab) and TG (not PlanetLab)	25

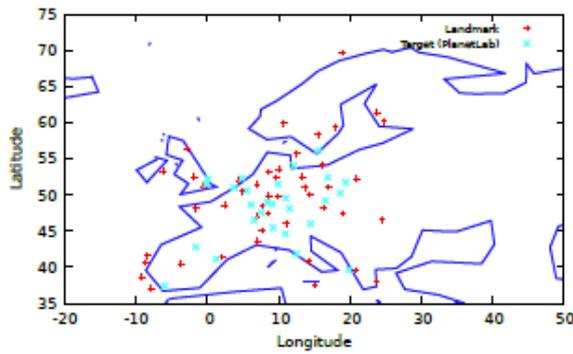


Figure 4. Landmarks and targets, subset A

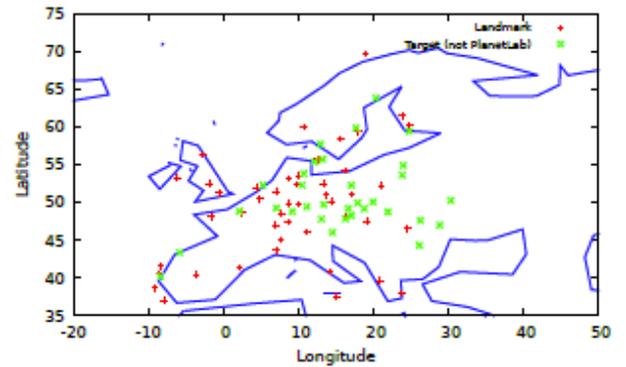


Figure 5. Landmarks and targets, subset B

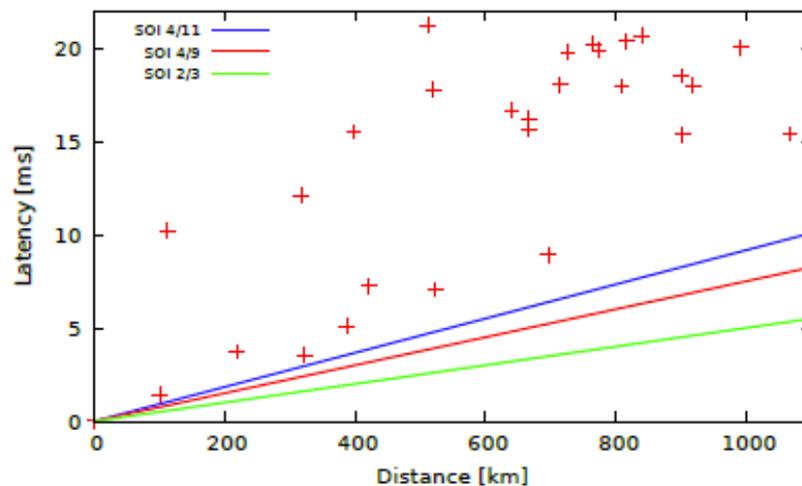


Figure 6. New static calibration line ‘SOI 4/11’ for landmark

The placement of the nodes is shown in Fig. 4 and 5. The figures differ by the targets. The first figure shows the targets belonging to PlanetLab. The targets do not share their position with any landmark and there is only one target at a single location. The second figure shows the targets outside PlanetLab. Again, any target is not close to any landmark and there is only one target at one location.

Using the selected nodes we proved that our hypothesis was correct. With subset A, we located all the nodes without any underestimation. On the other hand, locating targets from subset B resulted in a number of underestimations.

We observed that using a static latency-to-distance conversion reduced the location underestimations in our developed real-world scenario. Both the known static methods, ‘SOI 4/9’ and ‘SOI 2/3’, worked without any underestimation. Moreover, by experiments, we found a new static value of 4/11 of the speed of light which met all our tested criteria. The conversion line ‘SOI 4/11’ is shown in Fig. 6.

Fig. 7 shows a fixed location of the same target which previously failed to be located (Warsaw, Poland – Fig. 3). The figure also shows the estimated location of the target.

Our aim was to limit the number of location underestimations in real-world Internet applications. We fulfilled this by using the ‘SOI 4/11’ method. Next we compare this method to the original methods CBG, ‘SOI 2/3’, and ‘SOI 4/9’ in terms of location accuracy changes. As the real-world configuration, we used the landmarks from PlanetLab and the targets outside PlanetLab. We again limited the maximum estimated distance (great circle radius) to 3500 km. The detailed location accuracy results are shown in Table 3. The table shows that all the methods give similar results. The maximum median difference is 8 km and the maximum average difference is 19 km. However, a first-look assumption is that ‘SOI 4/11’ should give worse results as a penalty for the location underestimation reduction. The ‘SOI 4/11’ line is below the original CBG line and, thus, the latency-to-distance conversion results in greater maximum distances for the latencies measured. This also holds for ‘SOI 2/3’ and ‘SOI 4/9’. We investigated it and found out that this positive result was caused by the targets additionally located using the ‘SOI 4/11’ method (and other SOI methods) compared to the CBG method. The additionally located targets were found with a smaller location error than the average and they improved the final location accuracy.

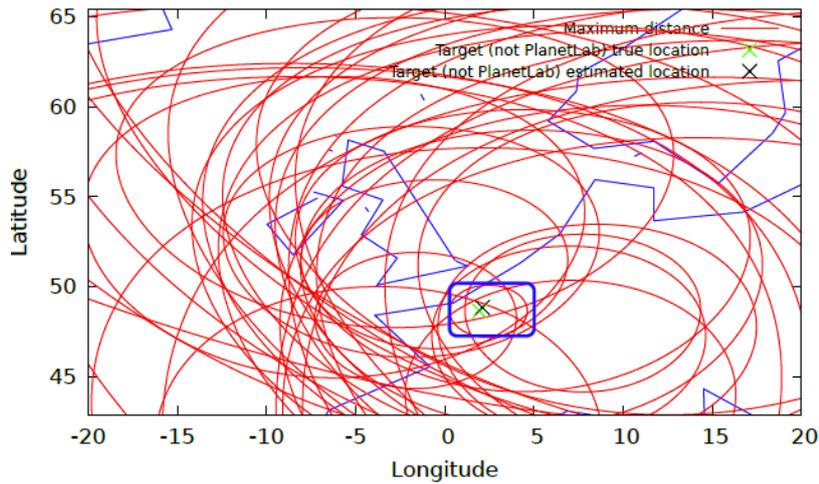


Figure 7. Example of fixed location using line ‘SOI 4/11’, TG subset B, estimated target location shown

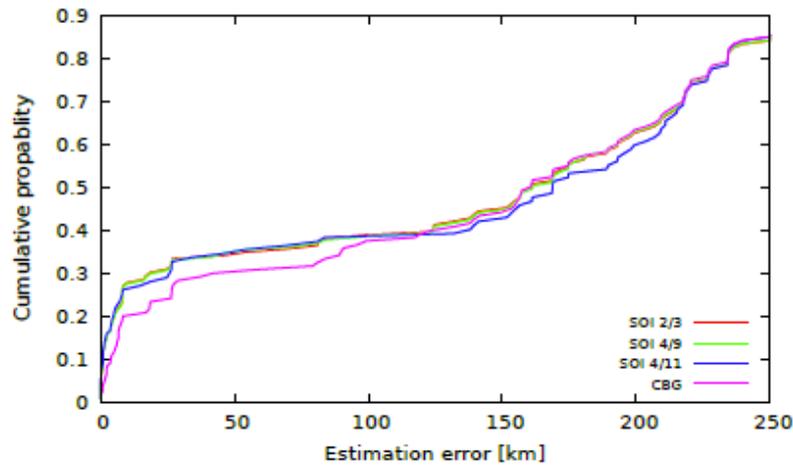


Figure 8. ‘SOI 2/3’, ‘SOI 4/9’, ‘SOI 4/11’, and CBG location accuracy, TG outside PlanetLab

Table 3. Location Accuracy results

Method	1 st quartile [km]	2 nd quartile [km]	3 rd [km]	Mean	Std. Dev.
SOI 2/3	8	161	226	147	127
SOI 4/9	8	161	226	147	126
SOI 4/11	8	168	226	166	219
CBG	26	160	223	166	207

The graphical result of the analysis is shown in Fig. 8. The figure plots a cumulative distribution function of the location accuracy for all the targets used. The function shows the probability of the location error for each method. It can be seen that all the SOI methods, including ‘SOI 4/11’, outperform or equal the CBG method.

4. Conclusion

In this paper, we dealt with the underestimation problem when estimating an Internet node’s position using latency measurements.

We implemented an IP Geolocation system based on the planetary-scale research network PlanetLab. We used 215 PlanetLab nodes as the landmarks and targets, and 122 nodes outside the PlanetLab network as the targets.

We evaluated several scenarios. We located all the targets available, and then we divided the targets into disjunctive sets. We also left the targets which shared the same geographically-close location with another target or landmark. In this way, we observed that location underestimations did not occur when using both landmarks and targets from PlanetLab. We explain this by the fact that the high-capacity research

PlanetLab network gives better latency measurement results than regular commercial networks. On the other hand, when we located the targets outside the PlanetLab network, we observed a number of location underestimations. However, this is the global IP Geolocation scenario.

In order to reduce the location underestimations, we used a static latency-to-distance conversion. We set the slope of the static line as $4/11$ of the speed of light in a vacuum. The number of underestimations was reduced to zero and IP Geolocation accuracy was better than with the CBG method which produced the undesired location underestimations.

Acknowledgement

The research described in this paper was financed by the National Sustainability Program under grant LO1401 and by grant SGS SP2015/82. For the research, infrastructure of the SIX Center was used.

References

- [1] **4G Americas**. Evaluation of Short-Term Interim Techniques for Multi-Media Emergency Services. White paper, 2011.
- [2] **V. Alarcon-Aquino, J. M. Ramirez-Cortes, P. Gomez-Gil, O. Starostenko, Y. Garcia-Gonzalez**. Network Intrusion Detection Using Self-Recurrent Wavelet Neural Network with Multidimensional Radial Wavelons. *Information Technology and Control*, 2014, Vol. 43, No. 4, 347–358.
- [3] **M. J. Arif, S. Karunasekera, S. Kulkarni**. GeoWeight: internet host geolocation based on a probability model for latency measurements. In: *Proceedings of the Thirty-Third Australasian Conference on Computer Science*, Australian Computer Society, 2010, Vol. 102, pp. 89–98.
- [4] **B. Eriksson, P. Barford, B. Maggs, R. Nowak**. Posit: An Adaptive Framework for Lightweight IP Geolocation. BU/CS Technical Report, Boston University, 2011.
- [5] **B. Eriksson, P. Barford, J. Sommers, R. Nowak**. A learning-based approach for IP geolocation. In: *Proceedings of the 11th international conference on Passive and active measurement*, 2010, pp. 171–180.
- [6] **Federal Communications Commission**. Interconnected VoIP Service; Wireless E911 Location Accuracy Requirements; E911 Requirements for IP-Enabled Service Providers. FR Doc No: 2011-24865, 2011.
- [7] **T. Fioreze, G. Heijenk**. Extending DNS to support geocasting towards VANETs: A proposal. In: *Vehicular Networking Conference (VNC)*, IEEE, 2010, pp. 271–277.
- [8] **P. Gill, Y. Ganjali, B. Wong, D. Lie**. Dude, Where’s That IP? Circumventing Measurement-based IP Geolocation. In: *Proceedings of the 19th USENIX Security Symposium*, USENIX Association Berkeley, 2010, pp. 241–256.
- [9] **M. Gondree, Z. Peterson**. Geolocation of Data in the Cloud. In: *Proceedings of the Third ACM Conference on Data and Application Security and Privacy*, ACM, 2013, pp. 25–36.
- [10] **B. Gueye, S. Uhlig, A. Ziviani, S. Fdida**. Leveraging Buffering Delay Estimation for Geolocation of Internet Hosts. In: *Proceedings of the 5th International IFIP-TC6 Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, Springer-Verlag, 2006, pp. 319–330.
- [11] **B. Gueye, A. Ziviani, M. Crovella, S. Fdida**. Constraint-based geolocation of internet hosts. *IEEE/ACM Transactions on Networking*, 2006, Vol. 14, No. 6, 1219–1232.
- [12] **R. Jayant, E. Katz-basnett**. Toward Better Geolocation: Improving Internet Distance Estimates Using Route Traces. Report, The Pennsylvania State University, 2004.
- [13] **Q. Jiang, J. Ma, G. Li, Z. Ma**. An Improved Password-Based Remote User Authentication Protocol without Smart Cards. *Information Technology and Control*, 2013, Vol. 42, No. 2, 113–123, 2013.
- [14] **E. Katz-Basnett, P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, Y. Chawathe**. Towards IP geolocation using delay and topology measurements. In: *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, ACM, 2006, pp. 71–84.
- [15] **D. Komosny, J. Balej, H. Sathu, R. Shukla, P. Dolezel, M. Simek**. Cable Length Based Geolocation. *Przeglad Elektrotechniczny*, 2012 (7a): 26–32, 2012.
- [16] **D. Komosny, J. Pruzinsky, P. Ilko, J. Polasek, K. Oguzhan**. On Geographic Coordinates of PlanetLab Europe. In: *37th International Conference on Telecommunications and Signal Processing (TSP)*, 2014, pp. 229–233.
- [17] **D. Komosny, M. Simek, G. Kathiravelu**. Can Vivaldi Help in IP Geolocation? *Przeglad Elektrotechniczny*, 2013, Vol. 2013, No. 5, 100–106.
- [18] **O. Krajsa, L. Fojtova**. RTT measurement and its dependence on the real geographical distance. In: *Proceedings of 34th International Conference on Telecommunications and Signal Processing*, 2011, pp. 231–234.
- [19] **S. Laki, P. Matray, P. Haga, T. Sebok, I. Csabai, G. Vattay**. Spotter: A model based active geolocation service. In: *IEEE Conference on Computer Communications*, IEEE, 2011, 3173–3181.
- [20] **P. Moravek, D. Komosny, R. Burget, J. Sveda, T. Handl, L. Jarosova**. Study and performance of localization methods in IP based networks: Vivaldi algorithm. *Journal of Network and Computer Applications*, 2011, Vol. 34, No. 1, 351–367.
- [21] **P. Moravek, D. Komosny, M. Simek**. Multilateration and Flip Ambiguity Mitigation in Ad-hoc Networks. *Przeglad Elektrotechniczny*, 2012, Vol. 2012, No. 05b, 222–229.
- [22] **P. Moravek, D. Komosny, M. Simek, J. Sveda, T. Handl**. Vivaldi and other localization methods. In: *Proceedings of the International Conference on Telecommunications and Signal Processing (TSP)*, Brno University of Technology, 2009, pp. 214–218.
- [23] **V. Padmanabhan, L. Subramanian**. An investigation of geographic mapping techniques for internet hosts. *SIGCOMM Comput. Commun. Rev.*, 2001, Vol. 31, No. 4, 173–185.

- [24] **R. Percacci, A. Vespignani.** Scale-free behavior of the Internet global performance. *The European Physical Journal B - Condensed Matter and Complex Systems*, 2003, Vol. 32, No. 4, 411–414.
- [25] **S. Ch. Shah, M. S. Park, W. S. Choi, Z. H. Mir, S. H. Chauhdary, A. K. Bashir, F. H. Chandio.** An Adaptive Distance-based Resource Allocation Scheme for Interdependent Tasks in Mobile Ad Hoc Computational Grids. *Information Technology and Control*, 2012, Vol. 41, 4, 307–317.
- [26] **Y. Shavitt, N. Zilberman.** A Geolocation Databases Study. *IEEE Journal on Selected Areas in Communications*, 2011, Vol. 29, No. 10, 2044–2056.
- [27] **M. Voznak, M. Halas, F. Rezac, L. Kapicak.** Delay variation model for RTP flows in network with priority queueing. In: *10th WSEAS International Conference on EHAC'11 and ISPra'11*, WSEAS, 2011, pp. 344–349.
- [28] **M. Voznak, J. Rozhon, H. A. Ilgin.** Network delay variation model consisting of sources with Poisson's probability distribution. In: *Proceedings of the 15th WSEAS International Conference on Computers*, WSEAS, 2011, pp. 240–244.
- [29] **Y. Wang, D. Burgener, M. Flores, A. Kuzmanovic, C. Huang.** Towards street-level client-independent IP geolocation. In: *Proceedings of the 8th USENIX conference on Networked systems design and implementation*, USENIX Association, 2011, pp. 27–27.
- [30] **B. Wong, I. Stoyanov, E. Sirer.** Octant: a comprehensive framework for the geolocalization of internet hosts. In: *Proceedings of the 4th USENIX conference on Networked systems design & implementation*, USENIX Association, 2007, pp. 313–326.

Received September 2014.