Online Self-Calibration of the Propagation Model for Indoor Positioning Ranging Methods

Grigorios G. Anagnostopoulos, Michel Deriaz and Dimitri Konstantas

Information Science Institute, GSEM/CUI
University of Geneva
Geneva, Switzerland
Email: $\{grigorios.anagnostopoulos, michel.deriaz, dimitri.konstantas\}$@unige.ch

***Abstract*—A common problem for indoor positioning methods is the fact that the differences in the reception characteristics among devices may significantly deteriorate the performance of a positioning system. Ranging algorithms for positioning rely on the accuracy of the parameters of the propagation model. This model is used to infer an estimate of the distance of a mobile device from each access point from the Received Signal Strength Indication (RSSI). In this work we present an algorithm which dynamically recalculates and improves the propagation model. The improvement of the model parameters fits the environment's characteristics and, more importantly, the reception characteristics of the device used. The proposed algorithm is tested with different devices at an indoor deployment covering a large area where Bluetooth Low Energy (BLE) technology is used. The experimental results show that the proposed method offers a significant accuracy improvement to some devices while it slightly improves the performance of those that are more properly tuned.**

***Keywords*—Indoor Positioning, Localisation, Bluetooth, RSSI, Propagation Model, Device Independence**

I. Introduction

The interest in the field of indoor positioning has risen over the last years. The broad use of smartphones has familiarized the public with Location-Based Services (LBS). Most users of smart devices have had the experience of being positioned outdoors, with the use of Global Navigation Satellite Systems (GNSS) like the Global Positioning System (GPS). Indoor positioning is following up, without having offered so far a generic solution as the GPS.

Several technologies are being used in order to localize users indoors. A technology commonly used for positioning in indoor environments is Wi-Fi [1],[2]. A main advantage of Wi-Fi is that most buildings have several Wi-Fi Access Points (AP) in order to provide internet access, so the required hardware is already installed. On the other hand, this may also be a disadvantage, because the way the access points were placed may be convenient for network coverage, but not optimal for positioning. Another restriction is that the Wi-Fi APs must be plugged, a fact that may restrict the freedom in the way they can be placed.

A technology which has been widely used during the last years is the BLE technology. Bluetooth beacons function with batteries and are small in size, thus they offer flexibility in the way they can be deployed in a building.

One of the most common positioning techniques is the utilization of the Received Signal Strength (RSS). RSS fingerprinting is a popular approach, which requires an off-line phase in which a radiomap of the localization area needs to be created. During this procedure, a reference device is used to record fingerprints of specific locations. The fingerprint of a location is the set of RSS receptions of each access point at said location. At the online phase, the receptions from all access points are compared with the fingerprints. An estimated position is inferred based on a similarity measure between the receptions at a certain time, and the recorded fingerprints.

Another way of utilizing the RSS is by inferring a distance estimate. Having a distance estimate from each access point, a position estimate can be calculated, with the use of ranging methods like multilateration [1] or weighted centroid [3]. The exact location of the access points is needed for these methods, along with a propagation model that corresponds the RSSs to distances. An advantage of these methods is that they do not require an off-line surveying phase, which can be very time consuming. Another advantage is that a potential displacement of an AP or the addition of new ones, brings only the obligation of adding the new APs' positions in the list of known APs, whereas in RSS fingerprinting such a change would require the repetition of the off-line phase.

A common problem for both fingerprinting and ranging methods that rely on signal strength is that they suffer from fluctuations of RSS receptions. Moreover, ranging methods use a propagation model whose parameters rely on many factors, such as the environment and the devices used. Subsequently, a specific setting of the propagation model's parameters may be adequate for a reference device with which the positioning system is tested, but it may also be faulty for another device with different reception characteristics.

In this work, we present a novel approach on recalculating the propagation model parameters on-line, in ranging positioning methods. In this way, a device, whose reception charac-

This work was co-funded by the State Secretariat for Education, Research and Innovation of the Swiss federal government and the European Union, in the frame of the EU AAL project SmartHeat (aal-2014-153).

teristics differ from the ones used to create the propagation model, is using the initial range estimates along with the position estimates to gradually correct the propagation model. The experimental testing of the method is very promising, since an accuracy improvement above 13% was achieved with a device different from the one used for the creation of the propagation model. It is noteworthy that not only several devices were used for the test, but also the beacons used and the environment of the deployment where the method was tested were different from those that were used for the initial propagation model calculation. The performance of the device used to create the propagation model also showed slight improvements. The presented method contributes to the recurrent research with the goal of device independence in indoor positioning systems. It also makes it possible to skip the propagation model calculation when a system is deployed at a new environment, with a different kind of beacons.

The rest of this paper is organized as follows. After commenting the related work in Section II, and introducing some preliminaries in Section III, in Section IV the proposed method is extensively presented. In Section V, we analyse and discuss the results of the experimental measurements from a real deployment. Lastly, drawn conclusions along with future directions are discussed in Section VI.

II. Related Work

During the last years, the research towards propagation model correction and device independence for indoor positioning techniques has blossomed. Many studies have focused on this goal for both surveying (fingerprinting) and ranging positioning techniques.

An early work in this domain by Mazuelas et al. [1] tunes the path loss exponent characterizing each AP on a wireless local area network (WLAN). The motivation of that work is mainly directed towards skipping the propagation model calibration step and not device independence per se. The tuning is done by finding the set of path loss exponent values that solve a least square optimization problem concerning the distance of the estimated position from the radical axes of the range estimates. The proposed solution is elegant and appealing, thought quite complex, and also relies on a rather strong assumption when working with several devices, that parameter p (received RSS at a reference distance) can be considered a constant.

In another study [4], the authors adjust the propagation model by utilizing contextual knowledge of where the walls are in a building, in order to adjust the model's parameters of each AP. They do so by introducing an attenuation factor for every wall between said AP and the estimated position, which requires a procedure of defining the areas where walls exist. Recent studies [5], [6], battle the problem of RSS-based localization when the channel parameters are considered unknown, providing however only simulations and no experimental results.

Extensive literature exists ([7], [8], [9], [10], [11], [12], [13] and references therein) dealing with the impact of device diversity on fingerprint positioning. Inspiring solutions have been proposed by these studies, including among others, signal strength histogram equalization [7], [8], spatial mean normalization [9], signal strength ratio utilization [10], differential fingerprinting [11], fusing of crowd-sourced RSS data into usable radiomaps of differential fingerprints [12], and ranking of RSS values from a set of APs from high to low [13], since ranking is device independent.

From the volume of the relevant work of this rather recent field, it is evident that methods of automatic self-calibration or re-calibration as well as calibration-free methods are a trending topic. Before proceeding to the presentation of the self-calibration method, the explanation of some preliminary elements is in order.

III. Preliminaries

A. Propagation Model

For ranging indoor positioning techniques, a distance estimation from the APs is necessary. More specifically, when using RSS methods, a propagation model is used to infer a distance estimate from a value of the Received Signal Strength Indicator (RSSI). The propagation model commonly used for indoor positioning is the log-distance path loss model, presented in Equation 1. The propagation model, with which the expected received power p_i in distance d_i is calculated, is characterized as:

$$p_i = p - 10\, n\, log_{10}(d_i/d_0) + X \tag{1}$$

In this formula, p is the received RSSI at a reference distance d_0, and n is the path loss exponent which depends on the transmission channel. The path loss exponent n can be considered to be also influenced by the way the transmitter and the receiver are made (for example, different device packaging materials alter the channel) or placed (since the transmission is not uniform towards all directions). Theoretically, $n = 2$ for no attenuation in power, whereas in actual indoor deployments values of $n > 2$ better describe the power loss over distances, while values $n < 2$ are suitable if the signal is enhanced by the environment. Lastly, X is a random noise, which is assumed to have a Gaussian zero-mean distribution.

In order to translate an RSSI reception to a distance estimate, the parameters p and n need to be defined. Usually, the determination of these parameters requires a calibration step. This calibration can be completed with a straightforward procedure: several receptions are recorded at predefined distances from the emitting beacon. The best fitting curve describing these measurements, obtained by regression, provides the p and n values that optimally describe the RSSI-distance relation in Equation 1. The accuracy of this method can be increased by using a big amount of measurements. This is because the noise is assumed to have a Gaussian zero-mean distribution, and thus, using a plethora of measurements for each predefined distance increases the probability of reducing the average error.

However, this method has some clear limitations. Firstly, the parameters inferred rely on the environment of the calibration.

Performing the calibration with one beacon at a certain location, and using the obtained parameters for all the beacons of the same type that are placed at a deployment may be handy, but does not describe the particularities of the environment around each beacon, like possible reflections and Non-Line Of Sight (NLOS) cases. On the other hand, calibrating the propagation model of each beacon after it is placed may require a big effort and it can be a great restriction for big deployments with many beacons. Secondly, apart from the transmitter (the beacon) and the channel (the environment between the beacon and the area that users move), the parameters also depend on the receiver's (the mobile device's) characteristics. If the calibration procedure is repeated with other mobile devices, with different reception characteristics, the resulting estimates of the propagation model parameters will differ among them. Thus, using a propagation model inferred by a single device introduces an inherent error when the model is used by another device.

B. Positioning Algorithms Used

With the use of the propagation model, a distance estimate can be inferred from the RSSI received from each beacon. Having obtained an estimation about the distance of the mobile device from each beacon, we proceed to the position estimation. In this work, we use two positioning algorithms, the weighted centroid algorithm [3], and a multilateration algorithm. In both algorithms, only the N beacons that are detected to be closer to the mobile device are used. In [3], it is shown that keeping the $N = 4$ closest beacons minimizes the expected estimation error, and thus this value is used in the experiments of this work.

The weighted centroid algorithm is a simple, straightforward positioning method, with very low computational complexity. Let d_i be the estimated distance from the i_{th} closest beacon, and (x_i, y_i) the beacon's position. The estimated position is given as the weighted centroid of the positions of the $N = 4$ closest beacons, using as weight $w_i = 1/d_1$ the inverse of the distance estimate from this beacon.

$$x^{est} = \frac{\sum_{i=1}^{N} \frac{x_i}{d_i}}{\sum_{i=1}^{N} \frac{1}{d_i}} \; , \; y^{est} = \frac{\sum_{i=1}^{N} \frac{y_i}{d_i}}{\sum_{i=1}^{N} \frac{1}{d_i}} \tag{2}$$

A notable property of this method is that it restricts the position prediction inside the area that the beacons' positions define, making it impossible to give a position estimate outside this area. Thus, it is indispensable for this positioning algorithm that beacons are placed in a way so that they surround all the area that is desired to be covered. When working with a badly tuned propagation model, limiting the estimates inside the area of the surrounding beacons can be a desirable property, for limiting the area of the potential positioning error.

The other approach is the multilateration method. Four circles are drawn, using each beacon's position (x_i, y_i) as the center, and the corresponding estimated distance d_i from the i^{th} beacon as radius. If the distance estimates are correct, all circles will intersect at a single point. As the distance estimates are subject to noise, a single intersection is unlikely. Therefore, all the crossing points of each pair of circles are stored. In case the circles do not intersect, we store the points of the circles which are crossed by the line connecting the centres of the two cycles. We characterize each crossing point with a weight equal to the inverse value of the smallest radius of the two circles in which the point belongs. Then, the weighted centroid of all the stored intersection points, is the estimated position.

IV. Self-Calibration Method

A. Presentation of Self-Calibration Method

The proposed self-calibration method acts dynamically, as the user moves and receives position estimates. Let M be the path loss model used (defined in Equation 1), characterized by the two parameters:

$$M = [p, n] \tag{3}$$

Initially, a set of default values of the parameters p and n is used. The position estimates received are used for gradually adjusting the values of the parameters. Before proceeding to details, some definitions are in place. Let

$$d^{est} = [d_1^{est}, d_2^{est}, \ldots, d_N^{est}] \tag{4}$$

be the set of distance estimates from each of the N closest beacons used for positioning, as inferred by using the propagation model defined in Equation 1. These estimated distances are used in order to infer a position estimate (x^{est}, y^{est}). Furthermore, let

$$d^{pos} = [d_1^{pos}, d_2^{pos}, \ldots, d_N^{pos}] \tag{5}$$

be the set of the distances of the position estimate (x^{est}, y^{est}) from each of the N closest beacons. Ideally, d^{est} and d^{pos} should be the same, but due to noise and the inaccuracy of the model used, they tend to differ. The method consists of two steps that are performed after each position estimation: an optimization step, and an update step.

At the *first optimization step*, the optimization problem presented in Equation 6 needs to be solved for each of the N closest scanned beacons.

$$M^* = \arg\min_{M} \left| d^{pos} - d^{est}_{(M)} \right| \tag{6}$$

The model M^* resulting from this step is the one that minimizes the difference between the two distances: d^{pos}, that is the distance from the beacon to the estimated position as calculated before the optimization step, and $d^{est}_{(M)}$, that is the estimated distance of the mobile device from the beacon as inferred from the received RSSI by using the model in search M, that is the tunable argument. We will refer to M^* as the optimal model for the consistency of the latest reception.

Since the algorithm updates the parameters of the model at each reception, we will refer to the state of the model at time t as $M[t]$.

At the *second updating step*, the current model $M[t]$ is updated with the result of the optimization step ($M^*[t]$) based on the latest reception, providing the model $M[t+1]$ to be used at the next reception, at time $t+1$. The update is made with the following logic.

$$M[t+1] = \alpha M^*[t] + (1-\alpha)M[t] \quad , \quad \alpha \in [0,1] \qquad (7)$$

The update rate α in Equation 7 determines the level of influence of the optimal model for the consistency of the latest reception $M^*[t]$ in updating the model used for the next step $M[t+1]$. The tuning of the update rate, as well as other issues regarding the optimal setting of the self-calibration method are discussed in the following subsection.

B. Tuning of the Self-Calibration Method

The proposed method has several settings that need to be tuned. One of the challenges in finding the best settings for this method is selecting the optimization algorithm for the first step (Equation 6). One option would be to perform a Brute Force search at the space of possible solutions (all acceptable values of the parameters p and n of the model M). This option has the evident drawback of being computationally expensive. Thus, the alternative of using a heuristic algorithm could reduce the computational cost. Furthermore, the method starts with a set of default values for the model's parameters, so there exists a meaningful starting point for a local search algorithm, like Hill-Climbing. In addition, the parameters in question are expected to be relatively close to their default values rather than at the limits of the search space. For this reason, a local search solution is more likely to avoid some extreme values at the limits of the search space that could be provided due to a potential noisy reception. The efficiency and the computational cost of these two approaches are discussed in Section V.

Regarding the optimization algorithm, there are some important parameters to be set. Initially, the limits of the search space should be defined ($n_{MIN} \leq n \leq n_{MAX}$, $p_{MIN} \leq p \leq p_{MAX}$). Furthermore, the granularity of the search algorithm should be defined, for both dimensions of search (p and n). Let p_{step} and n_{step} be the step size for each respective dimension.

Lastly, a crucial decision to be taken is the value of the update rate α. Choosing a large value for α, which would be close to 1, would mean that there is a big danger of overfitting the model in cases of noisy receptions. Using this completely wrong model at a following positioning step could deteriorate the quality of the position estimations, and eventually lead to diverging from the optimal model. On the other hand, a small value of α, close to 0, offers the opportunity of reducing the influence of occasional outliers, or even evening out their effect smoothly. A low update rate however should be carefully chosen in a way that allows the model to change with a speed that is sufficient in order to improve the overall positioning system.

The following Section contains a detailed practical examination of all the above presented aspects of the self-calibration method.

V. Results of Experimental Measurments

For the evaluation of the proposed self-calibration method, measurements from an actual deployment of a positioning system were used. A broad area (120 * 40 m) of an underground parking was used as the test environment (left half part of Figure 1). In this area, 40 BLE beacons were placed at a rhombus grid pattern. The path shown in Figure 1 was followed by a user holding three different mobile devices: a Samsung Galaxy S5, a Samsung Galaxy S4, and a Samsung Galaxy Note 3. During this path, each device recorded all the necessary information of the received signals (RSSI, timestamp, beacon ID) in order to later run the positioning algorithm off-line. Using the recorded raw data (RSSI, timestamp, beacon ID), position estimates can be calculated off-line either with or without the proposed self-calibration method. It was chosen to record and use the raw data in order to have a consistent comparison over the same dataset.

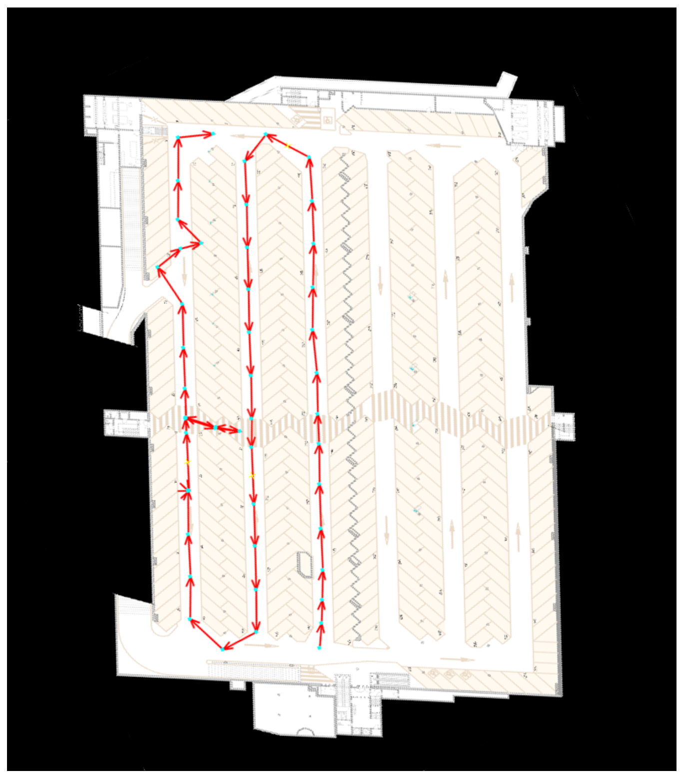

Fig. 1. The path followed in the parking. The area where coverage is provided is the left half side of the parking, and its dimensions are 120 by 40 m.

Moreover, the user holding the devices was informing the recording application at every moment that he was passing by each one the 50 predefined checkpoints of the path (the nodes of the path in Figure 1). Thus, not only signal receptions have their exact timestamps, but also the 50 checkpoints of the path are linked to the exact time that the user was there. In this way, assuming that the user was moving at a steady pace between two consecutive checkpoints, the real position (ground truth) of the user can be inferred at any moment during the path. Consequently, after obtaining the position estimates from the raw data by a positioning algorithm, the error of each position estimate at any moment can be precisely calculated.

It was chosen to use as default values of the model, the values of an old deployment, in a different environment (corridors of the University of Geneva) with a different brand of beacons ('tod' beacons [14]) than those of the current deployment at the parking ('kontakt' beacons [15]), that were estimated using a Samsung Galaxy S4. These default values were calibrated to be p=-62.72 and n=2.28. Also, regarding the tuning of the parameters presented in Section IV-B, the following values were chosen empirically: the values n_{MIN}=1, n_{MAX}=3, p_{MIN}=-45 and p_{MAX}=-75 set the limits of the search space, while p_{step}=1 and n_{step}=0.02 determine the step size of the search.

In Figure 2, we see the statistics of the achieved positioning accuracy with the three devices (Samsung Galaxy S5, S4, Note 3), for several values of the update rate α. In Figure 3, the box plot characterizing the accuracy of one device (Samsung Galaxy S5) highlights the median value of the positioning error with the red line inside each box. The limits of each box represent the 25th and the 75th percentile of the error. Apart from the box plot, the black stars connected with a black line show the mean value of the error of each case. The positioning algorithm used for these tests was the weighted centroid algorithm. Similar results were achieved also with the multilateration approach.

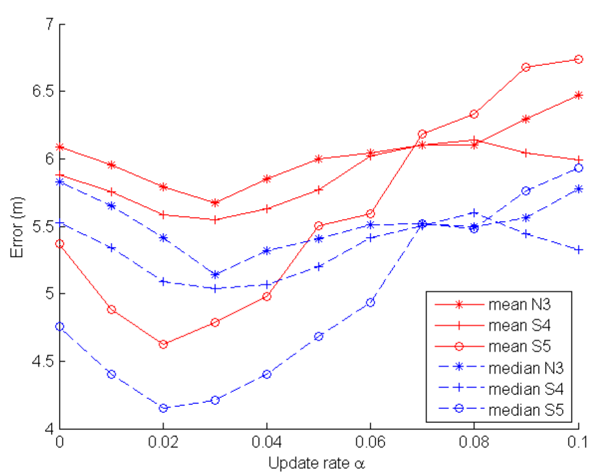

Fig. 2. Mean and median values of the positioning error of the three devices used, for several values of the update rate α.

For α=0, the statistics concern the pure positioning algorithm without the self-calibration method. For low values of α in the range $0.01 \leq \alpha \leq 0.04$, we see that all devices achieved an improvement in accuracy. In Figure 2, we see that the S5 had the greatest improvement in accuracy among

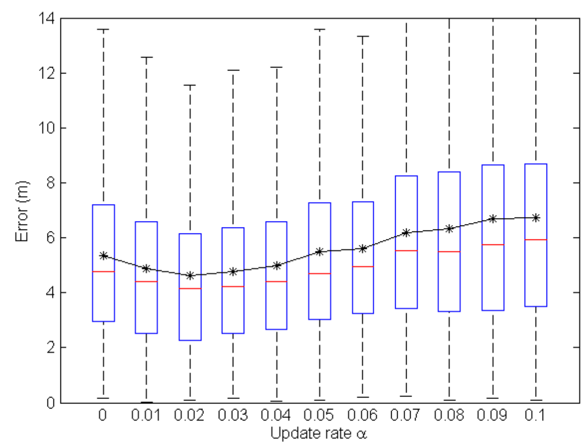

Fig. 3. Box plot and mean values (black line) of the positioning error for several values of the update rate α, using a Samsung Galaxy S5.

the devices, going from a mean value of 5.37 m for $\alpha = 0$ to 4.62 for $\alpha = 0.02$ (13.88% improvement). For greater values of α, it is evident that the accuracy degrades. In addition, the other two devices have a less impressive but still satisfactory performance. Both S4 and Note 3 achieved a $\sim$5% improvement for $\alpha = 0.02$.

An important desired feature for the self-calibration method is that not only it improves the accuracy when this is possible, but also that it does not degrade the system's performance when the conditions (high noise, very imprecise default model) do not allow the convergence to a more appropriate model. After performing multiple tests, we have concluded empirically that a value of $\alpha = 0.02$ handles a satisfactory trade-off of these two attributes.

In Table I, the mean error before and after introducing the self-calibration is presented, for several pairs of initial values of the propagation model parameters. The data collected with the S5 are used. The first value of each cell is the mean error of pure positioning, while the second one is the one using the self-calibration method with $\alpha = 0.02$. Lastly, the percentage change of the mean error appears in the brackets.

TABLE I
MEAN ERROR BEFORE AND AFTER SELF-CALIBRATION

p	n=2	n=2.3	n=2.6
-50	9.78 - 9.74 (-0.4%)	8.41 - 7.91 (-5.9%)	6.78 - 6.04 (-10%)
-55	8.30 - 7.65 (-7.8%)	6.68 - 6.05 (-9.4%)	5.23 - 5.14 (-1.7%)
-60	6.56 - 5.51 (-16%)	5.19 - 4.77 (-8.0%)	5.36 - 4.72 (-12%)
-65	5.26 - 4.73 (-10%)	5.34 - 4.92 (-7.8%)	5.36 - 5.37 (+0.1%)
-70	5.31 - 5.14 (-3.2%)	5.34 - 5.82 (+8.9%)	5.36 - 6.43 (+19%)

Having a very bad initial estimation of the propagation model parameters, far from the optimal one, significantly deteriorates the position estimates, which are used in the self-calibration method. Thus, in these cases of bad initial estimations, as for example with (p, n)=$(-50, 2)$ or with (p, n)=$(-65, 2.6)$, we observe that the self-calibration method has similar performance to the calibration-free case. In most of the cases of Table I the method improves the average performance. Only at the two extreme cases, for $(p, n)=(-70, 2.3)$ and $(p, n)=(-70, 2.6)$, the low quality of the initial estimations does not allow the method to improve the accuracy, but also results in a drop of the achieved accuracy.

In Figure 4, accuracy statistics of the two positioning methods are reported. Both multilateration and weighted centroid (which was used for the previous experiments) achieve similar patterns of improvement in accuracy for the presented values of α.

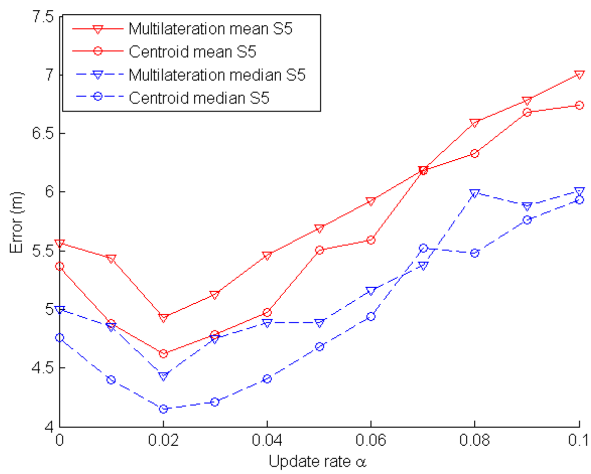

Fig. 4. Mean and median values of the positioning error using multilateration and weighted centroid, for several values of the update rate α, using a Samsung Galaxy S5.

Lastly, it is worth comparing the two algorithms used for the optimization step, Brute Force and Hill-Climbing. Both algorithms have a very similar performance for all values of the update rate α. Similarly to Hill-Climbing, Brute Force has the best performance for low values of the update rate and more specifically, for α=0.02. In terms of computational effort though, Hill Climbing clearly outperforms Brute Force, as expected. The average time for estimating the positions of the path of Figure 1 over 1000 repetitions was 82 ms for Hill Climbing and 342 ms for the Brute Force search. The tests were done off-line using the recorded data. Nevertheless, since the algorithm is to be used on-line, on mobile devices, it is indispensable to minimize the computational effort, especially when this is possible without any impact on the performance.

VI. Conclusions and Future Work

A simple and effective algorithm for automatic self-calibration of the propagation model in ranging positioning techniques is introduced in this work. At every new deployment of a positioning system, a potential calibration procedure that tries to model the propagation characteristics, apart from being time consuming, cannot predict the reception characteristics of any mobile device that could use the positioning system. To cope with this issue, the proposed method offers a robust way of correcting the propagation model. It offers a significant improvement to some devices (13.88% improvement of mean error for the Samsung Galaxy S5) while it slightly improves the performance of those that are more properly tuned (a $\sim$5% improvement for the Samsung Galaxy S4, and the Samsung Galaxy Note 3). The proposed method can be seen in scope of the research towards device independence, as well as in the context of facilitating calibration-free deployment at new areas.

As future work, we plan to perform extensive tests, in search for the optimal tuning of the numerous parameters that play a role in this method. We firstly intent to gather a big amount of data, in the same way that we recorded with the three devices the data used for this study. Having the ground truth and the raw signals from numerous paths at several deployments, we believe that we can obtain a strong certainty about the optimal tuning of the method.

References

[1] S. Mazuelas, A. Bahillo, R. Lorenzo, P. Fernandez, F. Lago, E. Garcia, J. Blas, and E. Abril, "Robust indoor positioning provided by real-time rssi values in unmodified wlan networks," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 3, no. 5, pp. 821–831, Oct 2009.

[2] M. Lee and D. Han, "Voronoi tessellation based interpolation method for wi-fi radio map construction." *IEEE Communications Letters*, vol. 16, no. 3, pp. 404–407, 2012.

[3] G. G. Anagnostopoulos and M. Deriaz, "Accuracy enhancements in indoor localization with the weighted average technique," in *SENSORCOMM 2014*, 2014, pp. 112 – 116.

[4] P. Barsocchi, S. Chessa, E. Ferro, F. Furfari, and F. Potorti, "Context driven enhancement of rss-based localization systems," in *Computers and Communications (ISCC), 2011 IEEE Symposium on*, June 2011, pp. 463–468.

[5] Y. Xu, J. Zhou, and P. Zhang, "Rss-based source localization when path-loss model parameters are unknown," *Communications Letters, IEEE*, vol. 18, no. 6, pp. 1055–1058, June 2014.

[6] M. Gholami, R. Vaghefi, and E. Strom, "Rss-based sensor localization in the presence of unknown channel parameters," *Signal Processing, IEEE Transactions on*, vol. 61, no. 15, pp. 3752–3759, Aug 2013.

[7] C. Laoudias, R. Piche, and C. Panayiotou, "Device signal strength self-calibration using histograms," in *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, Nov 2012, pp. 1–8.

[8] C. Laoudias, R. Piche, and C. G. Panayiotou, "Device self-calibration in location systems using signal strength histograms." *J. Location Based Services*, vol. 7, no. 3, pp. 165–181, 2013.

[9] C.-H. Wang, T.-W. Kao, S.-H. Fang, Y. Tsao, L.-C. Kuo, K. Shih-Wei, and N.-C. Lin, "Robust wi-fi location fingerprinting against device diversity based on spatial mean normalization," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*, Oct 2013, pp. 1–4.

[10] M. B. Kjrgaard, "Indoor location fingerprinting with heterogeneous clients." *Pervasive and Mobile Computing*, vol. 7, no. 1, pp. 31–43, 2011.

[11] C. Laoudias, P. Kolios, and C. Panayiotou, "Differential signal strength fingerprinting revisited," in *Indoor Positioning and Indoor Navigation (IPIN), 2014 International Conference on*, Oct 2014, pp. 30–37.

[12] C. Laoudias, D. Zeinalipour-Yazti, and C. Panayiotou, "Crowdsourced indoor localization for diverse devices through radiomap fusion," in *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, Oct 2013, pp. 1–7.

[13] J. Machaj, P. Brida, and R. Piche, "Rank based fingerprinting algorithm for indoor positioning," in *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, Sept 2011, pp. 1–6.

[14] Tod smart beacon (dragonfly 1.0). [Online]. Available: http://www.discoveraero.com/shop/dragonfly-1-0/

[15] Kontakt beacons. [Online]. Available: https://kontakt.io/