WiFi access point localization in urban multi-sensor environment

Boris Tomaš, Lovro Posarić
Faculty of Organization and Informatics, IOT Laboratory
University of Zagreb
Pavlinska 2, 42000 Varaždin, Croatia
`{btomas, lposaric}@foi.hr`

Abstract. *Urban sensor networks produce large amount of data including available WiFi networks information and current GPS location. This work focuses on finding usable information in this WiFi "noise" in urban environments, envisioning the new WiFi access point localization mapping technique. Main challenges of such service are identified and their solution is proposed as an application with specific algorithm for WiFi access point localization. The application architecture is a simple client server model with external actors responsible for data acquisition. Source of the data is an urban sensor network, which is sort of wireless network specifically set up in urban mobile environments.*

Keywords. trilateration, localization, mapping

1 Introduction

In an urban multi-sensor environment there is a large amount of gathered data. After the initial gathering stage, these data need to be synchronized with a central database at the end of data gathering, but ideal would be real time data synchronization. However, for the real time synchronization stable and constant Internet access is required. In urban environments there are many open access WiFi access points (hotspots) that provide free Internet access, location of those devices is fairly constant, exceptions are mobile hotspots on mobile devices and in special occasions dedicated hotspots for conferences, fairs, exhibitions,... Location of fixed hotspots is mostly unknown and it is difficult to determine location of hotspot with high precision because hotspots are mostly in private homes and apartments. Localizing hotspot would mean that someone should visit every home in an area and interview the residents whether they have or don't have WiFi hotspot. Hence, different method needs to be used in order to create a localization mapping of available access points.

2 Problem

Mobile object during movement in the urban sensor environment may require stable and constant internet access using existing WiFi infrastructure. Problem is that this infrastructure is not know. Location of the WiFi access point (AP) needs to be known, but APs themselves don't advertise their exact location. On the other hand, urban multi-sensor networks (UMN) like one described in (Rodrigues, Aguiar, Vieira, Barros, & Cunha, 2011) collect various data types like GPS location, velocity, acceleration, time, WiFi information (SSID, MAC address, signal and noise strength,...) and many more. UMN are usually implemented in the urban environments that do have many WiFi hotspots deployed around the area. UMN provide valid data input for the localizing AP in urban environments. The end goal is to create the mapping of access point locations present in the UMN.

Application that may use localization of AP is the assignment of WiFi AP to the roaming object in UMN (or any other network). This assignment would be necessary because of the offloading existing mobile data networks networks with WiFi is more efficient (Deshpande, Hou, & Das, 2010). Every time a moving object is changing its assigned AP it costs time because of authentication process and assignment procedure called handshaking which lasts for about 330ms (Park, Han, & Kim, 2009). During this time vehicle travelling 80km/h can travel up to 7.3m which can be crucial for some applications using urban sensor networks.

3 Related work

For the indoor localization, WiFi positioning techniques based on Wireless Local Area Network (WLAN) or Wireless Personal Area Network (WPAN) is commonly employed presently (Mok & Retscher, 2007). Given the ubiquity of WiFi networks and WiFi access points, as well as the fact that nearly all smartphones nowadays have a built-in WiFi module, WiFi positioning has become a prominent tool for indoor po-

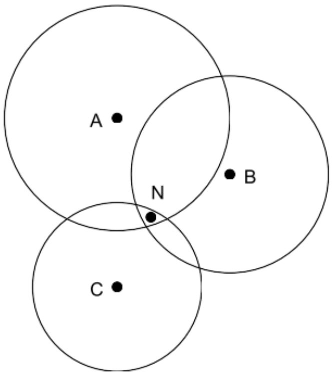

Figure 1: Circle intersections with clustering (Kaminsky, 2007)

sitioning. In open environments the most famous positioning technology is the GPS/GLONASS positioning. It is only even more reinforced by the fact that it does not require additional special-purpose hardware and to top that off, location estimation can be easily attained by measuring the received signal strength (RSS) from a Wi-Fi access point (Li et al., 2019). This signal strength can be converted to the distance which can represent circle radius, more on this later on. Alternative to the signal strength is the use of radio signal propagation over time (Mirisola, 2003). As an approach based on signal strength, this technique is also not bullet proof from reflections and multipathing. Time measurement of signal propagation requires very precise measurement equipment that can not be found in the mainstream WiFi systems. Distance, using time, can be calculated by multiplying the speed of light with measured timespan, this distance can also represent the radius of eventual coverage estimation circle In both cases WiFi AP position can be determined using relatively easy calculations like the special case of multilateration - trilateration: "The trilateration problem is to find the coordinates of node $N = (nx, ny)$ from the given information. A complicating factor is that the known nodes' coordinates and distances typically include measurement errors. Two methods of solving the trilateration problem are nonlinear least squares and circle intersections with clustering." (Kaminsky, 2007) Main prerequisite for the good trilateration is to have a minimum of 3 non linear sources (Konrad & Wölfel, 2012). In urban multi-sensor environments, like urban travel trajectory WiFi scans, this requirement is not always achievable.

In Figure 1, three points can be seen: A, B and C, each of those points have a circle around them. Trilateration procedure is based on solving the equation set of 3 circle equations. The ideal result would be a single point, however in real-time scenarios result usually is a polygon and location point (N) is the centroid of resulting polygon.

$$(x - x_1) + (y - y_1) = r_1^2 \tag{1}$$

$$(x - x_2) + (y - y_2) = r_3^2 \tag{2}$$

$$(x - x_3) + (y - y_3) = r_3^2 \tag{3}$$

It is required that at least three circles are used in the calculation. If only two circles are used, it will be accurate in one dimension (the distance between the APs), but won't be able to accurately detect the location in the second dimension. On the flip side, more than five or six APs could limit the effectiveness by adding unnecessary noise and interference in the environment. In these related cases, trilateration is used to determine a location of a object in space (open or closed) using the fixed location sources' radiation. In this paper, trilateration is used in a case where the object location is relatively known (using GPS), but the location of radiating WiFi hotspots is unknown. Basically it is the same principle, but in a reverse direction.

3.1 WiFi AP localization

Each moving object scans its surroundings for the existing WiFi networks. For each WiFi AP scan, relevant data is gathered:

- GPS *location*;
- MAC address;
- SNR Signal Noise Ratio in dBm.

Procedure follows the case shown in Figure 1 where for each scan circle the coverage area is calculated. Assume that A, B and C are the GPS location of the moving object. Radius represents the distance to the probable WiFi location. Radius (r) is calculated using *free-space path loss* formula:

$$r = 10^{\frac{\ln(10)(T - R - K - 20\log(f))}{10n}}$$

where:

- R - receive power level, this value is being gathered using WiFi scanning equipment,
- T - transmit signal power strength is 17dBm. According to IEEE 802.11b standard[1] the maximum power output level is 20dBm and the minimum gives 13dBm, this gives average of 17dBm, while the actual value depends on the firmware used by the router that powers the access point,
- K - path loss constant, value is -147.55,
- f - the WiFi frequency and is set to 2450 MHz,
- n - the path loss exponent, it is set to 3 because we are measuring distances in the highly urban environment. Typically used values are: 2 for free space, 2.7 to 3.5 for urban areas, 3.0 to 5.0 in suburban areas and 1.6 to 1.8 for indoors when there is line of sight to the router,

---

[1] `http://www.cisco.com/en/US/docs/optical/15000r7_0/15327/reference/guide/2770spcx.html`

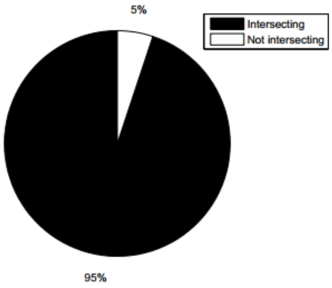

Figure 2: Circle intersection states for assigned AP

- r - is the distance (radius), the result in meters.

After the radius has been calculated, a circle around each moving object can be constructed. This circle represents "how well does an object hear a WiFi AP" not the location estimation of a WiFi AP. To determine the AP location it would be necessary to undertake the trilateration procedure. Figure 1 shows a case with 3 scan points from moving object each with its own radius. First it would be good to investigate the correlation between circles defined by point A, B and C and the appropriate radii. Possible correlation states are:

- One point intersection: This would be the best case scenario but it is rare;
- Polygon intersection: This is the most common case where the intersection is a polygon like shown in Figure 1;
- No intersection. If there is no intersection, it is not possible to carry out trilateration between the three circles as the mutual AP can only exist in the intersection

Figure 2 shows the current state of intersections circles in the system. To simplify, states 1 and 2 make no difference for the trilateration procedure. Important is the existence or no existence of intersection.
If there is an intersection then probable location of AP is the centroid of intersecting polygon (or point). In Figure 1 this is the point N.

Trilateration starts from the starting circle and then calculates intersection with all the other circles. For each circle the intersection is calculated between the circle and the intersection itself. In turn, this makes the intersection (polygon) shrink with every circle considered.
Problem is how to select the starting circle if there are some circles that do not intersect with any other circle, or there may be a clusters of intersecting circles that may result in two or more intersecting polygons. This paper identifies two strategies:

- Most Intersections First (MIFS). This strategy selects the starting circle by selecting one with the most intersections. This means that for each circle it should be calculated with how many other circles it intersects.
- SIF: Smallest Circle First (SIF): This strategy selects the starting circle as one with the smallest radius (strongest signal) with at least one possible intersection.

Evaluation of each strategy will be done later. After the intersection (polygon) is found, then centroid of a polygon is the WiFi AP location estimation. Next, it would be necessary to define the radius of WiFi AP coverage estimation. Radius is calculated using the same path-loss equation (3.1) Input signal strength in this case is the lowest signal strength of the scan circle that contributed in the final intersection forming.

4 Implementation

This procedure follows following steps:

- Calculate the distance (*radius*) using the *path loss* equation defined before: 3.1 using scanned SNR as input.
- Create 12 side polygon with the centre in *location* point and *radius* (distance from centre to the middle of a polygon side). This polygon represents a circle[2] inside which there is probably a WiFi AP that is being scanned.
- Get the list of all WiFi AP that are scanned during trip.
- For each WiFi AP get all the scan circles; and for each circle calculate count of intersections it has with all other circles for the current WiFi AP.
- Select the starting circle using different algorithm variations defined before: (see: 3.1) *MIFS* or *SRF* Starting circle is intersecting polygon (I_p)
- For each circle for the current WiFi AP get the intersection with I_p. Intersection result is stored in the same variable I_p. When all circles are iterated then result is a polygon with the most probable WiFi AP location. Figure 3 shows the result of this step. Blue circles from A to F are scan circles, intersection of those circles is shown as the red polygon "I".
- Centroid of I_p represents the most probable location of current WiFi AP location.
- Find the weakest signal circle, one that is the largest and that forms the intersecting shape that is assigned to this WiFi AP. Radius of this circle is used as a coverage radius for the WiFi AP. Retrieving distance using path-loss model using SNR of weakest signal circle would be redundant because this has been already done. Coverage area is represented as a circle that is also a 12 side

---

[2] it is referred to as a circle although it is a dodecagon.

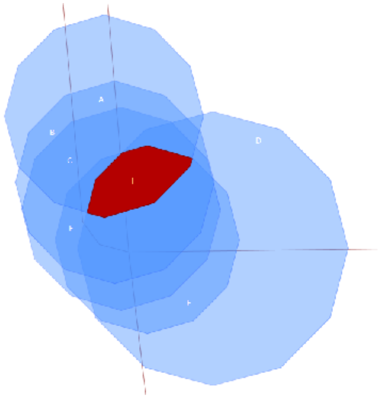

Figure 3: Intersecting polygon/shape (I_P)

polygon that has a radius equal to the radius of the weakest circle and center is equal to the centroid location of I_P.

Database used for this work is PostgreSQL with PostGIS extensions for geo-spatial functions.This implementation uses the geometrical intersection function of underlying database. It is used instead of custom method being implemented that only solves the set of two circle equations mathematically, because, in future work, scan area does not need to be a circle at all, it can be any shape. This way, described algorithm and implementation will work even if the scan shape is not the circle but some complex polygon.

5 Results

There are total of 4 algorithm variations (A):

- MIFS epe+ (Most intersections first with the GPS error over $10m$);
- SRF epe+ (Smallest radius firs with the GPS error over $10m$);
- MIFS epe- (Most intersections first with the GPS error below $10m$);
- SRF epe- (Smallest radius first with the GPS error below $10m$);

Analysis of algorithm variations is done on a test case scenario[3] where the real location of 35 WiFi AP is known.

5.1 Metrics and validation

In the test case scenario for each AP there are 4 locations based on 4 algorithm variations. Distance between each variation and real location is calculated: Distance measurements (d) are:

$$
\begin{array}{l|l}
d_1 = d(A_1, A_2) & \text{Where Ai is:} \\
d_2 = d(A_1, A_3) & A_1 = MIFS\ epe+ \\
d_3 = d(A_1, A_4) & A_2 = SRF\ epe+ \\
d_4 = d(A_2, A_3) & A_3 = MIFS\ epe- \\
d_5 = d(A_2, A_4) & A_4 = SRF\ epe- \\
d_6 = d(A_3, A_4) & \\
d_7 = d(R, A_1) & \\
d_8 = d(R, A_2) & \\
d_9 = d(R, A_3) & \\
d_10 = d(R, A_4) &
\end{array}
$$

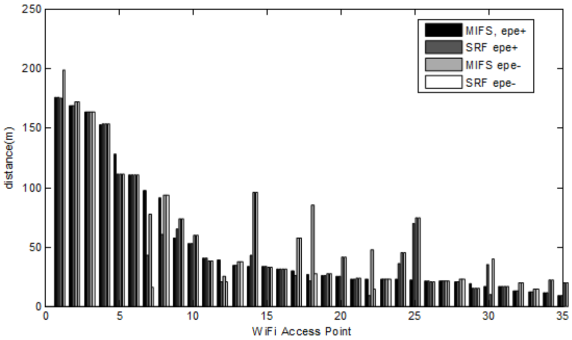

Figure 4: Distance to the real location by algorithm type

Figure 4 shows the distances from the estimated location (for each algorithm combination) to the WiFi AP real location. Test scenario is made of 35 WiFi AP and it can be seen that the distance difference is mostly insignificant (few meters), but there are "spikes" which indicate that the algorithms 3 and 4 (MIFS epe- and SRF epe-) provide worse results than algorithms with GPS error. Figure 5 shows the number of measurements taken for each WiFi AP. Total scans line shows the total number of scans taken, *Total-snr5%* shows the number of scans without 5% of WiFi scans with lowest signal strength and *Total-snr5%-epe* is the number of scans without 5% of the worst signal and without scans with the GPS error above 10m. Each "spike" from Figure 4 is backed by the spike in Figure 5 which means that in the case of exception there is a huge number of scans being measured. On the other hand it can be seen

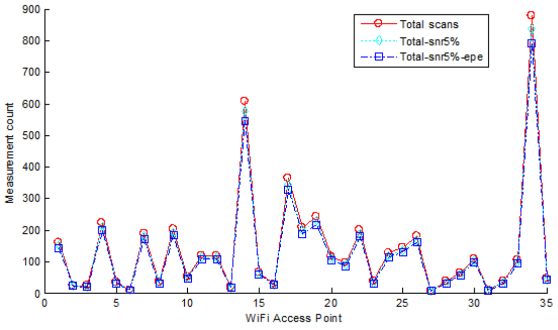

Figure 5: Measurements count for each WiFi AP

---

[3] Grounds of Campus S.João in Porto, Portugal

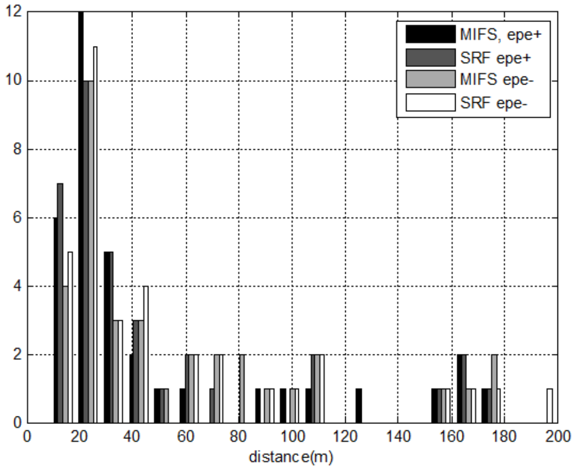

Figure 6: Histogram of distances from estimated to real location by algorithm variations

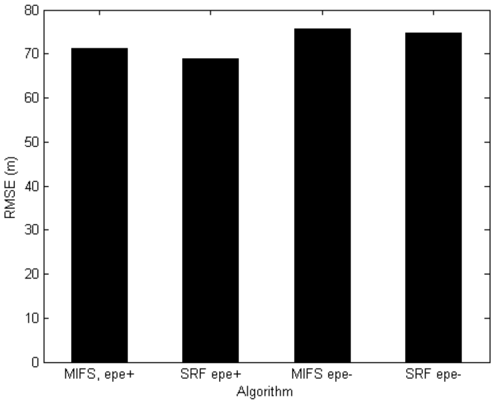

Figure 7: Root-mean square error for location estimation using algorithm variations

that the number of measurements(scans) do influence deviation of estimated location to the real location. Although, there is no rule except that in case of high distance deviation number of scans is low (less than 210 scans), on the other hand small deviation (high precision) does not necessary need to be backed with large amount of measurements(scans). There is, however, an interesting case of AP 34 where distance deviation is small 20m and measurements count is very high, around 850 measurements. Also it can be seen that the algorithms with GPS error produced more precise results in this case. Which may be interpreted as that the GPS error is averaged out by the large number of measurements. Algorithms with GPS error produce better results, likely due to the number of scans undertaken.

Finally, Figure 6 shows the histogram of mentioned test case. It can be seen that the algorithm variations with GPS error included do produce better results regarding distance of the estimated location to the real location.

Figure 7 shows root-mean squared error that is calculated as distance between the estimated location and the real location. It shows that SRF with GPS error does produce better location estimation.

6 Conclusion

This paper describes solutions to the problem of crowsourcing data in urban sensor environment and localization of access points in space, however there are different approaches that may provide different results. As a part of the future work one such approach is the probabilistic approach; instead of geometric analysis, calculation of probability is done using input data. It would be interesting to see the difference between geometric and probabilistic location estimation. Part of the future work is a final implementation of this solution; it should be deployed as a web service that can be used in many applications.

Acknowledgments

This work has been fully supported by Croatian Science Foundation under the project IP-2019-04-4864.

References

Deshpande, P., Hou, X., & Das, S. R. (2010). Performance comparison of 3G and metro-scale WiFi for vehicular network access. *Proceedings of the 10th annual conference on Internet measurement - IMC '10*, 301. Retrieved from `http://portal.acm.org/citation.cfm?doid=1879141.1879180` doi: 10.1145/1879141.1879180

Kaminsky, A. (2007). Trilateration. , *1*(2), 8–11.

Konrad, T., & Wölfel, P. (2012). WiFi Compass. , *2012*.

Li, S., Hedley, M., Bengston, K., Humphrey, D., Johnson, M., & Ni, W. (2019, 03). Passive localization of standard wifi devices. *IEEE Systems Journal*, *PP*, 1-4. doi: 10.1109/JSYST.2019.2903278

Mirisola, L. G. B. (2003). The Localization Problem on Sensor Networks.

Mok, E., & Retscher, G. (2007). Location determination using WiFi fingerprinting versus WiFi trilateration. *Journal of Location Based Services*, 1–15. Retrieved from `http://www.tandfonline.com/doi/abs/10.1080/17489720701781905`

Park, H.-S., Han, S.-H., & Kim, J.-D. (2009). Vehicular client roaming and location-based handoff through multiple WLAN APs in a container terminal. *Proceedings of the 2009 International Conference on Hybrid Information Technology - ICHIT '09*, 465–472. Retrieved from `http://portal.acm.org/citation.cfm?doid=1644993.1645079` doi: 10.1145/1644993.1645079

Rodrigues, J. G. P., Aguiar, A., Vieira, F., Barros, J., & Cunha, J. P. S. (2011, October). A mobile sensing architecture for massive urban scanning. *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 1132–1137. Retrieved from `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6082958` doi: 10.1109/ITSC.2011.6082958