AoA estimation with time separated RSSI readings from a single AP and IMU feedback

Agrim Gupta, Debarnab Mitra, Varunesh Goyal, Prasanna Chaporkar
Department of Electrical Engineering, IIT-Bombay
{agrim,debarnab,varunesh,chaporkar}@ee.iitb.ac.in

Abstract—**Many methods in literature which propose calibration free indoor localization using Received Signal Strength Indication (RSSI) readings of WiFi and use multilateration techniques. In this work we propose indoor localization of a smart phone using a single AP, by interleaving distance measurement via RSSI readings and user displacement measurement from Inertial Measurement Unit (IMU) readings. The proposed algorithm can be intuitively thought of as a variant of multilateration technique, albeit done over time. The algorithm provided Angle of Arrival (AoA) estimate within 20° of the true AoA with 80% probability upon experimental evaluation in LOS setting.**

I. INTRODUCTION

Currently, WiFi based indoor networks and smartphones have become ubiquitous, and hence Indoor Localization using WiFi is an attractive problem with many real life applications. WiFi based localization methods revolve around utilizing [1]:

1) **RSSI measurements** indicate the actual received power at the receiver and by exploiting a path loss model, we can estimate distance between AP and receiver.
2) **Time of Flight (ToF)** or **Time of Arrival (ToA)** indicate the distance between AP and receiver by estimating the signal propagation time, and then multiplying it with speed of light to give an estimate of the distance.
3) **Channel State Information (CSI)** is a fine-grained value describing the amplitude and phase on each subcarrier in the frequency domain [2]. Hence using CSI, one can estimate both distance and AoA directly.

Among these, RSSI based algorithms are most widespread because firstly, getting RSSI readings from physical layer is much easier than CSI since most of the chip designers expose it directly to higher layers [3] and secondly, it does not involve design of new specialized hardware. RSSI based algorithms can themselves be broadly be divided into 2 categories:-

- **Calibration free algorithms**: These algorithms attempt to localize the user in a general environment without assumed prior knowledge of the environment user is in. Though the algorithms may have some sort of calibration, the calibrations done do not go over the environment.
- **Fingerprinting based algorithms**: These algorithms investigate the environment before localizing a user. This involves fingerprinting of RSSI values obtained at various places, and then applying estimation techniques ranging from simpler Bayesian to complex neural network based algorithms.

The fingerprinting based algorithms do manage to get much better accuracy as compared to calibration free algorithms, but this is at the cost of 'fitting' to the environment, and any changes in environment will effect the accuracy. Hence, these methods are less commercially and practically viable.

Most calibration free RSSI based localization methods use at least three anchor APs (i.e. APs whose location are known) and estimate position by getting the distance estimate via RSSI from the three APs using a technique known as triangulation. The common problems associated with this method include requirement of time synchronization between these three APs, performance degradation due to multipath effects and poor performance in Non Line of Sight (LoS) regimes. Complex algorithms have been proposed to achieve high accuracy, by taking care of lack of time synchronization, but these methods make the mass deployment of such systems difficult. The current CSI based algorithms have been known to be susceptible to placement of smart-phone with respect to body and multipath effects similar to RSSI [3].

In this paper we propose a new algorithm which interleaves IMU measurements with RSSI measurements to estimate AoA without help of CSI values. The key concept in the algorithm is to use multiple RSSI measurements from the same antenna, albeit separated over time with displacement of the user as side information to estimate AoA using IMU readings. There are localization algorithms which use RSSI along with IMU measurements [4] [5], but most of them are based on localization using both the methods and then sensor fusing them via a Kalman filter. In our case, the IMU measurements interleave with the RSSI measurements which is different from the sensor fusion counterpart.

Some novel points of our proposed algorithm are :-

- The infrastructure required as compared to existing methods in literature is lesser since it requires just a single AP. Hence, our algorithm can be readily applied to scenarios like localization of a moving Peer in a Peer to Peer network.
- Using just a single AP also makes the algorithm independent of time synchronization problems. Interleaving IMU measurements with RSSI and adoption of a FSM algorithm solves the problem of blowing up of error with time, which other PDR schemes are prone towards.
- Our algorithm can be considered as a dual method to the CSI based algorithms, which utilize offsets in phase to estimate AoA, whereas we use the changes in RSSI over time. Hence, sensor fusing the proposed algorithm with CSI based algorithms will be an attractive problem to look at. After sensor fusing with CSI approaches, the algorithm can

be readily applied to scenarios like localization of a moving consumer in indoor areas like malls.

Section II of the paper discusses the system model and assumptions made in detail. The algorithm to estimate AoA is presented in Section III. The results and future work possible are subsequently discussed in Sections IV and V.

II. System Model

The algorithm assumes a stationary AP and a moving UE (User Equipment). It is also assumed that the moving UE has gyroscope, accelerometer and magnetometer sensor support and it communicates these values along with RSSI values to the AP [1] periodically with the period being referred to *Observation Period (OP)* hitherto. These sensors are required to estimate the displacement of the user which acts as side information to RSSI. We propose estimation of displacement of the user by measuring distance using a step counting algorithm and heading via orientation sensors.

Estimation of heading is done using the orientation sensor implementation in android. The approach forms a rotation matrix using filtered accelerometer and magnetometer readings and computes orientation of the UE using the rotation matrix. It is assumed that the user places the UE such that it's y-axis aligns with his direction of motion. The angle made by the y-axis and north pole, which is obtained from the rotation matrix formed, thus indicates the heading of the user with respect to North Pole. The rotation matrix approach gives better performance than using the raw sensor readings, which is indicated in the android developer guide and also reported by [6]. The heading of the user with respect to North Pole is denoted by θ in this paper.

Directly integrating the accelerometer readings to obtain distance provides erroneous results. An alternate method is to detect the steps via available pedometer algorithms. The current step detection algorithm performs moving average of the accelerometer readings. It detects when the instantaneous acceleration in z direction with respect to the current moving average estimate, is above a set threshold, which is indicative of a step taken by the user. [7] also presents a similar algorithm (however for acceleration in y direction instead of z) which obtains reasonable accuracy. The distance moved by the user is indicated by d in this paper.

Since the speed of UE is assumed to be low, the algorithm assumes that there are no abrupt changes in θ readings recorded in the OP time. Thus, at instances of abrupt changes in θ (like when UE takes a turn while moving), the algorithm needs to be started again and all values are reset. The reset decision is taken by thresholding the standard deviation of θ readings obtained over the OP time.

III. Estimation of AoA

We propose an FSM model (Fig. 1) to estimate the AoA. The model comprises of 4 states depending on whether the UE is stationary or moving. The state transitions are triggered by occurrence of steps over the OP time.

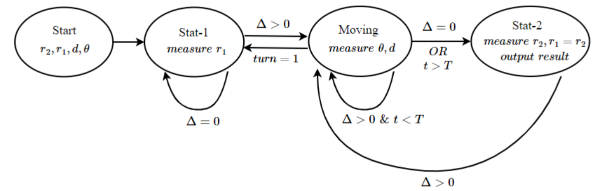

Fig. 1. Proposed FSM Model

The transition expressions are based on calculating the change in steps (Δ) taken by the user based on the sensor readings obtained over the OP. As an example, $\Delta = 2$ when OP is 3 seconds indicates that based on the sensor readings of the past 3 seconds, UE has moved by 2 steps; Similarly, $\Delta = 0$ indicates that UE was at rest in the previous 3 seconds. Next, we describe the states as in Fig. 1:

- **Start:** This is a placeholder state which indicates that the system always starts in STAT-1. r_1, r_2, d and θ are the system variables which are defined in the henceforth states.
- **STAT-1**: The system remains in this state till the UE is stationary ($\Delta = 0$). In this state, the system obtains r_1 via a calibrated path loss model[2], using RSSI measurements. r_1 is the initial distance of the UE from the AP. While the UE is in this state, it keeps measuring the RSSI values and averages them to improve the estimate of r_1. When the system detects a step ($\Delta > 0$), it goes to the Moving state (MOV).
- **MOV**: It is assumed that in this state, the UE moves roughly in a straight line. The system estimates the angle θ with north pole with which it moves as explained in Section II. If the standard deviation in θ readings exceed a threshold, the boolean variable *turn* is set to 1 and the algorithm is restarted as explained in Section II.

In this state, the system also estimates the distance d moved by UE by detecting the number of steps over the OP time. As long as FSM detects steps ($\Delta > 0$), it reiterates in this state and keeps averaging the estimate of θ, and updates the number of steps taken since the machine first moved to the MOV state. When it encounters a $\Delta = 0$ (i.e. no step in OP), it moves to the stationary-2 state (STAT-2). After a threshold time T the system moves to the STAT-2 to prevent deadlocks.

Once this state is over, FSM has estimate of θ obtained by averaging the magnetometer readings, and the distance moved by UE, $d = \Sigma_i^n \Delta_i * step\_len$, where Δ_i indicate the steps taken in the i^{th} MOV iteration, considering UE was continuously in MOV for n iterations, that is 'n' continuous $\Delta > 0$ measurements while being in MOV state.

- **STAT-2:** In this state, if the transition is made from MOV (i.e. the previous state was MOV), the system first measures r_2, the updated distance of the UE from the AP, by averaging RSSI similar to STAT-1. Then it uses θ and d from the previous state and calculates the AoA (and outputs), as explained in subsection III-A. The state betters upon it's estimate of r_2

[1]The communication is not required for the algorithm to work, but for the application case that AP wants to know the location of UE, and later to sensor fuse it with CSI based approach

[2]The path loss model for Motorola XPlay was taken from the android beacon library

successively till no steps are detected and moves back to the moving state once some steps are detected ($\Delta > 0$). STAT-2 also sets $r_1 = r_2$ before moving to MOV, since in principle, this is what STAT-1 was doing, apart from calculation of result. Subsequent transitions are made between MOV and STAT-2 unless a turn is detected in MOV, by which the machine is reset and goes back to STAT-1.

Thus finally to summarize, in STAT-1, FSM measures an accurate value to r_1, and once a step is detected, FSM transitions to MOV state. Then in MOV state, FSM estimates d and θ by step detection algorithm and appropriate averaging of magnetometer readings. After detecting no steps in the fixed time interval, it moves to the STAT-2. In STAT-2 it measures an accurate value if r_2, calculates and outputs the AoA, and sets r_1 to r_2. Once a step is detected, FSM transitions to MOV and carries out the same process again.

A. AoA estimation using r_1, r_2, d, θ

Fig. 2 displays the geometrical setting, with r_1, r_2, d, θ labeled appropriately. We find that with the information at hand (r_1, r_2, d, θ), there are two possible solutions. Resolving this ambiguity will require additional information.

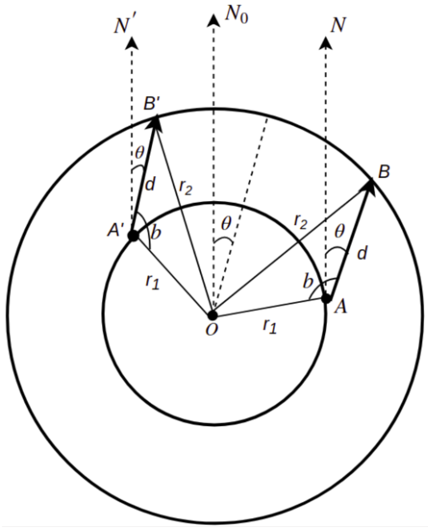

Fig. 2. Inherent Geometry of the problem

The point O indicates the center of the two concentric circles. A, A' are the two possible starting points of the UE during the time interval over which RSSI is recorded. B and B' are corresponding end points. We see that ΔAOB is congruent to $\Delta A'OB'$ (by Side-Side-Side congruency rule). Thus $\angle OAB = \angle OA'B' = b$ (say). The rays AN, ON_o and $A'N'$ all point to the north direction and are thus parallel to each other. We then see the following equations arising:

$$b = \cos^{-1}\frac{r_1^2 + d^2 - r_2^2}{2r_1 d}$$
$$\angle OAN = \angle OAB - \angle NAB = b - \theta$$
$$\angle N_oOA = \pi - \angle OAN = \pi - b + \theta$$
$$\angle N'A'O = \angle B'A'O + \angle N'A'B' = b + \theta$$
$$\angle N_oOA' = -(\pi - \angle OA'N') = -\pi + b + \theta$$

In the above equations, we have considered the angles to be positive or negative, with clockwise direction being positive.

IV. Results

For testing purposes of the algorithm, an android app was developed to send[3] the RSSI readings, step counts and orientation angles periodically to a PC connected to the AP. Although all the computation can be done completely on the phone, having this architecture supports the application we are looking towards (viz. detection of AoA and rough distance of the moving UE by the AP) and also sets up sensor fusion with CSI approach, since the CSI will be measured at the AP only[4]. The tests were conducted on Motorola X-Play smartphone.

The test scenario considered by us is pictorially depicted in the figure on right. The user is confined to motion in a small 5° cone and he repeatedly turns back and forth at boundaries so that we get repeated estimates of roughly the same AoA (α). This was done repeatedly till we hit STAT-2 around 100 times, and then CDF of angle error was plotted wrt the roughly true angle α. The fixed time period of FSM transition was kept to be 2 seconds. We get 80% confidence in detecting the user within 20° cone (Fig. 4).

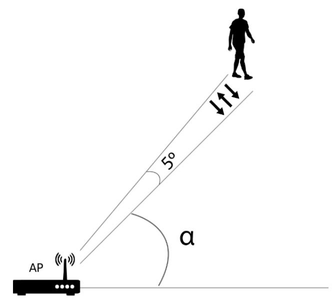

Fig. 3. Test Procedure Considered

All tests were performed considering complete LOS on a terrace setting. The tests also assume the system model stated in Section II. The CDF results obtained are comparable with the results obtained for PILA, MUSIC and CUPID by Tian et al [8] and SpotFi by Kotaru et al [9] given that we use just one AP and one receiver in the algorithm

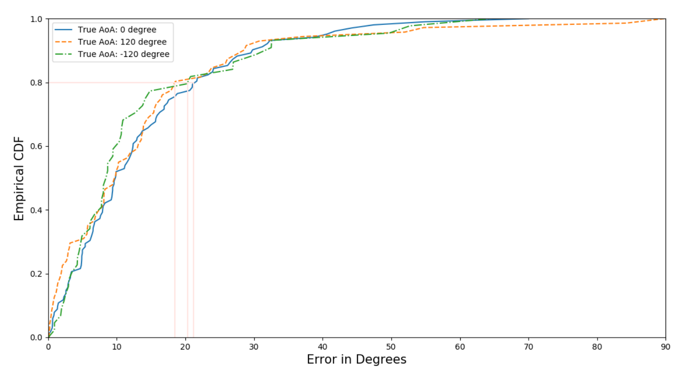

Fig. 4. Error CDF plot of deviation wrt true AoA $\alpha = 0^o, 120^o, -120^o$

Trade-off in choosing the Observation Period (OP): Recall the definition of OP in Section III. OP also becomes the time over which the RSSI readings are averaged upon.

[3]ROS Java was used to establish a connection between phone and PC using the publisher subscriber model of ROS

[4]As it requires special MIMO antennas, which not all phones may have

So, theoretically, increasing the OP increases the accuracy of distance estimates of r_1 and r_2, since more RSSI readings can be obtained and averaged upon. However, increasing the OP hampers the assumption that the UE is at rest for OP amount of time while in STAT-1 or STAT-2, made underlying to the model. Hence, there exists a trade-off here and we need to choose the maximum OP such that both the assumptions hold valid. The choice of OP was obtained experimentally, with Fig. 5 depicting variation of 80% confidence error with OP chosen in seconds. We get optimum OP to be around 3 secs. However the the curve in Fig. 5 is not very steep near the minima and hence a small skew around it's neighbourhood does not affect the performance much.

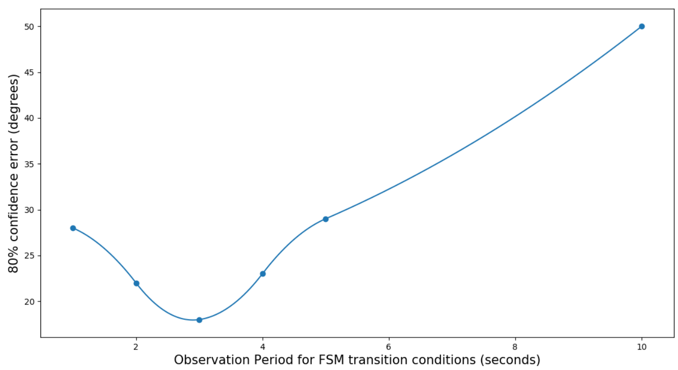

Fig. 5. Performance of the algorithm with different OP values of FSM

Major source of error identified was the random variations in RSSI readings, even though proper LOS was considered. This was also reported by [2].

- The FSM approach helps perform adaptive averaging when user is at rest, i.e. till the UE is at rest, it will remain in STAT and update its distance estimates. This did help in reduce the magnitude of variance as reported by [2].
- Since the algorithm requires RSSI measurement from just the connected AP, we get more RSSI readings per unit time, because the UE need not perform a full scan. A full scan is required in multi-lateration based approaches. Getting more RSSI readings per unit time translates to better distance estimation since we can average more effectively.

Type I and Type II errors in step detection also contribute as an error source. However, the effect of errors in step detection (i.e. parameter d) was observed to be lesser as compared to errors in distance (i.e. parameters r_1 and r_2) estimation, which can be explained by the fact that the optimum OP was 3 seconds, and errors in step detection can thus not be more than 2-3 steps considering normal pace of a person.

V. Future Work

- **Resolving of angle ambiguity**: Currently, we resolved ambiguities manually by considering the angle outputted closest to the expected angle. However, we can resolve the ambiguity by considering additional CSI information at AP, by deciding on the angle coming closest to the CSI angles
- **Sensor Fusion with CSI**: Since CSI is much more fine grained than RSSI, and also provides a direct method to estimate AoA, it is very attractive to do direct AoA esimation using CSI. However, not all network cards expose CSI to application layer and also, CSI is affected drastically with position of body with respect to AP [3]. We can overcome both these problems by considering a specialized AP which exposes CSI and stock android phones which expose RSSI measurement, calculate AoA directly from AP's CSI readings and from proposed algorithm using stock android phone's RSSI and IMU readings and sensor fuse them using Kalman filter to minimize the error.
- **Better Step Detection algorithm**: The naive moving average thresholding algorithm can be improved by considering elaborate filtering techniques as in [10]
- **Phone holding assumption**: It is assumed that the user holds the phone such that the y axis aligns with the direction of motion. That assumption can be improved by profiling the algorithm, viz. estimation of heading when phone is placed in pocket and other similar scenarios
- **Introducing more stages in FSM**: The FSM based algorithm proposed poses a significant advantage in terms of adaptability. By introducing state variables for detection of events like handoffs, door openings, upstair/downstair motions as in [4]. The FSM architecture ensures a smooth logical integration with detection of such events

References

[1] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *CoRR*, vol. abs/1709.01015, 2017. [Online]. Available: http://arxiv.org/abs/1709.01015

[2] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. M. Ni, "Csi-based indoor localization," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 7, pp. 1300–1309, 2013. [Online]. Available: https://doi.org/10.1109/TPDS.2012.214

[3] M. Schssel, "Angle of arrival estimation using wifi and smartphones," *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 4-7 October 2016, Alcal de Henares, Spain*, 2016.

[4] Z. Chen, H. Zou, H. Jiang, Q. Zhu, Y. C. Soh, and L. Xie, "Fusion of wifi, smartphone sensors and landmarks using the kalman filter for indoor localization," *Sensors*, vol. 15, no. 1, pp. 715–732, 2015. [Online]. Available: http://www.mdpi.com/1424-8220/15/1/715

[5] A. Correa, E. M. Diaz, D. B. Ahmed, A. Morell, and J. L. Vicario, "Advanced pedestrian positioning system to smartphones and smartwatches," *Sensors*, vol. 16, no. 11, p. 1903, 2016. [Online]. Available: https://doi.org/10.3390/s16111903

[6] A. Suprem, V. Deep, and T. A. Elarabi, "Orientation and displacement detection for smartphone device based imus," *IEEE Access*, vol. 5, pp. 987–997, 2017. [Online]. Available: https://doi.org/10.1109/ACCESS.2016.2631000

[7] Y. Bai, C. H. Yu, and S. Wu, "Using a three-axis accelerometer and GPS module in a smart phone to measure walking steps and distance," in *IEEE 27th Canadian Conference on Electrical and Computer Engineering, CCECE 2014, Toronto, ON, Canada, May 4-7, 2014*, 2014, pp. 1–6. [Online]. Available: https://doi.org/10.1109/CCECE.2014.6901041

[8] Z. Tian, Z. Li, M. Zhou, Y. Jin, and Z. Wu, "PILA: submeter localization using CSI from commodity wi-fi devices," *Sensors*, vol. 16, no. 10, p. 1664, 2016. [Online]. Available: https://doi.org/10.3390/s16101664

[9] M. Kotaru, K. R. Joshi, D. Bharadia, and S. Katti, "Spotfi: Decimeter level localization using wifi," *Computer Communication Review*, vol. 45, no. 5, pp. 269–282, 2015. [Online]. Available: http://doi.acm.org/10.1145/2829988.2787487

[10] H.-h. Lee, S. Choi, and M.-j. Lee, "Step detection robust against the dynamics of smartphones," *Sensors*, vol. 15, no. 10, pp. 27 230–27 250, 2015. [Online]. Available: http://www.mdpi.com/1424-8220/15/10/27230