Wi-Fi Device Localization in an Indoor Environment Using Graph Mapping

Abheek Saha*, Mayank Maheshwari†
Hughes Systique Corporation
email:{*abheek.saha*,mayank.maheshwari†@hsc.com*}

***Abstract*—Indoor location tracking is an emerging technology that enables consumer oriented businesses such as retailers and hoteliers to better track movement patterns of visitors and generate key metrics like in/out count (footfall), dwell time and understand the popular route(s) inside the facility. These are in turn used for maximizing customer safety, scheduling of optimal workforce and optimal product placements. In this paper, a method is demonstrated for user position mapping by tracking the Wi-Fi signals sent by their smartphones as they walk through an indoor environment, using only the anonymous network probing signals emitted intermittently by each device. Our approach is scalable, unobtrusive and does not require active participation or installation of any special software on tracked devices, with minimal infrastructure costs. Since the devices are not connected to any access point, the signal is anonymized, which aids in protection of user privacy. It is shown how the raw metrics are used to generate accurate position data and ensemble dynamics over a known indoor topology, and how individual models of human behaviour can be used to predict mass movement of crowds in an indoor setting.**

Keywords*—*Wi-Fi tracking; Wi-Fi Probe requests; RSSI fingerprint; group dynamics; MAC address randomization

I. INTRODUCTION

Indoor location positioning and non-intrusive tracking of users, by signals from their Wi-Fi device, is being used in various industry verticals to gain more insight about their customer behavior [1]. The insight thus obtained can be utilized to provide better customer experience. Liu et al in [2] describe various techniques that are being used for indoor location positioning and tracking. Some of the techniques make use of cellular signals for indoor location tracking (see also [3]). Other methods utilize Bluetooth(BT) or Universal Wideband (UWB) signals [4]. Accuracy of methods based on cellular technology is low (50-200 m) [2], and BT based techniques require a BT tag to be attached to tracked item. Further, BT and similar techniques have much smaller range as compared to WiFi signals and hence require a very different approach. In the rest of the paper, we will limit ourselves specifically to WiFi as a wireless access technology for our location determination purpose.

The application that we have in mind is crowd-modelling in an indoor arena. The key outcome of crowd-modelling is to know how many people are located in which part of the indoor arena, i.e., the size of the crowd and further, how the crowd is moving, within the arena. There are many different sensors available for this purpose, starting from optical processing of fixed cameras, footfall sensors, heat sensors, etc. All of them are similar in that they provide random samples of location information, which have to be converted to ensemble data. For our solution, we use Wi-Fi fingerprinting as an equivalent sensor. As we shall see, our method produces similar sampling output and our modelling approach may be used for any sensor based method. In our case the Wi-Fi signals of interest are the probes sent out by Wi-Fi end-points (typically, mobile phones carried by individuals) to detect Wi-Fi networks in the vicinity. We measure the Received Signal Strength Indicator (RSSI) for each probe received and then use a trained Machine Learning (ML) model to convert this into an indication of the zone from which the signal came. This kind of RSSI fingerprinting methods require training/calibration on each new place since no two indoor environments have the same signal propagation characteristics. While there has been a fair amount of research in this area (see Section II), to the best of our knowledge, earlier work have not considered all the other factors which impact the accuracy of location prediction, i.e., different types/make of devices, orientation of the device, etc. We shall show that all of these factors have significant impact on the accuracy of location estimation.

Typically, most systems which track Wi-Fi devices across various locations use the Wi-Fi MAC address as a device identifier. Clearly, there are privacy issues involved in tracking a device using a permanent identifier. In iOS 8, Apple introduced MAC randomization [5] to maintain user privacy during active scan for Wi-Fi networks. Since then, most of the Android phone OEMs have started doing MAC randomization. In the latest versions of Android, Google introduced support for MAC randomization in *Android open source project* (AOSP), hence covering nearly all modern devices. Even though it is possible to predict the location of the Wi-Fi device (identified by its real MAC or randomized MAC), the randomization of MAC makes tracking of a device across locations difficult, given that devices change to a new randomized MAC after transmitting few messages. Various *device fingerprinting* techniques have been identified, which utilize the information available in Wi-Fi messages (other than MAC address) [5][6]. However these techniques are better suited for identifying the type or brand of the device rather than a unique instance. Our method, on the other hand, does not require identification of individual devices; we use probe-measurements as a random sampling technique to generate ensemble location data, which is then fitted to our model.

In the rest of this paper, we use the term *Wi-Fi device* to refer to any consumer device with an active Wi-Fi interface (computers, mobile phones, tablets, etc.). We use the term *scanner* to refer to special access-points operating in monitor mode and placed at known locations, running a special application to capture Wi-Fi signals. One of the scanners is the *anchor scanner*, that is used as a reference to generate differential RSSI signals as mentioned in Section IV. Devices may either

 ISBN: 978-1-61208-802-0

be *training devices*, used to train the location prediction engine, or *tracked devices* whose readings are captured and fed to the location prediction engine for location estimation. The input is in the form of an *RSSI fingerprint*, comprising of differential readings from multiple scanners.

The rest of the paper is organized as follows. In Section II, we cover the current state of the art in Wi-Fi based indoor location positioning . In Section III, we discuss our experiments in indoor location calibration and identify the different non-environmental factors which impact the accuracy of our method. In Section IV, we give a brief description of the Machine Learning technique used for the backend location prediction. In Section V, we show how we convert the individual location samples to a model of the indoor location as a whole. Finally in Section VI, we discuss our final results and potential for future work.

II. Literature Review

Various methods based on Wi-Fi signals are being used for the localization of devices in the indoor environment. Some of the techniques perform lateration by measuring the distance of the Wi-Fi transmitter from Wi-Fi receivers placed at known locations. The distance is measured by calculating either the *Time of Arrival (ToA)* or *Time Difference of Arrival (TDoA)* [7]. In order to give accurate results lateration techniques require time synchronization either between the Wi-Fi transmitter and receiver or between multiple receivers. These techniques also need a very accurate measurement of arrival time since a minute error in measurement leads to an error of a few 100 meters in location calculation [7]. Other techniques measure the angle of incidence (AoA) of Wi-Fi signal at two or more Wi-Fi receivers [8]. The angle of incidence is then used to determine the location of the Wi-Fi transmitter. This method requires a clear line of sight between transmitter and receiver, which is not achievable in the indoor environment for most of the use cases.

The use of RSSI measured value for location determination has been widely discussed [7][9][10][11]. The most common RSSI based techniques employ propagation loss models to measure the distance between the Wi-Fi transmitter and receiver [12]. Distance between transmitter and three or more receiver is then used to find the location of the transmitter. These techniques do not work very well because of multipath in the indoor environment, given the frequency at which Wi-Fi operates, as we shall discuss in Section III. However, there are many other factors which impact the performance. For example, in [13], the authors have described the difference in Wi-Fi transmission characteristics in different Wi-Fi devices. It should be noted that RSSI fingerprinting is being investigated for millimeter wave radio, including the new 802.11ad Wi-Fi standards [14]. The problem of converting movement of ensembles of individuals on a graph have been studied in multiple contexts. These include the movement of a fluid within tubes [15][16], the transport of particles in a network [17], diffusions on graphs with random jumps [18][19][20], etc. The key constituents of these models are the evolution function for each particle on each edge and the transition functions at the vertices, to ensure that there is no build-up between edges. A good summary is provided in [21].

III. Factors Impacting RSSI Readings In an Indoor Environment

In our solution, we have utilized RSSI based techniques for location determination; this is known as location patterning [7]. There are obvious advantages to this technique; it does not require any specialized hardware and can be implemented using standard off the shelf Wi-Fi receivers. Further, it does not require any modification of behaviour, either by the user or by the user device, since measurements are taken passively based on the normal scanning behaviour of the device; this makes the technique less obstrusive and more scalable. The core of this technique is to learn RSSI patterns (RSSI fingerprint) for each location of interest and learn how to map these patterns to transmit locations. This technique consists of two phases. In the first phase, we use labelled data to calibrate our algorithm, which also takes into account the peculiarities of the environment. In the second, we can use real-time measurements for prediction. Obviously, the placement of the scanners cannot be changed between the two phases, neither the environment.

During the calibration phase, the RSSI values of messages received by scanners are tagged by the known location of the transmitter and collated to create the RSSI fingerprint corresponding to a location. The output of this phase is an RSSI fingerprint database corresponding to each location of interest. This database is used to match received RSSI tuples during the prediction phase, as we shall see below.

A. Measurement of RSSI in a closed area

As our work is based on location patterning, using RSSI readings as the fingerprint, we wish to find all the exogenous factors which can cause variations in the RSSI readings. To this end, extensive testing of RSSI readings was done for different categories of devices in an instrumented environment. For our experimentation, we used Raspberry Pi 3 $B+$ boards with Alfa AWUS036HEH Wi-Fi USB dongles to act as Wi-Fi scanners. The Alfa AWUS036HEH Wi-Fi USB dongle was put in monitor mode to sniff Wi-Fi packets while the inbuilt Wi-Fi of Raspberry Pi provided connectivity to LAN. We used one of the floors of our office building to install Wi-Fi scanners. Wi-Fi scanners were hung from the ceiling for better signal reception. Figure 1 shows the placement of Wi-Fi scanners. We divided our office floor into zones of similar size and placed one scanner in each zone. Placement of scanners was such that it avoided concrete/metal pillars and other wireless devices. The working of the system is shown in Figure 2. An application called *find3-cli-scanner* from the opensource package [22] was installed on the Raspberry PIs to sniff Wi-Fi probe packets and forward them to the central server. For each Wi-Fi device (identified by MAC address) the central server collates RSSI values received from scanners, forming a tuple of RSSI values received from all the scanners. For example, if the RSSI value of a probe request was R_j at scanner $S_j, j \in [1..7]$ a typical RSSI fingerprint tuple with

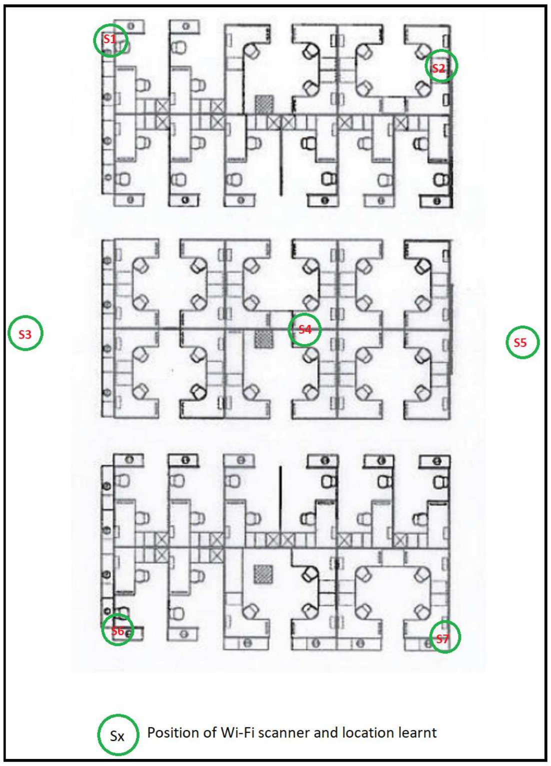


Figure 1. WiFi Scanner Placement.

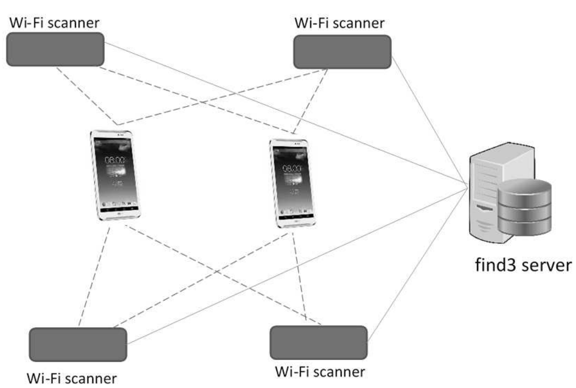


Figure 2. Test and calibration setup.

[S1:R1, S2:R2, S3:R3, S4:R4, S5:R5, S6:R6, S7:R7]

(a) Standard tuple.

[S1:R1-R4, S2:R2-R4, S3:R3-R4, S5:R5-R4, S6:R6-R4, S7:R7-R4]

(b) Differential tuple.

Figure 3. RSSI fingerprint formats.

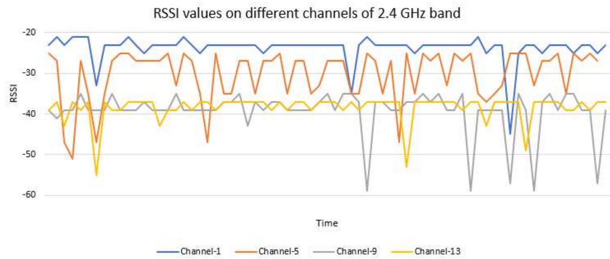
RSSI values from all 7 scanners is shown in Figure 3. For our test, we collected about 2000 RSSI fingerprints over a duration of 8 hours for the calibration phase for each learned location. Collection of RSSI fingerprints over a longer time period helps in capturing variation caused by movement of users in measurment environment. RSSI fingerprints collected during the calibration phase were used to train a ML classification algorithm. For location prediction, the RSSI fingerprint of a device is fed to the same ML classification algorithm, which predicts the probable location of the Wi-Fi device. During our initial tests with this arrangement, location predictions were not very accurate, with prediction accuracy ranging from 70% in the best case to 20% in the worst case. Where we defined accuracy as the percentage of times when system predicted the zone correctly. The reasons for accuracy variation will be explained in the Subsection III-B.

B. Factors impacting location prediction accuracy

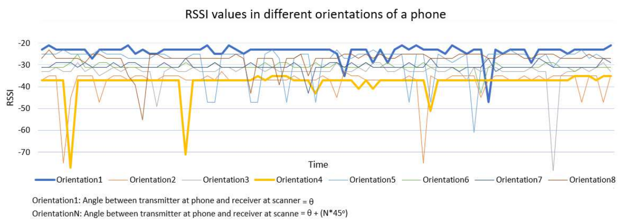
Significant variation in RSSI measurements are caused by non-environmental factors such as the channel, the orientation of the phone, device type and movement of users near the transmitting or receiving device. In this section, we shall report the outcome of experiments that we conducted on the effect of some of these factors. We start with the frequency configuration. Wi-Fi access points typically use frequency hopping, so as to reduce channel specific impairments. In our case, this means that the RSSI probe from the same location may be measured by multiple scanners on different channels, which will then be combined into the same tuple of measurements as a fingerprint. As it turns out this does not work, because different channels *even within the same 2.4Gz band* show wide variation in RSSI readings for the same scanner-transmitter pair. It is hard to say definitely whether this is because of noise in those bands or simply propagation related; however, sporadic interference can be ruled out, because the effects were sustained over a fairly large period of time. In our test, a Wi-Fi device (Moto G5 phone) was placed at a known location, and we measured the RSSI values of packets sent by this device at one of the scanners. The system was configured initially at Channel 1 (2.412 GHz), and then on channels 5(2.432 GHz), 9(2.452 GHz) and 13(2.472 GHz). Figure 4a shows the differences of RSSI values of probe requests received on different channels. There was a difference of about 20dB between the average of the RSSI values of probe requests received on channels 1 and 13.

Our next experiment considers the effect of phone orientation (angle between transmitter of phone and receiver of scanner). We conducted a test wherein one Wi-Fi device (Moto G5 phone) was placed at one of the locations, and RSSI values of its probes were captured at one of the scanners; both devices were set to channel 9. RSSI values of probe requests from Moto G5 were collected at the scanner at 8 different orientations $45°$ apart, for 15 minutes each. Figure 4b shows the average of the RSSI values for each orientation. The angle between Orientations 1 and 4 was $180°$, and RSSI values on these two orientations differ by about 18dB. It is

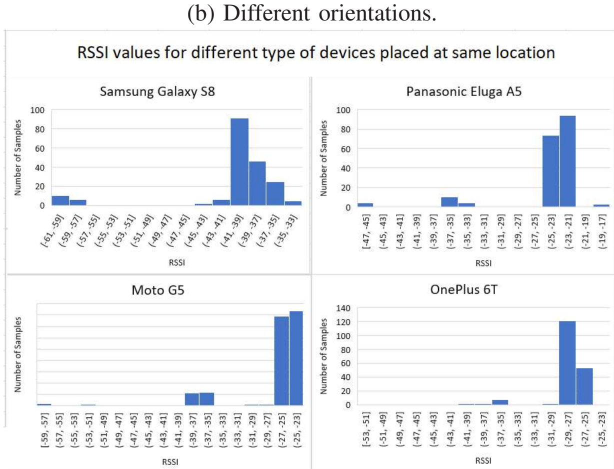
 ISBN: 978-1-61208-802-0



(a) Channel based.



(b) Different orientations.



(c) Different devices.

Figure 4. RSSI variations due to various factors.

clear that the RSSI values change considerably with the change in orientation. The indoor localization method should consider the possibility that the *tracked device* could be placed/carried in any possible orientation.

Finally, we consider the device itself. It turns out that if the *tracked devices* used during the location prediction phase are different from those with which the training was done, the accuracy of location prediction reduces significantly, as low as 20% at some locations. Figure 4c shows the recorded RSSI values for different Wi-Fi devices placed at the exact same location and orientation. We can see that the average RSSI value from OnePlus 6T (-27 dB) and average RSSI value from Motorola G5 (-39 dB) differ by about 12 dB. Our observations have been reproduced by other authors [13]; the difference is due to the combination of chipset, Power-Amp and antennae used by different manufactures.

IV. A Machine Learning solution to location identification

Based on the factors identified in Section III-B, we have identified methods by which we can improve the quality of the RSSI measurements. For example, to take care of the channel variation, we locked all AP scanners to the same channel. If the channel is changed, all the APs must switch to the new channel and a new calibration phase has to take place. For the other factors, we introduced the concept of differential readings using one of the Wi-Fi scanners (typically the one at the center of the coverage area) as an *anchor scanner*. For creating an RSSI fingerprint, instead of using absolute RSSI value, we subtracted the RSSI value at *anchor scanner* from RSSI value at every other scanner (Figure 3b). This takes care of most of variations caused by both the chipset specific and environmental sources. Differential reading removes the differences caused by Power-Amp and antennae used in different types of devices. It is possible to designate any scanner as the anchor, and even introduce multiple anchors for additional robustness. This will increase the complexity of the training but add even more robustness to the data. We shall study this in future work. We used the Machine Learning package *FIND3* [22] for location prediction. The *FIND3* package runs multiple machine learning algorithms in parallel and then chooses the best among them using the Youden's J-statistic diagnostic metric [23] as given in equation (1). These include the K-nearest neighbour, linear SVM, Decision tree, Random Forest, and Extend Naive Bayes algorithms. Using the labeled data provided, each algorithm is trained with a subset of the data and then tested using the remaining part of the data. The prediction is in the form of a probability factor P_L for each location L. Based on the predictions by ML algorithms *Youden's J statistic* is calculated for each location and each ML algorithm.

$$
\begin{aligned}
J &= \frac{T_p}{T_p + F_n} + \frac{T_n}{T_n + F_p} - 1 \\
T_p &= \sum \mathcal{I}_{P_{L_e} > \sigma} && \textit{–True Positive} \\
F_p &= \sum \mathcal{I}_{P_L > \sigma}, L \neq L_e && \textit{–False Positive} \\
T_n &= \sum \mathcal{I}_{P_L < \sigma} \forall L \in \mathcal{L}, L_e \notin \mathcal{L} && \textit{–True Negative} \\
F_n &= \sum \mathcal{I}_{P_L < \sigma} \forall L \in \mathcal{L}, L_e \in \mathcal{L} && \textit{–False Negative}
\end{aligned}
\tag{1}
$$

In the above, σ is an externally supplied goodness-of-fit metric. A value of 1 for J indicates that the prediction by algorithm is perfect and the value of *0* indicates the prediction by algorithm is useless. Based on this metric and a given set of training data, the *FIND3* package will find the best fit model for location prediction. Using this algorithm, the entire calibration was done using a single device and then applied the prediction to multiple different types of devices. We collected more than 2000 samples for each location. After calibration, we performed location prediction for three different types of devices; Samsung Galaxy S8, Panasonic Eluga A5, and Motorola G5. We obtained more than 80% prediction accuracy on all of these devices, within a 3 meter radius of the calibration positions.

V. Modelling the group dynamics of Wi-Fi users

Up to now, we have captured the individual probes from individual transmitters and resolved these into a location with

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-802-0

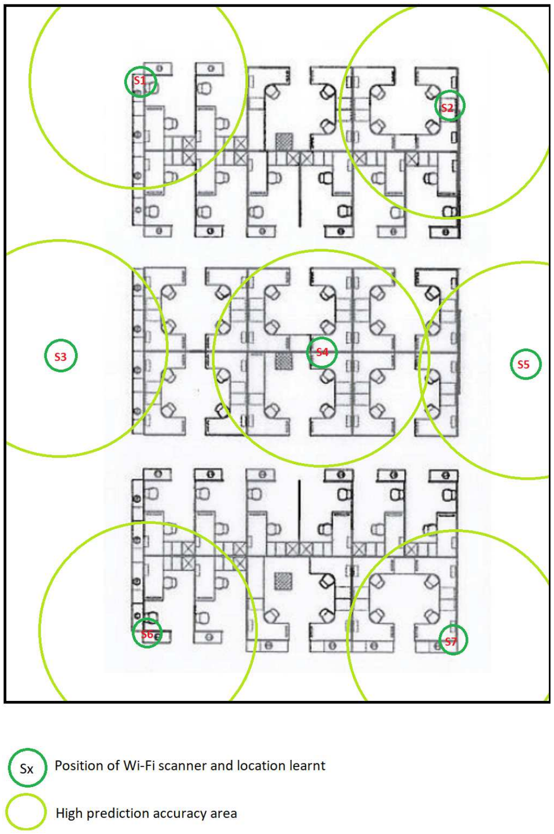


Figure 5. High Accuracy Zones.

a certain probability. In this section, we will see how we can convert the individual data-points into an aggregate model. To do this, we first use a spreading function to convert the impulse data from the samples to a continuous estimate of occupancy and then try to model this as a stochastic process. The model will allow us to predict ensemble behaviour, so that we can convert the movement of individuals into that of a crowd.

The problem of reconstructing occupancy data from sampling is made more complex by the fact that we have no control over the sampling points and further, that each brand of mobile transmits probes at different intervals as shown in Table I. The spreading function must be sufficiently broad so as to capture this variation, but not so broad that we over-value samples.

As discussed in Section IV, the machine learning algorithm gives results in terms of specific zones where the access points are centered with an effective radius of about $3m$. In order to use this, we convert the coverage region into a graph, where the zones represent edges and the vertices represent the transitions from one zone to the other (Figure 6). Each received probe, hence, has to be mapped by the location mapping algorithm into one or more zones with the associated probability of fit.

To despread the probing data and estimate the occupancy function for each zone, we use a root raised cosine spreading function $rrc(T,t,i)$ with a cutoff of 0.85 and a spreading interval of 50 seconds. The spreading function $rrc(T,t,k)$ estimates the likelihood of the transmitter being in the same zone in the time interval $[T-t, T+t]$ from which we have received a probe at time T. The computation of the occupancy function estimate $\hat{c}(l,t) : \mathcal{L} \times [0..T] \longrightarrow \mathbb{R}$ is the weighted sum of all the despread samples in that zone, as is shown in (2). Here $\mathcal{L}$ is the set of all locations within the coverage area and $s(l,\tau)$ refers to a probe request received at time τ which is resolved to be in location l with the probability $s(l,t)$.

$$\hat{c}(l,t) : \sum_{\tau} s(l,\tau) rrc(t-\tau) \tag{2}$$

Once we implement this over all the resolved probes $s \in \mathcal{S}$, we have continuous occupancy estimates $c(l,t)$ for all locations l and for all $0 \leq t \leq T$.

A. Modeling individual user behaviour

Once we have converted the empirical probe data into continuous occupancy estimates $\hat{c}(l,t)$ and the coverage area into a graph $\Gamma = \{\mathcal{N}, \mathcal{V}\}$, we now have to choose a model to fit the empirical data. The baseline assumption is that all the users are homogeneous and the basic individual model is only affected by the edge (location) of the transmitter and the position within that edge. Within the evolution function, we have to identify the parameter which captures the effect of the environment on the particle. If we model the network as a series of pipes and the individuals within it as a frictionless fluid, then the cross-section of each pipe (edge) determines the transport rate within that edge. The entire system of equations must then be solved simultaneously, taking into account the topology of the graph, i.e., the number of edges coming together at each vertex.

In our case, the movement of individual users is modelled by an Ito Diffusion $dx_t = b(x)dt + \sigma(x)dW_t, x(0) = a$ with the $b(), \sigma()$ obeying the usual Lipschitz conditions and W_t being a standard Brownian motion. The diffusion captures both the variability of the data and the drift of the user within a given edge. A key statistic is the *exit* process, which is defined as follows: given that there are N transmitters in a given edge, moving as per a given process, what is the likelihood that the transmitters will exit the edge at a given vertex within the next T_e time period. If we can compute the probability function $P(T_e)$ for the exit time on the eth edge, we can predict the flow of movement from within an edge to the neighbouring edges. A second interesting statistic is the transition process, which determines the user behaviour when she reaches a vortex and has to choose among the edges meeting at the vortex. In this paper, we focus on the exit time. For a diffusion starting from any point a within the known domain, the exit time is given by Dynkin's formula (3).

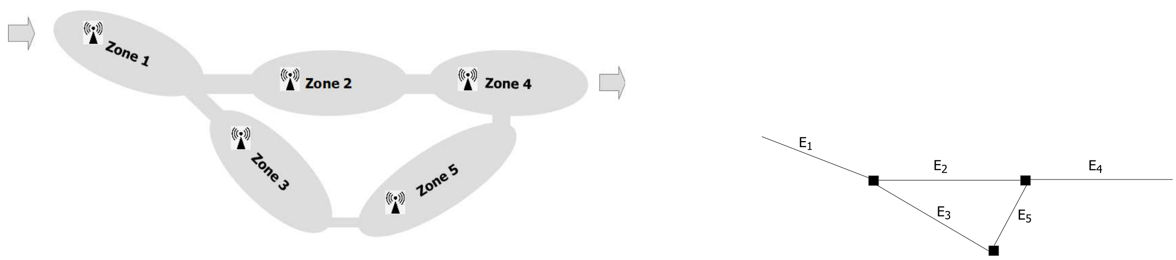
$$\mathbb{E}^a(f(X_t)) = f(a) + \int_0^t \mathcal{A}f(x)ds \tag{3}$$

$$\mathcal{A}f = b(x)\partial_x f + \frac{1}{2}\sigma^{(}x)\partial_x^2 f \tag{4}$$

By setting boundary conditions $f(a) = a$, $f(1) = 0$, we can convert the above to an ordinary partial differential equation $b(x)\partial_x u(x) + 1/2\sigma^2(x)\partial_x^2 u(x) + a = 0$, which is solvable using standard techniques. Once we know the function $a(x)$, we can predict the movement of users from one edge to the other.

TABLE I: PROBE REQUEST TRANSMISSION INTERVAL IN SECONDS

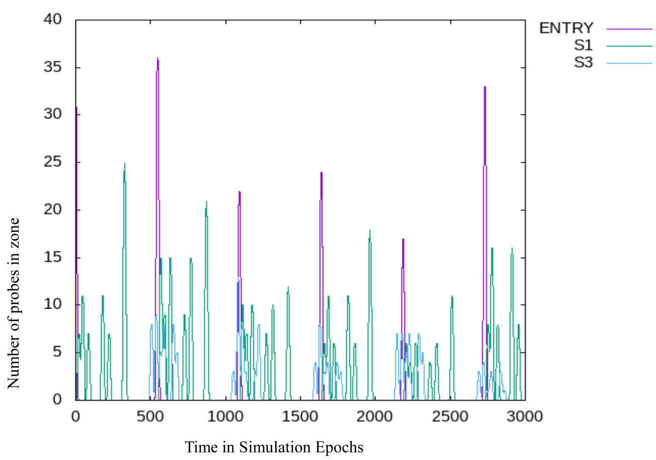
Phone	Phone Screen On			Phone Screen Off			Wi-Fi settings screen open		
	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min
Samsung Galaxy S8	383	702	96	420	787	120	10	10	10
Motorola G5	124	210	49	1020	1920	286	8	3	11
Samsung Galaxy Tab 4	130	130	130	380	600	120	10	10	10
Panasonic Eluga A5	133	268	15	Does not transmit when screen is off			9	11	8



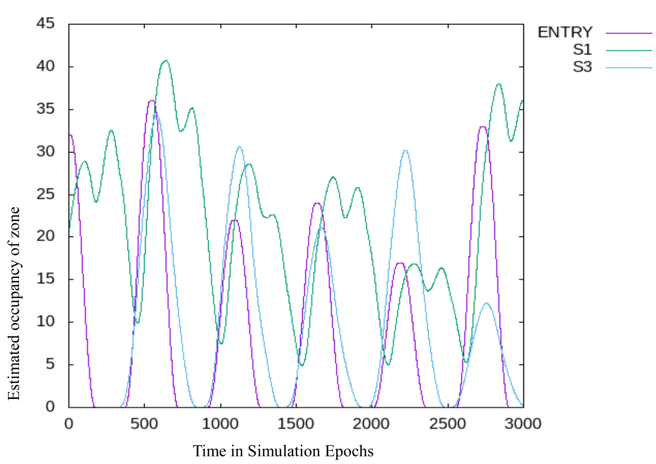
(a) Mapping of topology.

(b) Conversion to graph.

Figure 6. Topological mapping of user locations to a graph.



(a) Probe Data - raw.



(b) Probe data - Fully despread.

Figure 7. Effect of de-spreading on raw probe data.

In our simulation, we use a Brownian Bridge as a model for the movement of individual users within each edge, as shown in (5), where each edge is mapped onto $[0..1]$. The edge specific drift rate γ^i. The advantage of the Brownian bridge is that each user is guaranteed to exit the edge at time γ^i, since $\lim_{t \to \gamma^i} x_t = 1$. Note that x_t^i is normalized with respect to hypothetical length of the edge L^i. The second term is the transition condition at each vertex ($t = 1$), where $\mathcal{I}_j$ is the set of vertices which meet at the jth vertex. The entire set of equations has to be solved for all the edges simultaneously, with the transition conditions providing the boundary value functions.

$$dx_t^i = \frac{1 - x_t}{\gamma^i - t} dt + dW_t, c_0^i = 0 \quad (5)$$

$$\sum_{i \in \mathcal{I}_j} c_1^i = 0 \quad (6)$$

B. Simulation and mass dynamics

While diffusions on graphs can be solved numerically [24], or by using vanishing viscosity techniques [18], we opt to use a simulation method. We seed the prediction by taking a snapshot of the occupancy data at a time $t = 0$ and then use our model to predict the expected $c_t^i \forall i$ edges. The corresponding estimated distribution $p(c(T))$ is compared with the actual empirical distribution $\hat{c}(T)$ to get an idea of how close our model is to reality.

A key metric is the correlation of the modelled occupancy for edges m and n adjacent to each other in the graph 6b. If $c^i(t)$ captures the estimated occupancy of the ith edge,$0 \leq t \leq T$ and the edges m, n are adjoining with a drift rate of $1/\gamma^i$ we should see a cross correlation peak for $c^m(t)c^n(t - 1/\gamma^i)$. In figure 8, we have plotted the correlation between adjacent edges. The first curve shows the correlation of the empirical data and the second shows the correlation peaks for the modelled data. We note that the two curves are in relative match with each other, with the gap between successive peaks capturing the drift from one branch to the other.

 ISBN: 978-1-61208-802-0

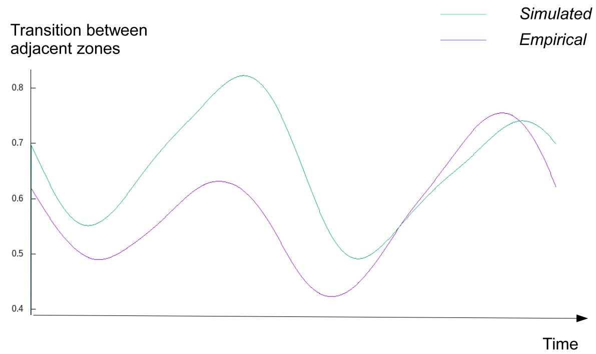


Figure 8. Transition of crowd across adjacent edges measured using estimated occupancy.

VI. Conclusion and future work

In this paper, we have considered the problem of localization of users in an indoor environment of known topology, using only the probes transmitted by their Wi-Fi enabled devices. By using differential measurements, we have shown that a machine learning solution can accurately pinpoint the location within given zones with 80% accuracy, without requiring any kind of user tracking. Further, we have shown the use of empirical measurements to reconstruct the group-dynamics of the ensemble user population by modelling the behaviour of the individuals as diffusions on a graph. The results show that is possible to use the observed mass dynamics of users to derive the individual models of user movements within zones. Our basic approach of using Machine Learning to map between RSSI fingerprints and transmitter locations is a domain of active research. In this paper, we have used the ML algorithm from *FIND3*, without any significant re-architecting. In future work, we would refine the algorithm to take into account our particular use case. One of the challenges for FIND3 (as with many other algorithms) is that it cannot handle incomplete input. In our situation, this means that probe messages which are not picked up by all scanners cannot be processed at all and have to be discarded. In a large indoor arena, it is impractical to expect all probe messages to be picked up by all scanners. Further, like all ML based algorithms, the questions of stability, accuracy and computational requirement require further work. Retraining of the ML for each change in interior topology is CPU intensive and slow; hence, we would like to find ways to augment existing algorithms for minor changes, rather than retrain the entire ML. This is under active consideration. For the group dynamics part, we intend to focus on better models of user behaviour and better metrics for capturing mass dynamics as measured through empirical data. This will help us to create more accurate models of user behaviour which can be validated empirically.

References

[1] D. Goodin., “No, this isn’t a scene from minority report. this trash can is stalking you.” ars Technica, Tech. Rep., Sep. 2013. [Online]. Available: https://arstechnica.com/information-technology/2013/08/no-this-isnt-a-scene-from-minority-report-this-trash-can-is-stalking-you/ [accessed:2020-08-01].

[2] H. Liu, H. Darabi, P. Banerjee, and J. Liu, “Survey of wireless indoor positioning techniques and systems,” IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 6, Nov. 2007, pp. 1067–1080.

[3] V. Otsason, A. Varshavsky, A. LaMarca, and E. de Lara, “Accurate gsm indoor localization,” in UbiComp’05 Proceedings of the 7th international conference on Ubiquitous Computing, Tokyo, Japan, Sep. 2005, pp. 141–158.

[4] S. Gezici and H. V. Poor, “Position estimation via ultra-wide-band signals,” Proceedings of the IEEE, vol. 97, no. 2, 2009, pp. 386–403.

[5] J. Martin and et al., “A study of mac address randomization in mobile devices and when it fails,” in Proceedings on Privacy Enhancing Technologies, 2017, pp. 365–383.

[6] M. Vanhoef, C. Matte, M. Cunche, L. Cardoso, and F. Piessens., “Why mac address randomization is not enough: An analysis of wi-fi network discovery mechanisms.” in In ACM Asia CCS,, May 2016, pp. 413–424.

[7] Cisco, “Wi-fi location-based services, 4.1 design guide.” Cisco, Tech. Rep., May 2008. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Mobility/WiFiLBS-DG.html/ [accessed:2020-08-01].

[8] J. Xiong and K. Jamieson, “Arraytrack: a fine-grained indoor location system,” in in Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13), 2013, pp. 71–84.

[9] M. A. Youssef, A. Agrawal, and A. U. Shankar, “Wlan location determination via clustering and probability distributions,” in Proceedings of the First IEEE International Conference on Pervasive Computing and Communications,, March 2003, pp. 143–150.

[10] M. Youssef and A. Agrawal, “The horus wlan location determination system,” in ACM Proceedings of the 3rd international conference on Mobile systems, applications, and services,, June 2005, pp. 205–218.

[11] P. Bahl and V. Padmanabhan, “Radar: An in-building rf-based user location and tracking system,” in Proceedings IEEE INFOCOM, 2000, pp. 143–150.

[12] P. Kumar, L. Reddy, and S. Varma, “Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks,” in IEEE Conference on Wireless Communication and Sensor Networks,, 2009, pp. 1–4.

[13] G. Lui, T. Gallagher, B. Li, A. G. Dempster, and C. Rizos, “Differences in rssi readings made by different wi-fi chipsets: A limitation of wlan localization,” in IEEE International Conference on Localization and GNSS (ICL-GNSS),, 2011, pp. 53–57.

[14] T. Nitsche and et al., “Ieee 802.11ad: directional 60 ghz communication for multi-gigabit-per-second wi-fi [invited paper],,” IEEE Communications Magazine, vol. 52, no. 12, December 2014, pp. 132–141.

[15] G. Coclite and M. Garavello, “Vanishing viscosity for traffic on networks,” Siam Journal of Mathematical Analysis, vol. 42, no. 4, 2010, pp. 1761–1783.

[16] Y. Achdou, F. Camilli, A. Cutri, and N. Tchou, “Hamilton-jacobi equations on networks,” in Proceedings of the 18th World Congess, The International Federation of Automatic Control,, 2011.

[17] S. Albeverio and S. Kusuoka, “Diffusion processes in thin tubes and their limits on graphs,” The Annals of Probability, vol. 40, no. 5, 2012, pp. 2131–2167.

[18] F. Camilli, C. Marchi, and D. Schieborn, “The vanishing viscosity lime for hamilton-jacobi equations on networks,” Journal of Differential Equations, no. 254, 2013, pp. 4122–4143.

[19] M. I. Freidlin and S.-J. Sheu, “Diffusion processes on graphs: stochastic differential equations, large deviation principle,” Probability Theory and Related Fields, 2000, pp. 181–220.

[20] M. I. Freidlin and A. D. Wentzell, “Diffusion processes on graphs and the averaging principle,” Annals of Probability, vol. 21, no. 4, 1993, pp. 2215–2245.

[21] Y. V. Pokornyi and A. Borovskikh, “Differential equations on networks (geometric graphs),” Journal of Mathematical Sciences, vol. 119, no. 6, 2004, pp. 691–718.

[22] find3, “The framework for internal navigation and discovery,” 2018. [Online]. Available: https://github.com/schollz/find3-cli-scanner/ [accessed:2020-08-01].

[23] W. J. Youden, “Index for rating diagnostic tests,” Cancer, vol. 3, no. 1, 1950, pp. 32–35.

[24] M. K. Fijavz, D. Mugnolo, and E. Sikolya, “Variational and semigroup methods for waves and diffusion in networks,” Applied Mathematics and Optimization, vol. 55, no. 2, 2007, pp. 219–240.

 ISBN: 978-1-61208-802-0