Improving close contact measurements and enabling them in challenging scenarios

Abraham Martn-Campillo[a], Eiko Yonek[a]

[a] *University of Cambridge, United Kingdom*

Abstract

RFID technology has been widely used in recent works for detecting close face-to-face contacts. This information is a very good resource for information diffusion analysis or epidemiology. RFID technology provides more accurate information of contacts than others technologies using location information, as it allows to detect face-to-face contacts. But this technology requires of receivers or readers that have to be installed in the measured area. Current readers require of some properties that make them unusable in challenging scenarios like remote locations in Africa, where measuring close face-to-face contacts can provide valuable information for better vaccination protocols. We present in this paper a new RFID reader device based on affordable Rasbperry Pis that does not require ethernet connection, and is able to work outdoors using delay tolerant networking based on wifi and batteries. In this paper we also analyze the accuracy of the RFID system using real world observable tests. We detect and show some current issues and propose a method to solve them consisting in adding an additional signal for getting additional space information.

Keywords:
RFID, Delay Tolerant Networks, Sensor Networks, Epidemiology

1. Introduction

The interest of measuring contacts between people has increased over the last years. This data can reveal mobility patterns, social interactions, or personal habits. Different models can be applied to this data, like information diffusion or infectious diseases.

There exists different ways to measure interactions: Ranging from social network analysis to face-to-face encounters measurements. The first one provides the social structure of an individual, that can be used to measure influence, information diffusion, etc. The latter, instead, provides physical contacts from the participants, when they have been in close contact, for how long, the frequency of those close contacts, etc. This information, can also be used for information diffusion for example, but in contrast with the social network analysis, face-to-face contact data can also be used in epidemiology for infectious diseases spreading measurements.

Therefore, the target of the study defines the type of the data required. When talking about close contact measurements we also need to differentiate between different subcategories inside this category. One way to measure close contacts for example can be collecting geo-positions using a GPS, for instance, using a mobile phone. This method does not requires a big deployment but at the same time it has some drawbacks, for example, GPS cannot be used indoors, and it has an error range from 5 to 15 meters. Furthermore, extra data (orientation) will be needed if we want to determine if two users have been in close contact, a requirement for some epidemiology studies.

For more accurate measurements, wireless radio-frequency can be used to detect close contacts between between. A low power signal is emitted that can only be detected in a close range of about 1 - 1.5 meters, thus enabling face-to-face detection when the participants wear the emitter and receptor on the solar plexus. The OpenBeacon platform [1] is the most used platform using this technology for face-to-face close contact detection.

This platform has been previously used for measuring face-to-face contacts in several papers [2][3][4][5] to get information of the frequency and length of close face-to-face contacts in different scenarios (schools, hospital wards, conferences, etc).

As the paper [2] states there are some limitations of this system. The main one is that it requires RFID readers that limit where the measurement can be done. The deployment require RFID tags worn by each participant and RFID Ethernet readers that will read the contact data reported by the tags and forward it to a server where it will be stored.

RFID Ethernet reader require Ethernet connections, therefore a network infrastructure, and a gateway or server where to forward and store the contact data. These requirements limit the possibilities of deployment in some scenarios. Challenging environments, the focus of this paper, like remote locations in Africa, require a more relaxed requirements. Most of them are characterized for not having network infrastructure available, thus making more difficult to deploy this type of RFID readers. Furthermore, measurement of outdoors areas becomes more important

where some RFID readers are not prepared to be deployed in.

These scenarios have not been explored yet in terms of close face-to-face measurements. The analysis of this data could bring improved vaccination protocols as infectious diseases models (that require frequency and length information of close contacts between participants) could be applied on top of this collected network.

We present a new RFID reader with more relaxed requirements that can be deployed outdoors, and does not require network infrastructure or a server. The reader is based on a Raspberry Pi, and therefore has a low power requirements, being able to work with a 7000 mAh battery for more than 10 hours.

We also test the accuracy of the current OpenBeacon platform and propose a more expensive in terms of power consumption solution that can provide more accuracy. This solution is based in an additional signal with more transmission power that will be used for additional spatial location, thus correcting the possible errors of the current RFID tags. A more accurate (in terms of frequency and length) close contact measurement will provide better infectious diseases simulations results.

The paper is divided in two parts. The first part of the paper shows the new RFID reader that enables face-to-face close contact measurements in challenging scenarios. The second part of the paper is dedicated to improve accuracy of the current system.

2. Background

In these section we describe the technologies and architecture behind the current contact measurement tools based on RFID.

2.1. RFID System

RFID (Radio Frequency IDentification) tags have been used in many previous experiments to measure person to person interactions. Particularly, for the RFID devices, the OpenBeacon platform [1] has been widely used in these studies. OpenBeacon is an open platform that uses active RFID devices. It consist in two main elements: The RFID tag, and the RFID reader. The RFID tag is a small device that users wear, usually on the chest (solar plexus point), that constantly sends beacons (air messages) that can be read by other RFID tags and RFID readers. When an RFID tag detects other tags (reads the beacons sent by them), it sends a message to inform that a contact has been produced. These messages are received by RFID readers that forward them to a gateway where these messages are stored.

2.1.1. RFID Tag

An OpenBeacon RFID Tag [6] (Figure 1) is an active RFID tag that transmits signals (also known as beacons or messages) in the 2.45 GHz band using an nRF24L01 transceiver. The nRF24L01 [7] can transmit beacons at different power strengths: -18dBm, -12dBm, -6dBm, and 0dBm. Different power strengths are used to estimate the distance between tags and the RFID readers. When a beacon is received, the power strength that was used to sent the beacon is used to calculate the distance from the reader (this information is inside the beacon or message) and can be used as well for location using trilateration if several RFID readers have been deployed.

Tags can detect beacons from other tags because they are active RFID tags. This feature is used to measure face-to-face contact detections. These contacts are stored in a short memory of the RFID Tag and a report message is sent periodically to the readers with the list of tags contacted.

The tags are flashed with the tagPRO firmware, that provides two different antenna power configurations. The first one uses the nRF24L01 transceiver with any of the four different RF output power configurations in TX mode: -18dBm, -12dBm, -6dBm, or 0dBm. Beacons sent with this configuration will be used for distance calculation. The second type uses the nRF24L01 transceiver plus a resistor that lower the output power making the signal reduce its range to only 0.6-1.2m instead of 5-20m. The resistor is activated when the tag wants to send a proximity beacon, used for close face-to-face contact detection. The low output power value of the signal in this case produces that the signal will not be able to be detected behind the body of the participant carrying because the human body will absorb most of the signal. Thus making the signal or beacons only detectable face-to-face.

Figure 1: RFID Tag [6]

2.1.2. RFID Readers

The RFID ethernet readers [8] detect messages or beacons sent by RFID tags. These messages are formatted and sent to the gateway or server where they will be stored. As ethernet RFID readers do not have large permanent storage, a server is required. The server is configured as the gateway in the network configuration of all the ethernet readers. Figure 2 shows the structure needed.

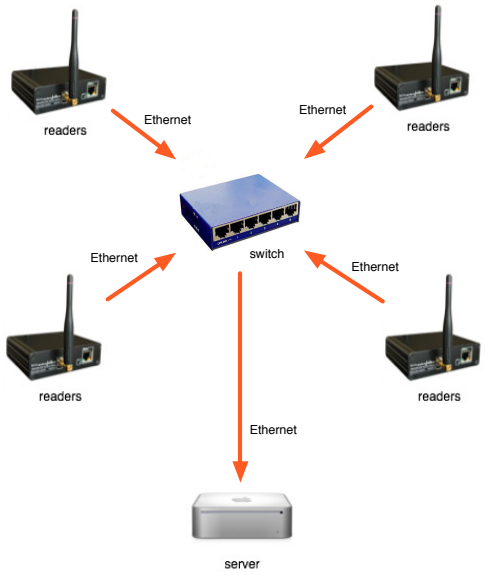


Figure 2: RFID System scheme

These RFID readers require an ethernet connection supporting Power over Ethernet (PoE) to power the reader (or a standard mini-USB power port suppling 5V). They also require an ethernet network infrastructure with hubs and/or routers that connects all the readers with the gateway.

3. Opportunistic RFID Reader

In this paper we present a new opportunistic RFID Reader which new characteristics closely relates to the concepts of opportunistic networks or Delay Tolerant Networks (DTNs). The new RFID reader can work outdoors, does not need network infrastructure, is mobile/portable, and is decentralized (does not requires of a server). It is based on a Raspberry Pi Model B, a tiny computer equipped with a 700 MHz ARM11 CPU, 512 MB of RAM, an Ethernet port, two USB ports, an SD card port for storage, and several GPIO (General Purpose Input Output) pins, between others.

An OpenBeacon USB RFID reader [9] dongle is used with the Raspberry Pi with an equivalent firmware to the one in the ethernet RFID reader. The dongle uses the same nRF24L01 transceiver as the OpenBeacon RFID Tags [6] and therefore it is fully compatible. A script collects the data from the USB RFID Reader via serial port (USB) and process the streaming received containing all messages (or beacons) read by the dongle. These messages are stored to the permanent storage of the Raspberry Pi (an SD card).

The new RFID reader, therefore, supports local storage and does not require a centralized server. A WiFi dongle is also used, eliminating the need of an ethernet network infrastructure. To completely eliminate the need of network infrastructure, even wifi infrastructure, we use Delay Tolerant Networks (DTN). As the main purpose is to use the system in challenging environments, the support of this type of networks will help in some extreme cases.

DTNs support allows to exchange messages between devices without an existing end-to-end path between the sender and the receiver. Thus, opportunistic RFID readers can be deployed isolated from others or in disconnected groups, supporting big delays and disruption on paths when forwarding and/or delivering messages. Depending of the forwarding algorithm used, the messages will be spread to any other reader they contact (epidemic like) or will follow an strategy to only forward a message to selected nodes.

In our prototype, we have configured it to use a data mule approach. Data mules is a particularly case of delay tolerant networks in which, most of the nodes are stationary and a few of them move around, following routes that intersect with stationary nodes. These moving nodes are called ferries or data mules and are in charge of collecting data from the stationary nodes. This method is very used in sensors networks to collect data from sensors for example. Another example is the the postman example in rural regions. The postman carries with him (or in its motorcycle) a node that collects and delivers messages to other nodes in little villages without internet access. People living in these villages, then, can send and receive emails, social networks updates, digital newspapers, etc, without having internet access, in a delay way.

Our prototype works in the same way. We deploy several opportunistic RFID readers over the area we want to measure. When a collection of data is wanted, a data mule goes around where the readers have been deployed and those send a copy of the data collected to it.

We have installed the prototype inside a water proof plastic box (Figure 3) to make it resistant to rain, dust, etc. Thanks to this protection we can deploy the RFID reader outdoors and thus relaxing the restrictions of other RFID readers systems and opening the possibility to take measurements in wider areas.

To make it portable and for easily deployment, we have tested its use with a battery pack. A Raspberry Pi has low energy consumption, it only draws 3.5W. We performed several tests with a battery pack with a capacity of 7000 mAh. The average duration of the battery was around 10 hours, enough for a continuous measurement and collection of data during daylight.

3.1. Test

To test the system proposed we performed an experiment. We deployed 17 Raspberry Pi RFID Readers in the Computer Laboratory of the University of Cambridge (Figures 4, 5, and 6). The Readers were distributed based on the location of the participants and common areas (the cafe, dinning areas, corridors, etc) in each one of the three floors of the Computer Laboratory. Private areas where

Figure 3: Raspberry Pi OpenBeacon Reader

participants were not allow to go or areas where no participant was meant to be were not covered.

Over a period of three days we collected contact and location data from 27 participants, divided on different groups (students, interns and different research staff from different research groups). Each one of the participants was given RFID tags who had to wear during the time they were in the Lab.

As stated in the definition of the prototype, the readers collected all the data and store them into its permanent storage (an SDCard), we collected all the data using a data mule approach going through all the readers location with a data mule who was in charge of receiving all the data from the deployed stationary readers.

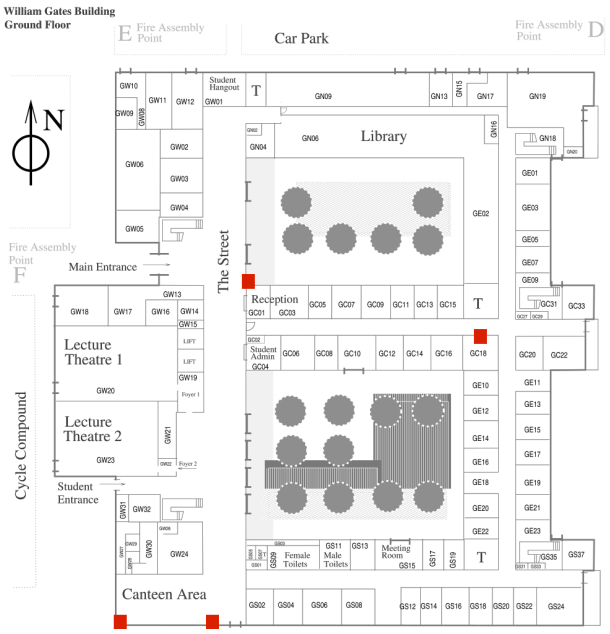


Figure 4: Deployment of the RFID readers on the ground floor

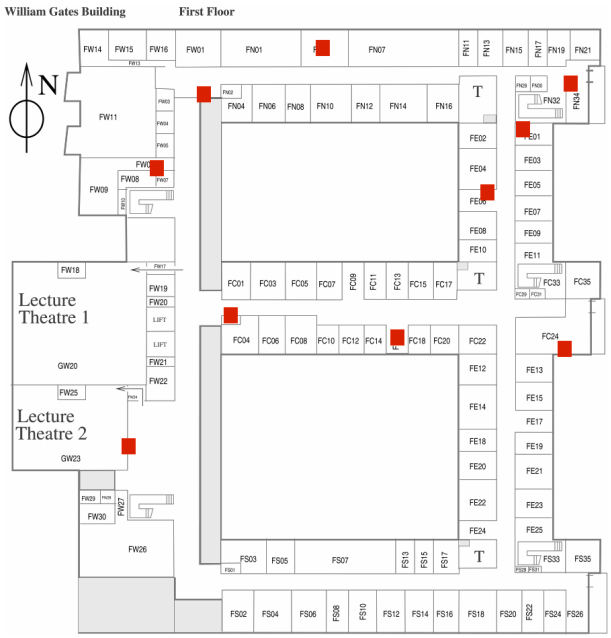


Figure 5: Deployment of the RFID readers on the first floor

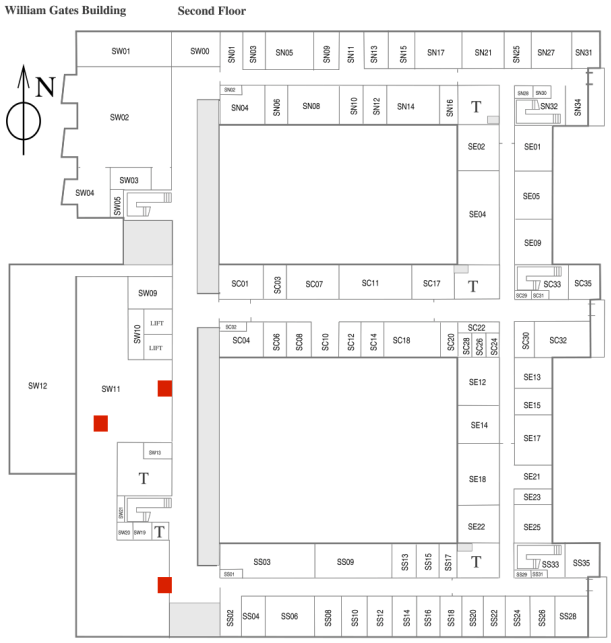


Figure 6: Deployment of the RFID readers on the second floor

The results were collected using a data mule approach with a moving node. The server moving node collected the data when passing next to a deployed opportunistic RFID readers and stored this data on its own storage. After all data was collected, it was aggregated and exported as a network, where nodes represent the participants of the experiment and the edges represent the interaction between two individuals. The edges are weighted by the number of beacons interchanged between them. Figure 7 shows the

network after applying a community detection algorithm. The results correctly show the different groups that participated in the experiment.

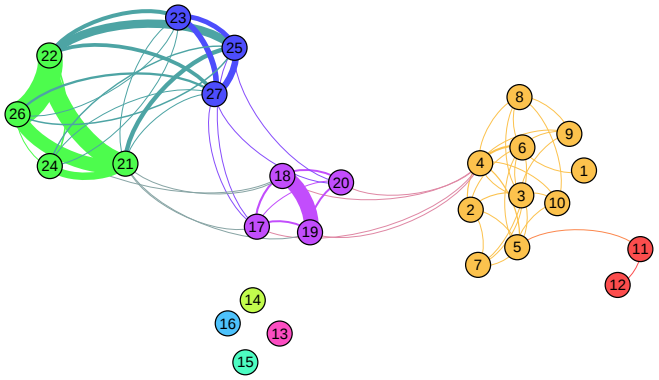
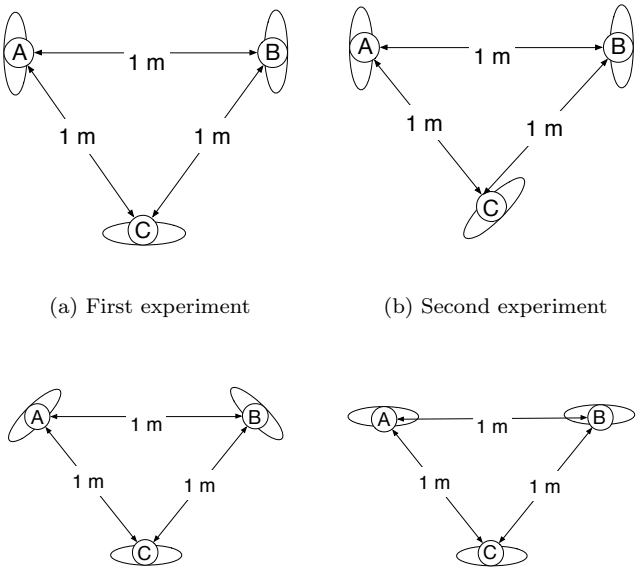


Figure 7: Results from the test using opportunistic RFID readers

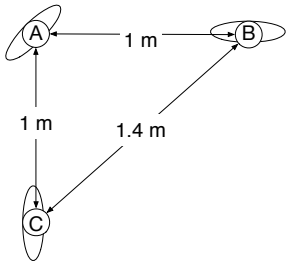


(a) First experiment

(b) Second experiment

(c) Third experiment

(d) Fourth experiment



(e) Fifth experiment

Figure 8: RFID experiments

4. Accuracy

The second part of the paper is an analysis and proposal of improvement of measurements accuracy. We performed a series of tests to measure distance sensitivity, position, signal problems, etc. Some of these tests are shown in figure 8 and figure 9. The first set of tests (figure 8) tests were done with 3 participants, each one of them carrying an RFID tag on the solar plexus. We located the participants in a big room and in fixed positions without allowing them to move during the experiment. We used a distance between them of 1 meter. Previous works in different locations like conferences [2] or primary schools [3] show that more than 85% of contacts have less than 1 minute of duration. We performed tests of around 1 minute of duration with stationary participants. For each one of the tests, we used different positions for the participants to measure the impact of the orientation and location. In figure 8 can be seen the configuration for each one of them.

Authors of previous works [2][3][4] have used the following algorithm to analyze contact data. First, they divide the time into intervals of 20 seconds. Then, they apply an algorithm that considers that two tags tag_a and tag_b have been in contact during a whole interval (20 seconds) if during this interval at least one contact message report has been received.

$contact\_time = Array.new$
$n_{contacts} = 0$
for i in $(1..time).step(20)$ **do**
  if $contacts[i..(i+19)].size > 0$ **then**
    if $contact\_time[ncontacts] == nil$ **then**
      $contact\_time[n_{contacts}] = 20$
    else
      $contact\_time[n_{contacts}] = contact\_time[n_{contacts}] + 20$
    end if
  else
    $n_{contacts} = n_{contacts} + 1$
  end if
end for

This threshold of 20 seconds is considered by the authors the minimum interval of time to have a probability of 99% that no contact have been produced during this interval if no contact report have been received. This means that if we take a tinier interval of 10 seconds and no contact report is received during this interval, it cannot be guaranteed that a contact has not been produced, the probability will be less than 99%. As authors state, this is the best timescale as a faster one do not improves accuracy due to noise issues.

Although this solution is useful in some situations, the 20 seconds timescale may not be enough when a higher

	Exp1	Exp2	Exp3	Exp4	Exp5
A-B	0.66	0.2	0.88	0.11	0.88
B-C	0.83	0.4	1	1	0
C-A	0.83	1	1	0.88	0.33

Table 1: Results of the tests after applying the timescale algorithm. The results show the percentage of the total time of the experiment that the two nodes have reported being in contact.

resolution is wanted. The distribution of the probability for intervals than less than 20 seconds is unknown. The known probabilities therefore are:

- For a timescale of 1 second, the probability that two tags are in contact if a contact report is received during this period of time is 1.
- For a timescale of 20 second, the probability that two tags haven't been in contact if no contact report have been received during this period of time is 1.

After applying this algorithm to the data from the tests performed, the table 1 shows the following results:

In all the tests the participants were still during the duration of the experiment in the position expressed in the figure 8. The results of the table 1 should be theoretically 1 as the two participant were in network range of their RFID tags. Although the timescale of 20 and the algorithm have been applied, some results do not show a continuous contact during the experiment.

5. Improving accuracy

The previous section showed that face-to-face contact measurement can be inaccurate in some cases. Some previous works already mentioned that measure that type of contacts in different environments have a significant amount of contacts of less than 20 seconds long (even more than 75% of all the contacts measured). This can be caused by two possible factors by our understanding.

First, some of these short measured contacts could be just two people passing by each other. Some works may require this type of contacts to have another type of consideration or even not to be taken into account. Lowering the time frame window in the current algorithm (20 seconds) will help in detecting this type of situations.

The second possibility of these short measured contacts could be high effects of noise or other elements, splitting a longer contact into shorter ones (missing face-to-face contact reports).

To increase the accuracy in contacts measurements we propose the use of an additional signal with more power strength, providing spatial information. Data about spatial information between two tags will help to correct possible errors in face-to-face contact data (missed contact reports), and help to reduce the time frame window, the two main problems we have detected that affects accuracy.

5.1. Distance estimation based on signal strength

Distance estimation based on the difference between the transmitter and the receiver signal strength has been used in several previous works [10][11][12][13][14][15]. The signal strength in the transmitter is known and in the receiver is provided by the received signal strength indicator (RSSI) that some wireless devices have. To calculate the distance between a transmitting and a receiving point the Friis equation [10] is used, under perfect conditions:

$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2}$

where P_r is the signal strength in the receiving point, P_t is the transmitting power, G_t is the transmitting antenna gain, G_r is the receiving antenna gain, λ is the wavelength of the transmitted signal, and d is the distance between the receiving and the transmitting point.

As signals are not transmitted in ideal conditions, and the environment where they are transmitted have an impact in the transmission. Reflection, absorption, scattering, or multipath are just some effects that affect the transmission of signals and therefore alter the signal strength received in the receiver. A more realistic equation [10] giving a closer approach to signal propagation in real conditions is the following:

$P_r = P_0 - q10log_{10}(\frac{d}{d_0}) + \alpha$

where P_r is the signal strength at a distance d from the transmitter, P_0 is the signal strength at a distance d_0 from the transmitter, q is the loss exponent that indicates how signal strength decreases through distance and it is dependent of the environment (indoor with walls, free space, outdoor...), and α is a Gaussian distributed random variable with zero mean and standard deviation σ that accounts for the random effect of shadowing.

The loss exponent is greater as more obstacles and other type of effects (reflections, absorptions, etc) have the environments. The usual values are from 2 to 6 ranging from outdoors to indoors, although can arrive to grater than 6 at indoors and lower than 2 in tunnels, as they act as a waveguide.

Most of previous works use this equation to infer location, applying estimation techniques like lateration. Lateration techniques are based in distance measurements from different static points, called anchor nodes. The first thing that has to be done is to "calibrate" all the anchor nodes. This is an offline phase that requires to measure the signal strength received at a distance of 1 meter of an anchor point. With this information, the path loss exponent can be calculated, which is different depending on the environment where the signal has to be propagated. This calibration has to be done for each one of the anchor nodes, as the effects of reflection, scattering, etc can be different on each set. Once all the anchor nodes are calibrated, the distance to each one of them can be calculated based on the RSS (Received Signal Strength) and the equation above. The location of the moving node then, is calculated using the intersection of the n distances measured.

5.2. Two moving signals

The face-to-face contact detection is substantially different from the previous works on distance estimation as in this case we want distance information between two moving nodes instead of a moving node and an anchor node. Therefore, the environment changes, making the loss exponent different under different locations.

Although the location and thus the loss exponent can change, a face-to-face contact usually happens in an static position. This means that during a face-to-face contact, the loss exponent will be constant or have a very low variation.

When a close contact report between two nodes $node_a$ and $node_b$ is received from an RFID tag at time t, we can say that the two tags are separated by less than 1.5 meters of distance. We will collect the received signal strength for the high power signal at that time t which will have the same characteristic (less than 1.5 meters of distance). Therefore, we will have a maximum distance d, a constant loss exponent q, and a received signal strength RSS. If at time t_1 a contact report is not received from an RFID tag, the high power signal will be analyzed and compared to the one collected at time t to see if the two nodes are still in range of < 1.5 meters of distance. $t_1 - t$ should be less than what the original authors consider in their papers as the maximum time after receiving a contact report with non-zero probability. Therefore, for the temporal analysis of the high power signal we will use the received RFID contact reports as ground truth.

Received signal strengths will also give information about the gaussian shadowing when analyzing the signal over time, being able to extract its characteristics (mean, standard deviation).

5.3. Experiment

For confirming the algorithm we have used as a prototype an OpenBeacon tag for close-contact measurements, and a Bluetooth device for high power signal strengths. The reason for doing this is because the nRF2401 radio transceiver does not have a received signal strength indicator, therefore, in case of building a new RFID tag, the developer will have to use another radio transceiver supporting a rssi.

The use of bluetooth based beacons, also commercially called iBeacons, have exponentially increased the last months. This is thanks to the notably decrease of energy consumption of the last version of Bluetooth 4.0, also known as Bluetooth Low Energy or Bluetooth Smart.

Bluetooth is a wireless radio technology standard designed for short distances. It uses the 2.40 GHz ISM band divided in 79 channels for transmission achieving a 1 Mbit/s data rate in its first version, and up to 24 Mbit/s in its last 4.0 version. This standard has been widely used for transmitting data between devices in short range, calling this type of networks, Personal Area Networks (PANs). The typical range of this technology is usually around 5 to 10 meters for Class 2 radio implementations, the most used one. It has also been used for contact detection, specially since the expansion of devices equipped with the version 4.0 of the standard, which includes a new mode (called Bluetooth Low Energy or Bluetooth Smart) that reduces drastically the amount of power required for sending beacons, thus increasing the number of applications.

This fact, together with the adoption of the last Bluetooth standard in the last smartphones from different manufacturers, have produced the appearance of several new devices implementing this wireless technology for several different purposes: distance, location, presence, etc. A lot of tiny bluetooth tags with a very low power consumption have appeared in the market [16][17] with the main goal to attach them to objects and monitor their approximate distance.

As a bluetooth device we have chosen a Samsung Galaxy Note 2 equipped with Bluetooth 4.0. We have also developed an Android application for bluetooth scanning that allows us to get between 1 to 4 rss readings per second.

We performed an experiment to show the feasibility of the proposal, figure 9 shows the configuration.

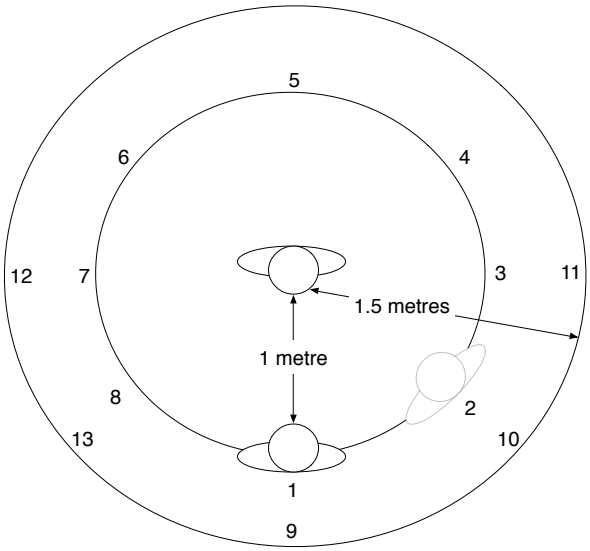


Figure 9: Experiment outdoors measuring RFID + Bluetooth

The two participants used an RFID and a Bluetooth devices. The configuration (firmware) used in the RFID tag is the standard used in another previous experiments (explained in previous sections) and participants wore it on the solar plexus. We used our opportunistic RFID readers, already described in this paper. The participants also carried the Bluetooth device under the location of the RFID tag.

While the participant A, in the middle, did not move during the whole experiment, the participant B moved from one location to another (marked with numbers in the figure) with a 2 minutes stay in each one of the locations and a 30 seconds gap between positions. Participant B

always looked at participant A (orientation), while participant A always maintained the same position. We tested two distance: 1 and 1.5 meters in different locations.

Figure 10 shows the data collected in the experiment. In green can be seen the time when a contact report beacon was received by either participant A or participant B. The red crosses mark the received signal strength (RSS) in dBm between participant A and B. The vertical lines separate each one of the experiments (locations marked in the figure 9) with a duration of 120 seconds.

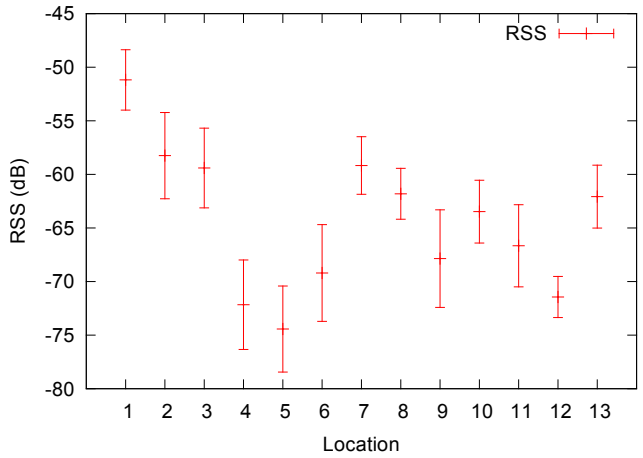


Figure 11: Received Signal Strength for each one of the locations in the experiment

5.4. Algorithm

Let $Contacts$ be the set of all the contacts c reported by the RFID tags between two tags $node_a$ and $node_b$ between the start of the experiment at time t_0 until the end of the experiment t_n. Lets define as well $RSSs$ as the set of all the Received Signal Strengths rss of the high power signal of $node_b$ from $node_a$. For each contact c in $Contacts$ at time t we have $rss\alpha(c)$, which contains all the high power (bluetooth) received signal strengths from $t-1$ to $t+1$ for a contact c.

For each second during the measured time from t_0 to t_n, we will calculate the average of the rss (received signal strengths) for each t‘, therefore $rss(t$‘$)$. We will compare $rss(t$‘$)$ with $rss_\alpha(c)$ only if c has been produced at time $t < t$‘$ + 40$ & $t > t$‘$ - 40$. The reason of having a 40 seconds interval temporal analysis is because this is the equivalent to two time frames in the previous algorithm used to analyze data (if a contact has been detected in the first second of the time frame 1, we do not discard a possible ongoing contact until the next time frame has passed: a total of 39 seconds). If after two time frame windows a contact c has not been received there is a very low probability that the close face-to-face contact is still ongoing.

In case that c has been produced at time $t < t$‘$ + 40$ & $t > t$‘$ - 40$, if $rss_{avg}(t$‘$) \leq rss_\alpha(c)_{mean} + rss_\alpha(c)_{std}$ then we say that at time t‘, $node_a$ and $node_b$ have been in contact. A greater received signal strength means closest distance between peers.

Position	Total time connected	Num contacts	Avg time	Std
1	60 (20, 40)	2	30	14
2	120	1	120	0
3	100 (20, 80)	2	50	42
4	0	0	0	0
5	0	0	0	0
6	60	1	60	0
7	100 (20, 80)	2	50	42
8	120	1	120	0
9	80 (60, 20)	2	40	28
10	100	1	100	0
11	60 (40, 20)	2	30	14
12	100 (80, 20)	2	50	42
13	120	1	120	0

Table 2: Results of the tests after applying the timescale algorithm (all results in seconds)

If we analyze figure 11, we can clearly see that during a contact between two nodes the received signal strength is similar (it has very similar average and standard deviation) as the nodes does not move during it. The standard deviation is always large in all the experiments, due to Gaussian distributed random variable with zero mean that accounts for the random effect of shadowing. The signal then, only changes when, (a) the environment changes (signal effects), or (b) one of both of the nodes move. If the two nodes get closer the received signal strength will get greater values, if the two nodes get farer the received signal strength will get lower values.

In the old algorithm to detect that two nodes are no longer in contact, we have to wait a complete time frame (20 seconds). With this new algorithm we can detect sudden changes much faster, as the received signal strength will heavily change getting much lower values meaning longer distance between nodes.

5.5. Results

After applying the algorithm described in previous section, the results are plotted in figure 12. The blue signal represents when the two nodes have been in contact based on the new algorithm. The table 2 show the results using the previous algorithm used by other authors. The expected results observable and measured in the experiments should be a “Contact rate” of 1 in all the experiments with only 1 contact of 120 seconds of duration. Table 3 show the results using the new algorithm.

In figure 13 we can see the mean and standard deviation time between contact reports RFID beacons. The figure shows a mean of more than 10 seconds from most

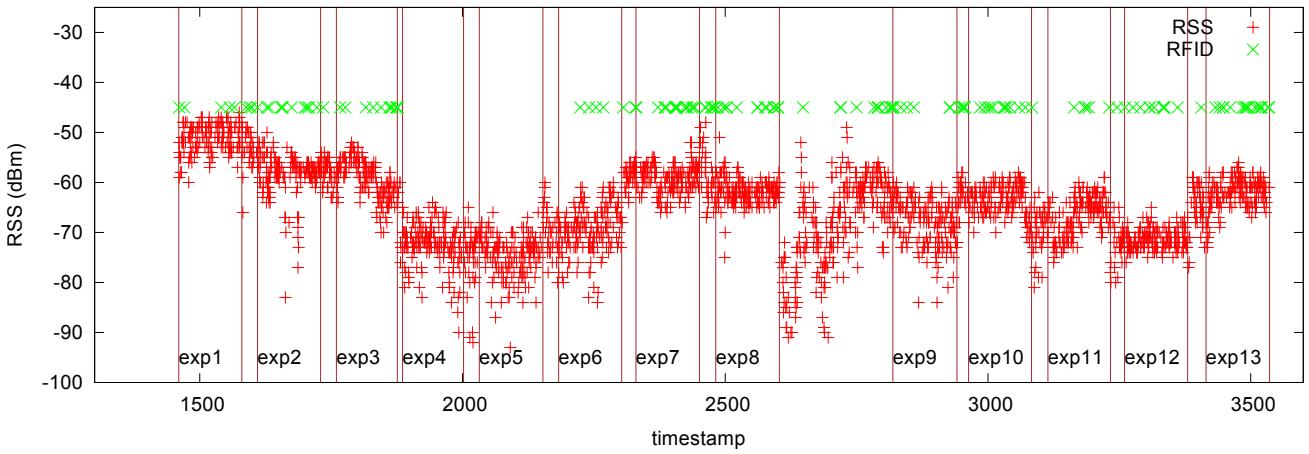


Figure 10: Received signal strength and RFID contact reports between the two participants during all the experiments

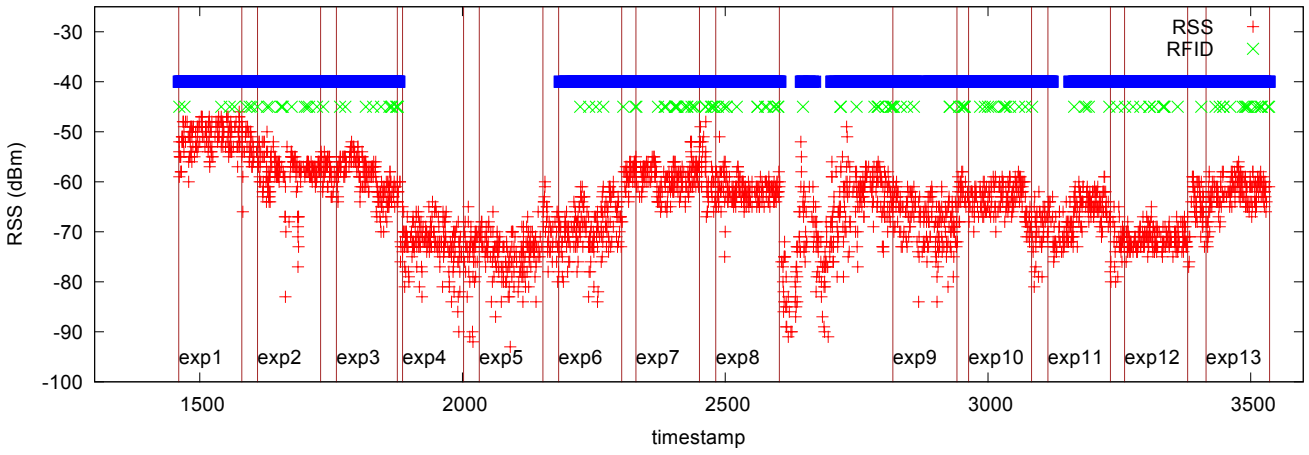


Figure 12: Received signal strength, RFID contact reports (and proposed algorithm in blue) between the two participants during all the experiments

Position	Total time connected	Num contacts	Avg time	Std
1	100 (20, 80)	2	50	42
2	120	1	120	0
3	120	1	120	0
4	0	0	0	0
5	0	0	0	0
6	120	1	120	0
7	120	1	120	0
8	120	1	120	0
9	100 (80, 20)	2	50	42
10	120	1	120	0
11	80	1	80	0
12	120	1	120	0
13	120	1	120	0

Table 3: Results of the tests after applying the new algorithm using the same timescale (all results in seconds)

Position	RSSI Mean	RSSI Std
1	-51.18	2.82
2	-58.25	4.03
3	-59.40	3.72
4	-72.16	4.18
5	-74.43	4.01
6	-69.2	4.52
7	-59.17	2.69
8	-61.81	2.38
9	-67.86	4.56
10	-63.47	2.94
11	-66.66	3.83
12	-71.44	1.92
13	-62.07	2.93

Table 4: Received signal strength based on the position in the experiment

of the experiments. Table 4 shows the mean and standard deviation received signal strength for different locations in the experiment. We can see how the body acts as signal absorber and therefore the rss received on not face-to-face experiments are much lower. We can also appreciate that a longer distance between the nodes produces a lower rss.

6. Discussion

As the results sections shows in tables 2 and 3, after applying the new algorithm and maintaining the same 20 seconds timeframe window, the results are much closer to the real ones (120 seconds total time connected where the two nodes are facing each other and only one contact). Furthermore, we can also obtain more fine grained results by lowering the timeframe windows from 20 seconds to

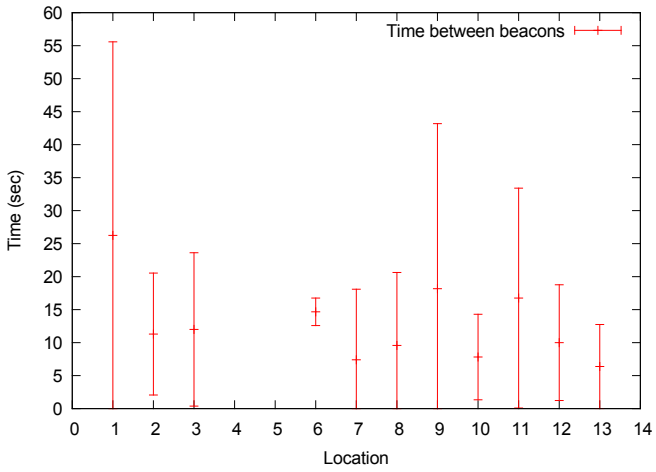


Figure 13: Time between contact reports RFID beacons for each one of the locations in the experiment

Position	Total time connected	Num contacts	Avg time	Std
1	100 (20, 80)	2	50	42
2	110	1	110	0
3	120	1	120	0
4	0	0	0	0
5	0	0	0	0
6	120	1	120	0
7	120	1	120	0
8	120	1	120	0
9	100 (80, 20)	2	50	42
10	110 (50, 60)	2	55	7
11	70	1	70	0
12	120	1	120	0
13	110	1	110	0

Table 5: Results of the tests after applying the new algorithm using the same timescale (all results in seconds)

a lesser value. If we relax the timeframe window to 10 seconds, we will get the results from table 5.

Although the results are worst than using a 20 seconds time frame window, they are still better than the results obtained by using only RFIDs thanks to the additional signal that gives added spatial information. Having a lower time frame window allows to help to decide whether a short contact consists in only two nodes passing by instead of a face-to-face contact. This type of information can be relevant under some type of experiments, for example epidemiology ones where only long contacts should be considered. Furthermore, it also gives us more accurate measurement of the length of the contact than the previous method, where the minimum length of a contact was of 20 seconds.

Also, the recovery of missed contacts should be easier.

	Total time	Num contacts	Avg time	Std
Real	1200	10	120	0
RFID	1020	17	60	39
RFID+BT	1240	13	95	38

Table 6: Different results from different measurements methods (all results in seconds)

As an example we can take the experiment 1; It is always measured as a 2 contacts no matter the method used (tables 2, 3, or 5). The correct measurement should be only 1 contact of length 120 seconds. Using only RFID (table 2 and figure 10) the gap between these two detected contacts is of 60 seconds, adding the extra spatial information from the high power signal, in table 5 the gap between these two contacts is of 20 seconds making it easier to recover that information and finally counting only 1 contact of 120 seconds.

The results can change a lot, for example, lets take tables 2 and 3. If we calculate the number of contacts and the average contact time in table 2 for all the experiments we get as a result: 17 contacts with an average contact time of 60 seconds and a standard deviation of 39 seconds, making a total contact time of 1020 seconds. A perfect measurement would have measured 10 contacts with an average contact time of 120 seconds and a standard deviation of 0 seconds, making a total contact time of 1200 seconds. Adding more spatial information with the high power signal we get the following results: 13 contacts with an average contact time of 95 seconds and a standard deviation of 38 seconds, making a total contact time of 1240 seconds; closer to the ideal ones. Table 6 sums up all these results showing the significant difference that could impact in the results some algorithms applied to the generated graph.

As a final remark, it is important to say that a high accuracy when doing this type of experiments is very hard to achieve due to the several effects that can affect a signal. For example, experiment 6 (figure 9) shouldn't have reported any contact but it did, probably to some reflection or other signal effect. One of the goals of this paper then is to improve accuracy of the already good results from the RFID tags contacts detection.

Some previous works already mentioned have a significant amount contacts of 20 seconds (or less) long. This can be caused by two possible reasons by our understanding. First, some of these short measured contacts could be just two people passing by each other. This type of contacts should have another type of consideration or even not taken into consideration for some type of analysis. To better detect this type of contacts and be possible to discard them, we have decreased the timeframe window to 10 seconds, therefore now the shortest contact that can be detected has a duration of 10 seconds instead of 20. The second possibility of these short measured contacts could be high effects of noise or other signal effects, splitting a longer contact into shorter ones. This example can be seen in the experiment 1 in tables 2, 3, or 5. It can be seen in these same tables that adding the new spatial information given by the extra additional signal, it helps to reduce the gap between these contact splits and thus be able to recover the long contact more easily. Furthermore, it is also able to reduce the number of splits of contacts as can be seen in these tables for experiments 3, 7, or 12.

6.1. Relevance of the measurements for some studies

Stehl et al [3] published a work measuring face-to-face contacts in a primary school using RFIDs. The results showed an average of 323 contacts per child per day, with an average contact time of 33 seconds. It has to be taken into account that the minimum time measurable in a contact is 20 seconds given the time frame window used in their algorithm. On another previous work from Glass et al [18] also measuring close contact interactions (less than 1 meter), the authors used a survey method. The results showed an average of 4.43 contacts per day per child with an average contact time of 1 hour.

The difference between the two studies are remarkable not only in the number of contacts but in the duration of these. One of the factor to this is that in the second study the survey only asked for contacts with a "recognizable length of time", thus getting only an average of 4.43 contacts per day per child. The automatization of contact detection using RFID allows to get a huge amount of contact information, even those contacts that weren't noticed by the participant.

From the other side, if we study the data from day 1 published from Stehl et al, we can see that only 20 from the 5539 edges in the graph (0.36%) have a total contact time of more than 1 hour. Furthermore, only 106 edges (1.9%) have an average contact time longer than 1 minute.

The contact time between two participant is relevant for this type of studies as it may influence in the possibilities of contagious of an infection disease. For this reason we believe that improving accuracy to measurements is important.

7. Related work

Bluetooth technology has been previously used to measure contact interaction between participants: iMotes were used in the Haggle project [19] and mobile phones were used in the MIT Media Lab [20]. These works do not measure face-to-face interaction but contact between participants with a wider distance, 5 to 10 meters in the case of the Nokia 6600 used in the MIT Media Lab experiment [20]. As a difference to the OpenBeacon RFID tag, the iMote (or other Bluetooth motes used for tracking contacts) do not use a low power beacon system that only measures close 1.5 meters (or less) face-to-face contacts but they can be used either indoor and outdoors.

More sophisticated image processing based contacts tracking systems using satellite images or video surveillance systems require a more complex deployment and expensive processing systems. Several cameras need to be installed to cover all the spots in an area and they also require complex image processing systems.

The MIT Media Lab has also used IR sensors to detect face-to-face contacts using the Sociometric Badges [21][22] (apart from other elements detected by the badge like speech). These badges only offer ± 15 degrees visibility cone and the IR technology requires direct line of sight.

Some localization systems can also be used to detect contacts if two nodes are detected in the same area. Some systems like RADAR [23], Horus [24], DIT [25], or ekahau [26] offer indoor localization techniques based on WLAN received signal strength (RSS). As explained previously in the paper, this type of systems require an offline phase to calibrate the system and most of them have an accuracy of a few meters [27] indoors. Furthermore, most of these options does not offer node orientation required for face-to-face interaction.

Between the few localization systems that also offer orientation we can find COMPASS and Active Bats. COMPASS[28], also based on WLAN RSS, uses a compass to add orientation information to improve accuracy by taking into account the effects of the human body that blocks signals by absorption. COMPASS is able to reduce the distance error to 1.7 meters but still needs an offline phase to calibrate the system and has only been tested indoors. Active Bat [29] is able to know orientation and location but requires a large number of sensors deployed (for high accuracy) on the ceiling of the indoor location. Therefore, this system does not works on outdoors locations.

The most common outdoor localization technologies include GPS/GLONASS, with an accuracy of a few meters, and GSM localization with much higher error rate. These systems do not have enough accuracy for measuring close face-to-face contacts interactions.

7.1. Filters

Kalman and Particle filters [30] have been widely used for noise correction based on pre-defined models, and for prediction. The contact model to be used can be different depending the scenario we want to measure (outdoors, indoors, schools, hospitals, etc) as the contact patters will change from each scenario as well as the type of signal effect depending of the location. If the user can identify the model, she will be able to apply a Kalman filter to reduce false negatives. Other works use Particle filters to estimate location. These works use the aggregation of several sensors signals deployed in the scenario. This data can give an approximate prediction of the location of a node at $t + 1$. Some works [31] have also used particle filters together with received signal strength for localization purposes. In the case of face-to-face interaction measurements, anchor nodes are not used (for distance between nodes measurements), neither the system is calibrated in an offline phase as environment changes when the nodes move to different areas. Thus, the distance between nodes is unknown even when the system receives a contact report from one of the tags. In this case, the unknown variable of distance between two nodes is relaxed to the condition that the two nodes are in less than 1.5 meters of separation.

8. Conclusion

In this paper we have seen how current methods for measuring close face-to-face encounters sometimes suffer from accuracy problems. We have proposed to use a supplementary no low power signal for getting additional spatial information.

The main purpose is to help reduce the face-to-face contacts splits and provide a easier recovery of the loss of signals from RFID. The RFID beacons are used as a face-to-face contact detector, while the additional signal is used to get spatial information between the two nodes and detect if the distance between the two have changed while there was no RFID contact report beacon.

Furthermore, we have developed a new opportunistic RFID reader based on a Raspberry Pi and using wireless networks supporting delays and disruptions. These new characteristics allows deployment of these readers in more challenging environments.

9. Acknowledgments

The research is part funded by EU grant FP7-ICT-257756, FP7-ICT-318398, and EPSRC grant EP/H003959.

References

[1] Openbeacon platform. `http://www.openbeacon.org/`.

[2] Cattuto C, Van den Broeck W, Barrat A, Colizza V, Pinton JF, et al. (2010) Dynamics of person-to-person interactions from distributed rfid sensor networks. PLoS ONE 5: e11596.

[3] Stehl J, Voirin N, Barrat A, Cattuto C, Isella L, et al. (2011) High-resolution measurements of face-to-face contact patterns in a primary school. PLoS ONE 6: e23176.

[4] Isella L, Romano M, Barrat A, Cattuto C, Colizza V, et al. (2011) Close encounters in a pediatric ward: Measuring face-to-face proximity and mixing patterns with wearable sensors. PLoS ONE 6: e17144.

[5] Vanhems P, Barrat A, Cattuto C, Pinton JF, Khanafer N, et al. (2013) Estimating potential infection transmission routes in hospital wards using wearable proximity sensors. PLoS ONE 8: e73970.

[6] Openbeacon tag. `http://www.openbeacon.org/OpenBeacon\_Tag`.

[7] nrf24l01 datasheet. `http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01`.

[8] Openbeacon ethernet reader. `http://www.openbeacon.org/EasyReader`.

[9] Openbeacon usb reader. `http://www.openbeacon.org/OpenBeacon\_USB`.

[10] Mao G, Fidan B, Anderson BD (2007) Wireless sensor network localization techniques. Computer Networks 51: 2529 - 2553.

[11] Blumrosen G, Hod B, Anker T, Dolev D, Rubinsky B (2010) Continuous close-proximity rssi-based tracking in wireless sensor networks. In: Body Sensor Networks (BSN), 2010 International Conference on. pp. 234-239. doi:10.1109/BSN.2010.36.

[12] Lau EEL, Chung WY (2007) Enhanced rssi-based real-time user location tracking system for indoor and outdoor environments. In: Convergence Information Technology, 2007. International Conference on. pp. 1213-1218. doi:10.1109/ICCIT.2007.253.
[13] Blumrosen G, Luttwak A (2013) Human body parts tracking and kinematic features assessment based on rssi and inertial sensor measurements. Sensors 13: 11289–11313.
[14] Blumrosen G, Hod B, Anker T, Dolev D, Rubinsky B (2013) Enhancing rssi-based tracking accuracy in wireless sensor networks. ACM Trans Sen Netw 9: 29:1–29:28.
[15] Chung WY, Lee BG, Yang CS (2009) 3d virtual viewer on mobile device for wireless sensor network-based {RSSI} indoor tracking system. Sensors and Actuators B: Chemical 140: 35 - 42.
[16] Button trackr. http://www.button-trackr.com/.
[17] Tile. http://www.thetileapp.com/.
[18] Glass LM, Glass RJ (2008) Social contact networks for the spread of pandemic influenza in children and teenagers. BMC Public Health 8: 61.
[19] Hui P, Chaintreau A, Scott J, Gass R, Crowcroft J, et al. (2005) Pocket switched networks and human mobility in conference environments. In: Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking. New York, NY, USA: ACM, WDTN '05, pp. 244–251. doi:10.1145/1080139.1080142. URL http://doi.acm.org/10.1145/1080139.1080142.
[20] Eagle N, (Sandy) Pentland A (2006) Reality mining: Sensing complex social systems. Personal Ubiquitous Comput 10: 255–268.
[21] Olguín-Olguín D, Pentland A (2010) Sensor-based organisational design and engineering. International Journal of Organisational Design and Engineering 1: 69–97.
[22] Olguin D, Waber B, Kim T, Mohan A, Ara K, et al. (2009) Sensible organizations: Technology and methodology for automatically measuring organizational behavior. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 39: 43-55.
[23] Bahl P, Padmanabhan V (2000) Radar: an in-building rf-based user location and tracking system. In: INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. volume 2, pp. 775-784 vol.2. doi:10.1109/INFCOM.2000.832252.
[24] Youssef M, Agrawala A, Udaya Shankar A (2003) Wlan location determination via clustering and probability distributions. In: Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on. pp. 143-150. doi:10.1109/PERCOM.2003.1192736.
[25] Brunato M, Battiti R (2005) Statistical learning theory for location fingerprinting in wireless {LANs}. Computer Networks 47: 825 - 845.
[26] ekahau. http://www.ekahau.com/.
[27] Liu H, Darabi H, Banerjee P, Liu J (2007) Survey of wireless indoor positioning techniques and systems. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 37: 1067-1080.
[28] King T, Kopf S, Haenselmann T, Lubberger C, Effelsberg W (2006) Compass: A probabilistic indoor positioning system based on 802.11 and digital compasses. In: Proceedings of the 1st International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization. New York, NY, USA: ACM, WiNTECH '06, pp. 34–40. doi:10.1145/1160987.1160995. URL http://doi.acm.org/10.1145/1160987.1160995.
[29] Active bat. http://www.cl.cam.ac.uk/research/dtg/attarchive/bat/.
[30] Ristic B, Arulampalam S, Gordon N (2004) Beyond the Kalman filter: Particle filters for tracking applications. Artech house.
[31] Chen X, Edelstein A, Li Y, Coates M, Rabbat M, et al. (2011) Sequential monte carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements. In: Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on. pp. 342-353.