Fingerprint Indoor Position System Based on Bitcloud and Openmac

José A. Gómez, A. Verónica Medina, Enrique Dorronzoro, Octavio Rivera and Sergio Martín
Departamento Tecnología Electrónica,Seville University, ETSI-INF, Avda. Reina Mercedes s/n, Seville, Spain Email:
(jgomezdte, vmedina)@us.es,(enriquedz, octavio)@dte.us.es,smartin@us.es

***Abstract*—This paper presents a research and a development of a fingerprint-indoor-positioning system using the Received Signal Strength Indication (RSSI) of a Wireless Sensor Network (WSN). The WSN implementation is based on two different protocol stacks: BitCloud and OpenMAC, a certified ZigBee Compliant Platform (ZCP) and an IEEE 802.15.4 embedded software implementation respectively, both from Atmel, and the system uses two different fingerprint algorithms, Simple and Centroid. A comparative analysis of both algorithms using both protocol stacks implementations have been performed to ascertain the best WSN protocol stack and the best algorithm for positioning purposes.**

***Index Terms*— IEEE 802.15.4, RSSI, Centroid, Indoor position, ZigBee, WSN, BitCloud, OpenMAC**

I. INTRODUCTION

WSNs are present in many applications. Examples of WSN applications are found in Ambient Living [1], [2], [3], [4] or Smart building [5], [6], [7], [8], [9] researching fields for solving data acquisition process. Depending on its applications, ambient or user sensors and actuators can be used for making decisions.

The knowledge of a subject's position is very useful in these kinds of systems because depending on it the decisions to be made are different. As stated in [11] and [12], an amount of indoor location tracking systems have been proposed in the literature, based on Radio Frecuency (RF) signals, ultrasound, infrared, or some combination of modalities.

Using RF signal strength, it's possible to determine the location of a mobile node with an acceptable accuracy. Given a model of radio signal propagation in a building or other environment, received signal strength can be used to estimate the distance from a transmitter to a receiver, and thereby triangulate the position of a mobile node. However, this approach requires detailed models of RF propagation and does not account for variations in receiver sensitivity and orientation.

An alternative approach is to use empirical measurements of received radio signals, known as RSSI, Receiver Signal Strength Indicator, to estimate location. By recording a database of radio ''signatures'' along with their known locations, a mobile node position can be estimated by acquiring the actual signature and comparing it to the known signatures in the database, also known as fingerprints. A weighting scheme can be used to estimate location when multiple signatures are close to the acquired signature.

All of these systems require the signature database to be manually collected prior to system installation, and rely on a central server (or the user's mobile node) to perform the location calculation. Several systems have demonstrated the viability of this approach, one of those is MoteTrack [11], [12].

Motetrack's basic location estimation uses a signature based approach that is largely similar to RADAR [10] that obtains a 75th percentile location error of just under 5 m, but in MoteTrack decreased the location error by 1/3.

We have implemented a similar system to MoteTrack, a signature-based localization scheme, but using other motes, Meshnetics´ ones (http://www.meshnetics.com/), that use different RCB (microcontroller and transceiver) and, also, different software, the BitCloud Stack, [13], a ZigBee PRO certified platform. (Atmel acquires MeshNetics´ ZigBee Intellectual Properties). The BitCloud-stack system has been tested and the same precision as MoteTrack has been obtained, but an amount of drawbacks have been found while its implementation was performed.

These drawbacks will be exposed later but in order to solve them, we have been working in the same way by using other WSN stack called OpenMAC, an IEEE 802.15.4 MAC level implementation from Meshnetics too [15] instead of BitCloud. This lower level protocol has enabled to implement applications by taking control of the RSSI measurements.

Two different fingerprint positioning algorithms, Simple and Centroid, have been implemented too in a desktop application, and a comparative analysis has been made in order to study the improvements from the OpenMAC protocol stack relative to the BitCloud one as well as from the first algorithm to the second one.

In Section 2 an overview of the system is presented. In Section 3 the used hardware is shown BitCloud Implementation is explained in Section 4. OpenMAC solution is presented in section 5. Finally conclusions are established in section 6.

II. SYSTEM OVERVIEW

An overview of the system is shown in Fig. 1. In our system, a building or an area is populated with a number

doi: 10.4304/jnw.8.1.37-44

of MeshNetics´s motes acting as fixed nodes, one of them acting as coordinator, C making up the WSN.

Fixed nodes send to C periodic beacon messages, beacon2. Each beacon2 sent by a fixed node, consists of an n-tuple of the format {MobileID, RSSI}, where n is the number of mobile nodes. MobileID is a unique identifier of a mobile node, and RSSI is the received signal strength from the last beacon message, beacon1, sent by aforementioned mobile node and received by the fixed node.

The location estimation problem consists of a two-phase process: an offline collection of reference signatures followed by an online location estimation. As in other signature-based systems, the reference signature database (the off-line phase) is acquired manually by a user with a mobile node and a PC connected to C. The reference signature database consists of a number of reference signatures. Each reference signature, shown as black dots in Fig. 1, is formed by a set of signature tuples of the format {sourceID, meanRSSI}, where sourceID is the fixed node ID and meanRSSI is the mean RSSI of a set of beacon messages received over some time interval. Each signature is mapped to a known location by the user acquiring the signature database (P1-P5 in Fig. 1).

In the online phase, given a mobile node's received signature, s, received from the fixed nodes, and the reference signature set R, the mobile node's location can be estimated applying one of the two algorithms described below.

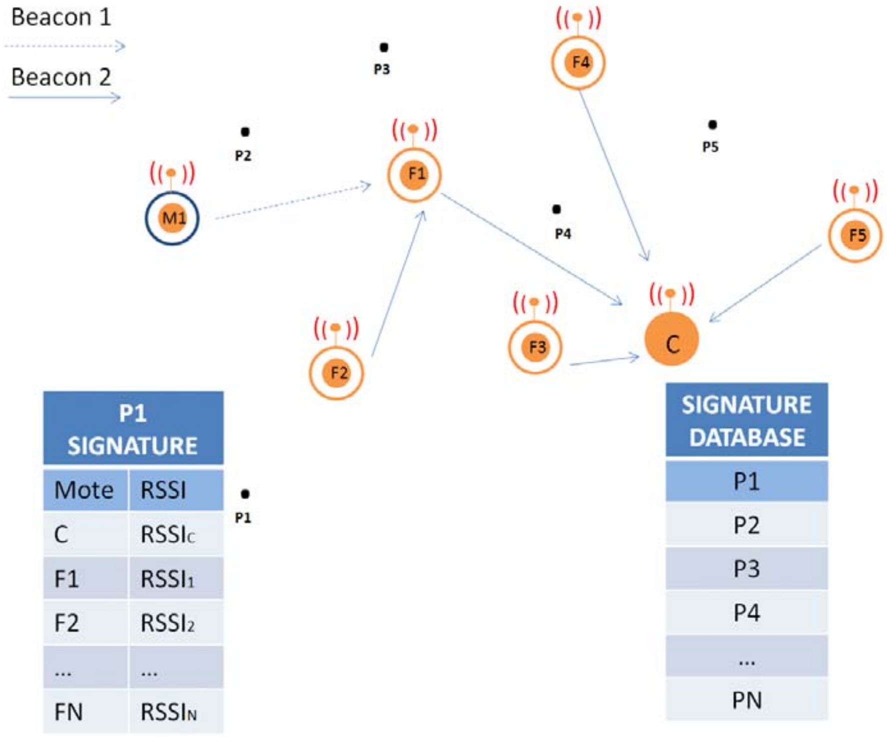

Figure 1. System Overview. M1 is a mobile node, F1-F5 are fixed nodes, and C is the coordinator, also a fixed node. M1 periodically sends a beacon message, beacon 1, to inform the others node that is present, all fixed node that receives it, save the RSSI of that message in a table. Fixed node periodically sends a message to C, beacon 2, to inform about the RSSI that they receive from mobiles node, M1 in this case.

The first step is to compute the signature distances, from s to each reference signature $r_i \in R$. We employ the Manhattan distance metric,

$$M(r,s) = \sum_{t \in T} \left| RSSI(t)r - RSSI(t)s \right| \qquad (1)$$

where T is the set of signatures tuples presented in both signature, RSSI(i)r is the RSSI value in the signature appearing in signature r_i and RSSI(i)s is the RSSI value in the signature appearing in signature s.

Given the set of signature distances, the location of a mobile node can be calculated in several ways applying one of the following fingerprint mechanisms.

A. *Simple Algorithm*

In this algorithm the location point will be one of the stored in the fingerprints database, where associated signature Manhattan distance is the lower from the one obtained in the online phase.

$$\min(M(r,s)) \qquad (2)$$

It must be noticed that through single fingerprint algorithm, it's only pretended to locate in rooms (not in hallways or courtyard) and only with room-level accuracy.

B. *Centroid Algorithm*

Centroid algorithm is similar to the previous one but considers the centroid of the set of signatures within some ratio of the nearest reference signature. Given a signature s, a set of reference signatures R, and the nearest signature $r^* = \arg\min r \in RM(r,s)$ we select all reference signatures r ∈ R that satisfy

$$\frac{M(r,s)}{M(r^*,s)} \qquad (3)$$

for some constant c, empirically-determined. The geographic centroid of the locations of this subset of reference signatures is then taken as the mobile node's position. Small values of c work well, generally 1.1 or 1.2.

When a more completely localization is necessary, not only locating in places where a point is previously calculated and saved into the fingerprint database, centroid algorithm is a better solution. Centroid algorithm is able to locate in hallways or courtyard, even in the entrance of each room.

III. HARDWARE USED

The system is composed of a WSN to acquire RSSI and a PC system where the location estimation is made via a positioning desktop application. This system is based on low-power, embedded wireless devices, MeshNetics´s sensor "motes" called MeshBean2.

The advantage of this platform over other motes is that it's equipped with leds, buttons, sensors, and extra sensors could be easily connected that might be used for different purpose applications for indoor position system, ambient living or smart buildings, if the application requires them, so for prototyping, those boards works quite well. They also have a USART accessible by a USB connector, so a PC can be connected via USB port, emulating a COM port, for both, programming and receiving information, in this case, beacons and sensor values.

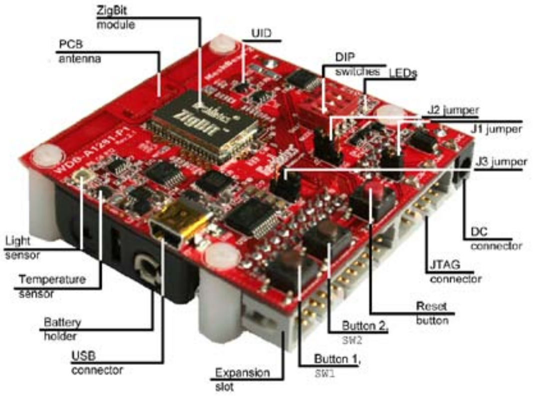

Figure 2. Meshbean development board.

Other advantage of this mote is that the supplier has developed the ZigBee RFC4 stack architecture [14] in a software pack called BitCloud Stack and also an IEEE 802.15.4 MAC level implementation, in a software pack called OpenMAC.

A MeshNetics´s mote is shown in Fig. 2, in this case, it has got an integrated PCB antenna, but we have used others that haven't, it affects only the range of coverage. This mote has a MCU wireless, called ZigBit, a compact 802.15.4/ZigBee module featuring record-breaking range performance and exceptional ease of integration. It integrates both the ATmega1281 microcontroller and AT86RF212 transceiver of ATMEL (www.atmel.com) so the AVR tools are necessary for programming purposes.

IV. BITCLOUD IMPLEMENTATION

BitCloud is a full-featured ZigBee PRO stack that supports a reliable and scalable wireless applications running on Atmel wireless platforms. In ZigBee there are three kinds of devices, each one having its own purpose:

1. Coordinator (C): A full function device (FFD) that it is in charge of creating the PAN (Personal Area Network) and typically is the point of the WSN (Wireless Sensor Network) to acquire all sensors information from all the other motes to be shown in a computer. The used icon which represents this device is a filled circle, Fig. 1 shown one.

2. Router (R): A FFD that it is in charge of routing when the range of coverage requires this capability, so it is possible to have dynamic topologies. The used icon which represents this device is a small filled circle inside a circle, Fig. 1 shown six ones.

3. End device (ED): A reduced function device (RFD) that is always slept (to reduce consumption) and only wakes up to do a specific task, for instance, to send sensor information to the WSN, typically directed toward C. The used icon is a not filled circle; this is, like the R icon in Fig. 1, but no filled circle inside.

So a ZigBee WSN is composed of one C, many EDs and many Rs. Each kind of devices can receive what the other transmit if they are in the same range of coverage, because the transmission media is shared by all one, but not all the received information is processed (the explanation of why this is that way is out of the scope of this paper).

As explained in the previous section, to determinate the position, we require two kinds of beacons, beacon1 and beacon2. Beacon1 is used to inform other devices that a mobile mote is present and beacon2 is used to inform C the RSSI value that fixed mote has received from a mobile one for getting location estimation.

To send both beacons in BitCloud Stack, the information saved in a table, called neighbour table, at the network layer of a certain mote is used. This table has registered all the FFD, this is, C or R that are in the same range of coverage of the mote and, for each one, it registers the RSSI value of the received signal from the mentioned mote. Periodically, a FFD device sends a Zigbee Network layer message to inform other that is in the PAN, so that message is used by neighbor motes to measure the RSSI value of the received signal and to save it in their own neighbor table. So beacon1 is sent automatically by the protocol stack. As only FFD sends this kind of message the mobile motes have to be R, as shown in Fig. 1.

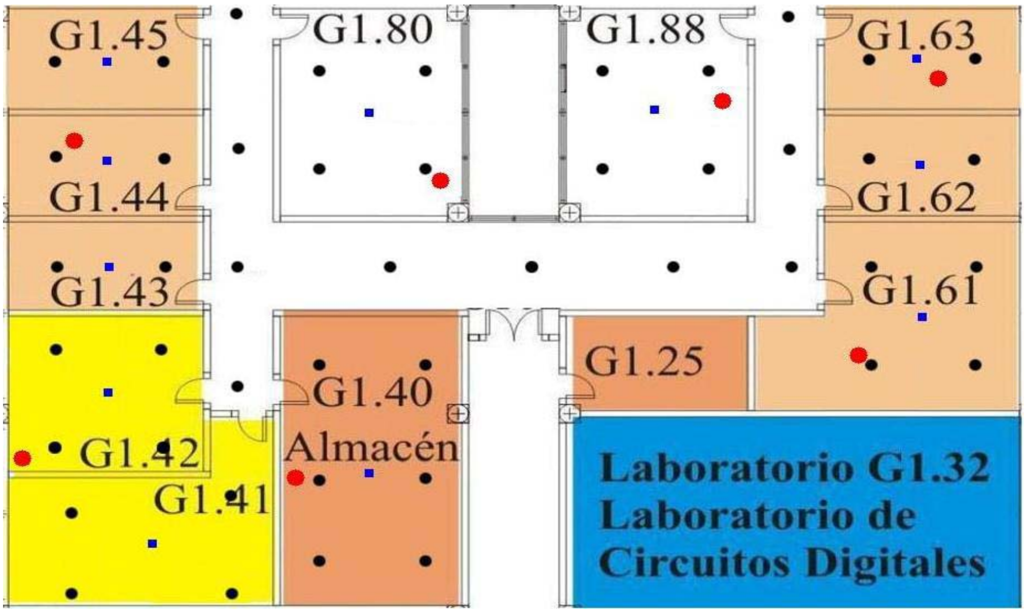

Figure 3. Fixed Motes Location and signature points saved in the database

To send periodically beacon2 messages, each fixed motes search in its neighbour table to find out if the mobile mote is in its range of coverage, if so, the beacon2 is sent to C with the information required as explained in section 3. The beacon 2 message is a Zigbee Application message provided by the APS (APplication Service). As neighbour table is only in FFD, fixed motes have also to be R.

We deployed the BitCloud solution over half floor of our Department Area, measuring roughly 225 m2. To cover all this area we required 7 fixed motes strategically placed, shown as red points in Fig. 3. An off-line phase was required for each algorithm to fill in the signature database, once it was full, the system was ready to be tested.

Fig. 4 shows the PC interface to presents the mobile mote position, four in this case. It also shows mobile mote sensors information.

A. BitCloud Test

We have tested the system to check if it can determine the location of the mobile mote using both algorithms. It must be noticed that we pretend to locate in rooms, hallways or courtyard, but with room-level accuracy. We considered it doesn´t matter exactly where it is inside de room. This has been this way, because the kind of applications for whom our indoor position solution is going to be used, doesn´t require more precision.

B. Simple Algorithm

For single fingerprint algorithm, 11 points were collected and saved in the fingerprint database, one point of each room, shown as blue points in Fig. 3. Through this algorithm is pretended to locate only in rooms, not halls or courtyards. In this case, we placed the mobile mote on each room (11 times) and nearly all of the position measurements performed good results. 82% of success was obtained. Only two rooms were unallocated. Fig. 5 shows the results. Empirical tests have demonstrated that 11 points were enough to conclude, because tests made on the same room, had behaved the same way.

C. Centroid Algorithm

However, centroid fingerprint algorithm is also intended to locate in corridors too, even in the entrance of each room. In this case, 46 points were used to fill the fingerprint database, distributed by rooms and hallways. See Fig. 3 black points

We decided, also based on empirical measurements, to test 30 points, to check how the positioning centroid algorithm worked. It was determined that the algorithm presented the right position in 23 points but in 7 points, it made a bad position determination. Fig. 5 shows those points. Therefore our precision was about 77%.

Although results were as expected, as shown in Motetrack this solution has two drawbacks:

The mobile node has to be FFD because it has to use the neighbour table to get the RSSI so the power consumption is very high and it causes consumption problems since mobile node is battery powered.

The periodicity of beacon1 messages can´t be controlled, and we can't find out the age of the RSSI values as it is a Zigbee Network parameter not accessible by BitCloud because it is an Application Level Stack.

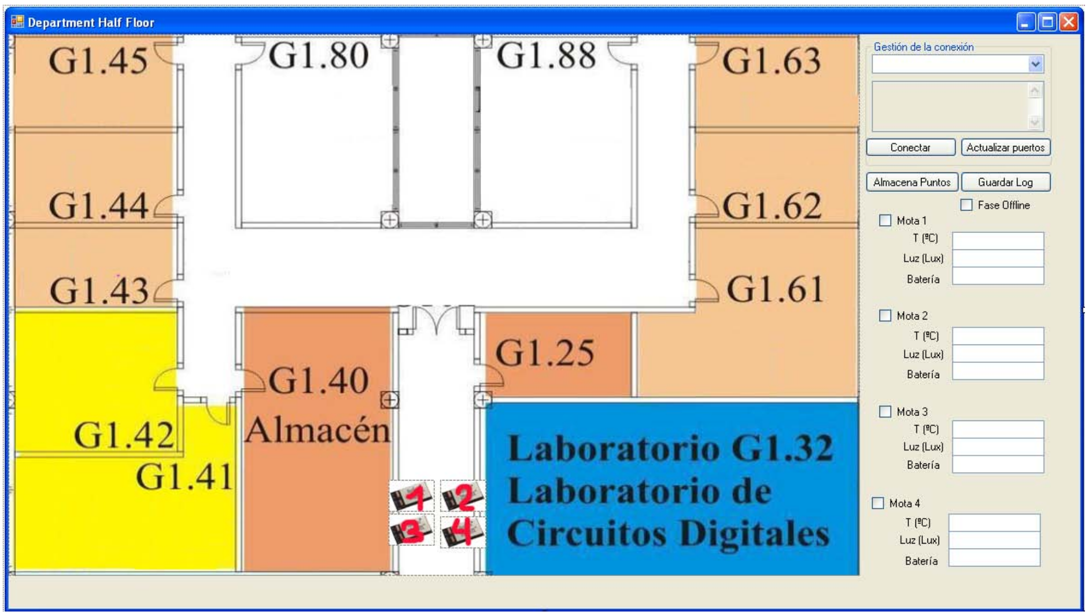

Figure 4. Position System Interface

Further work is done to optimize the system WSN stack, in order to fix the found drawbacks and to try to get more accuracy.

V. OPENMAC IMPLEMENTACION

OpenMAC is an open source implementation of IEEE802.15.4 Media Access Control (MAC) layer. It's an embedded software that provides basic networking functionality, only star and peer-to-peer topologies. Because of it, only Coordinator (C) and End Devices (ED) examples are implemented. To create all the PAN devices, the MAC services implemented in OpenMAC for doing so has been used.

It has some advantage over using BitCloud Stack:

1. Enables users, who do not require full functionality of BitCloud Stack, to develop custom WSN applications.

2. Enabled advanced users to modify OpenMAC internals to suit specific application needs.

3. Jump start application development on top of MAC with thoroughly documented sample applications.

4. Provide a convenient C API to developers not familiar with TinyOS or nesC programming language (technologies at the core of OpenMAC).

5. Provide a reference design to be ported to analogous hardware platforms.

To deploy the same indoor solution as in BitCloud Stack, a PAN like the one shown in Fig. 1 has been created, where all types of ZigBee devices, C, R and ED

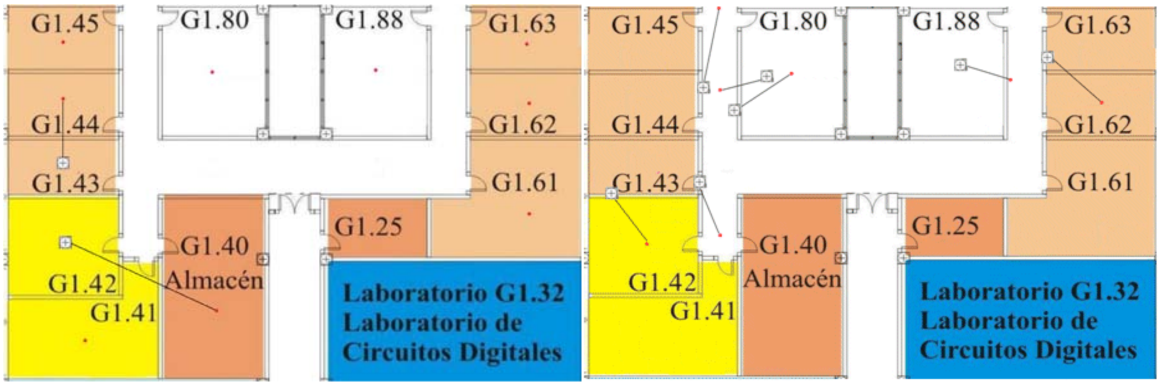

Figure 5. Fingerprint and Centroid Tests Points and Errors using BitCloud Stack

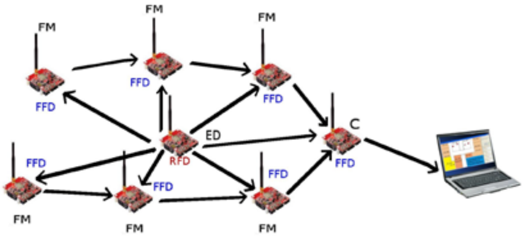

Figure 6. OpenMAC System Overview

are created and all of them implement the corresponding functionally. So, C creates the PAN and R and ED connect to the PAN via C or other R that is already in the PAN. See Fig. 6.

All data flow is towards C, so Rs are in charge of forwarding packets when other Rs or EDs requires it, this happens when the first ones are closer to C than the second ones.

The ED is in charge to send beacon1 messages. This message is broadcasted, so all its neighbours are able to calculate the RSSI value of the received message and send this information directly or indirectly to C. When an ED message is received, R sends beacon2 message to his father. This message is unicast, so a R that receives one, has to forward it if the source R is one of its child.

The OpenMAC PAN works correctly and is set to deploy it over the half flour of our Department too.

The PC system is nearly the same used with BitCloud stack, only some changes were necessary to collect data from beacon2 messages. Obviously OpenMAC beacon2 messages structure is different from BitCloud one.

A. *OpenMAC tests*

Simple Algorithm

For single fingerprint algorithm, 11 points were collected too, using the new protocol stack, one point of each room, as equal to BitCloud test, and the same results of BitCloud were obtained, two errors over 11 measures, one per room, 82% of success. Fig. 7 shows the test

Centroid Algorithm

Again, the same 46 points we used in BitCloud tests, were saved into the OpenMAC Centroid fingerprint database.

The same 30 points tested in BitCloud has been used to check how the positioning centroid algorithm works using the OpenMAC stack. It is determined that the algorithm gives the right position in 26 points, and in 4 points it presents a wrong position determination. Making numbers again our precision is about 87%. Better results than the first BitCloud tests. The test can be seen in Fig. 7

However, the two main drawbacks of BitCloud stack are solved this way. In OpenMAC, mobile motes are RFD (Fig. 6), so they are slept all the time and are only woken up when they have to send broadcast beacon1 message, the periodicity of the messages are controlled and updated every time, so the age of the RSSI value is now controlled. Using OpenMAC, it can be considered that the accuracy of the obtained off-line fingerprint database is better than the BitCloud. RSSI values are new on each time and the average is made using a realistic sample. In BitCloud, as said before, the age of the RSSI measure is unknown and if it isn't updated, the same value is always given. The fingerprint database made using this sample must be worse than the one made by OpenMAC.

VI. CONCLUSION

We have presented an indoor position system based on WSN using the RSSI. BitCloud stack has been used to implement the network functionality.

Some drawbacks have been found by using BitCloud stack and were considered they could affect positioning accuracy so another implementation has been done using another stack, OpenMAC. A PAN infrastructure for indoor position has been developed and used in a lab environment.

By checking the OpenMAC implementation improvements and compare it to the previous one, some off-line phase has been required to fill in the different signatures database and some tests have been done using

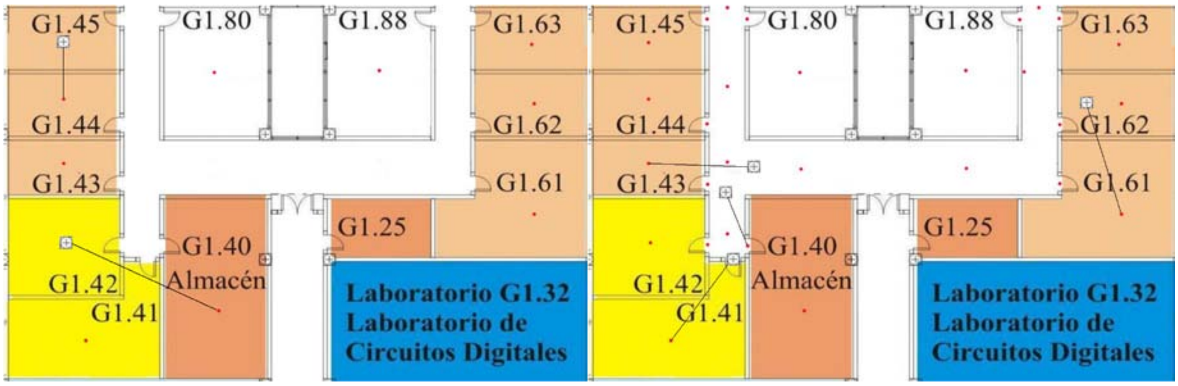

Figure 7. OpenMAC Fingerprint an Centroid Tests Points and Errors

different positioning algorithms, Simple (if only room positioning is required) and Centroid (if more complex positioning estimation is needed), both getting good results. Accuracy is increased 10% using Centroid algorithm. Using Simple algorithm we got the same results.

ACKNOWLEDGMENT

This work has been carried out within the framework of two research programs: (P08-TIC-3631) – Multimodal Wireless interface (IMI) funded by the Regional Government of Andalusia and Efficient and Health Intelligent Technologies Oriented to Health and comfort in Interior Environments (TECNO-CAI) approved project at the fifth call of CENIT program by the Innovation Science Ministry of Spain (CDTI and Ingenio 2010 Program).

REFERENCES

[1] Hristova, A.; Bernardos, A.M.; Casar, J.R.2008.Context-aware services for ambient assisted living: A case-study. First International Symposium on Applied Sciences on Biomedical and Communication Technologies, 2008. ISABEL '08.

[2] Figueiredo, C.P.; Gama, O.S.; Pereira, C.M.; Mendes, P.M.; Silva, S.; Domingues, L.; Hoffmann, K.-P. 2010 Autonomy Suitability of Wireless Modules for Ambient Assisted Living Applications: WiFi, ZigBee, and Proprietary Devices.*Sensor Technologies and Applications (SENSORCOMM*).

[3] Hong Sun; De Florio, V.; Ning Gui; Blondia, C., Towards. 2008. Building Virtual Community for Ambient Assisted Living. 16th Euromicro Conference on Parallel, Distributed and Network-Based Processing.

[4] Sun, H.; De Florio, V.; Gui, N.; Blondia, C., 2009. PRomises and Challenges of Ambient Assisted Living Systems. Information Technology: New Generations, 2009. ITNG '09.Sixth International Conference on. Page(s): 1201 - 1207.

[5] Martin, H.; Bernardos, A.M.; Bergesio, L.; Tarrio, P.,2009. Analysis of key aspects to manage wireless sensor networks in ambient assisted living environments. Applied Sciences in Biomedical and Communication Technologies, 2009. ISABEL 2009. 2nd International Symposium. Page(s): 1 - 8.

[6] Dietrich, D.; Bruckner, D.; Zucker, G.; Palensky, P. 2010.Communication and Computation in Buildings: A Short Introduction and Overview. IEEE Transaction on Industrial ElectronicVolume: 57 , Issue:11.

[7] Chen, Po-Wei; Ou, Kuang-Shun; Chen, Kuo-Shen. 2010.IR indoor localization and wireless transmission for motion control in smart building applications based on Wiimote technology.SICE Annual Conference 2010.

[8] Han Chen; Chou, P.; Duri, S.; Hui Lei; Reason, J. 2009.The Design and Implementation of a Smart Building Control . Page(s): 255 - 262.

[9] Snoonian, D , Smart buildings.2003. Spectrum, IEEE Volume: 40 , Issue: 8 Digital Object Identifier: 10.1109/MSPEC. Page(s): 18 - 23.

[10] Bahl P., Padmanabhan VN. 2000. RADAR: an in-building RF-based user location and tracking system. In: INFOCOM, pp 775–784.

[11] Konrad L. and Matt W. 2005. MoteTrack: A Robust, Decentralized Approach to RF-Based Location Tracking. In Proceedings of the International Workshop on Location and Context-Awareness (LoCA 2005) at Pervasive May 2005.

[12] Konrad L., Matt W. 2006. MoteTrack: A Robust, Decentralized Approach to RF-Based Location Tracking. To Appear in Springer Personal and Ubiquitous Computing, Special Issue on Location and Context-Awareness. . ISSN: 1617-4909 (Print) 1617-4917.

[13] Medina A. V., Gómez I., Romera M., Gómez J.A. and Dorronzoro E. 2011.Indoor Position System based on BitCloud Stack for Ambient Living and Smart Buildings. In 3rd International ICST Conference on IT Revolutions. Córdoba, Spain.

[14] ZigBee, 2009 RF4CE Specification. Version 1.00. ZigBee Document 094945r00ZB, March 17th, 2009.

[15] Medina A.V., Gómez J.A. Rivera O. Dorronzoro E. and Merino M. 2011. Fingerprint Indoor Position System based on OpenMAC. In International Conference on Wireless Information Networks and Systems, Sevilla Spain

Jose A. Gómez was born in Seville, Spain on 14 February 1983. He received the computer engineer degree from the Seville University in 2009 and network and computing engineering master in Computer Engineering and Network from the University of Seville in 2011

He has been worked as a Teacher in higher grade formative cycles. Currently he is a PhD student in Department of Electronic Technology where also works as a teacher. His research interest includes WSN and Indoor Localization Systems. He has published some technological papers.

A.Verónica Medina (IEEE Member'08) received her degree in computer engineering in 1993 and her doctorate (Ph. D.) in computer engineering in 1999 from the University of Seville (Spain). Since 1990, she has been working on the formal description techniques and their application to the development of communication protocols. In 1992, she was awarded a prize by the Andalusia government for her study "Application of Computer Sciences to the Control of Electric Power Networks." Presently, she is a Lecturer of electronic engineering at the University of Seville and a researcher in TAIS research group (Technologies for Care, Inclusion and Health, http://matrix.dte.us.es/grupotais/).

Octavio Rivera received the Ph.D. degree from the University of Seville, Seville, Spain, 2010.

His research interests include Assistive technology, Augmentative and Alternative Communication, HCI, Serious games and rehabilitation engineering.

He is currently a Lecturer and Researcher at University of Seville, Seville, Spain.

He has been a member of a working group in the communication field since 2001. He worked in some communication projects involving power-line communications systems, digital-subscriber-line technologies, indoor location systems and telecontrol protocols.

Enrique Dorronzoro received his master degree of Computer Science Engineer at University of Sevilla, Spain, in 2007.

He has worked as Instructor at the Cisco Networking academy program. He was on 2010 in a 6 months research visit at Norut in Tromsø, Norway. Nowadays he works as researcher at Electronic & Technology Department of the same University where he has worked in several research paper. His interest include wireless networks sensors systems applied to telemedicine.

Sergio Martín received his degree in Computer Engineering in 1994. Since 1990, he has been working on the Formal Description Techniques, their application to the development and also in the application of Expert System to the Industry. Now his research is focusing on adding expert rules to the management objects in TMN (Telecommunication Management Network). Presently he is an Associate Professor of Computer Engineering at the University of Seville.