

# Impact of Environmental Factors for Wireless Network Based in Indoor Positioning Algorithms

Abdoua ISSA, Philippe CANALDA, François SPIES  
Univ. Bourgogne Franche-comté FEMTO-ST Institute/CNRS 6174  
1, Cours Louis Leprince Ringuet 25200 Montbéliard France cedex

**Abstract**—The purpose of this paper is to show the impact of physical parameters (antenna gain, transmit and receive frequency, transmit and receive power, etc.) and atmospheric conditions (temperature, test time, humidity,) in the wireless network based indoor positioning algorithms. This study focuses on the accuracy and quality of the calculated position relative to the actual position. We first present the OWLPS (Owl Positioning System) which is a positioning system based on IEEE 802.11 signal strength, designed, developed and released by the FEMTO-ST/DISC/OMNI team of University of Franche-comté at Montbéliard. The data produced by the aggregator server are used to the positioning server in order to appreciate our approach. Finally, we will understand the results produced by our method with those produced by other algorithms.

**KEYWORDS:** indoor navigation, location based services, mobile sensing, indoor positioning.

## I. Motivation

The team of university of Franche-Comte were developing a plate-form Wi-Fi for Indoor positioning ,and the two principle goals which was done :

- 1) detect the position using Access points Wi-Fi.
- 2) Integration of the functionality of auto-calibration automatically ,fast and timely function which we can use for best and more precise of position .

the second functionality was made using mechanism of Token-Ring. The second functionality that is activated in a token-ring mechanism, wireless access points appear to participate in the evaluation explicit or implicate of mobile WI-FI. The initial phase self-calibration, fingerprinting using the terminology of history, is so full, incrementally, an update of a referential powers transmitted positioning were performed, mainly bases on the n- lateration, so that taking into account data such as: propagation model, the positioning of the 3D; infrastructure wi-fi devices, the topology of the building and its displacement plan, currently these mechanisms allow an assessment with a individual margin of error too large. We wish to investigate on the one hand the orientation of the mobile terminal to identify, as well as the position of the carrier humanoid, then the other hand the position of a new couple consideration the perturbation induced by the presence of human and terminal mobile.[1][2][3][4][5]

## II. Analysis of requirements

The application has two main stages: 1. First stage: In this stage we test the power and quality of the mobile single, after

that we start to study the aspects of the mobile phone (such as place of occurrence, orientation, rotation etc...). By using the auto calibration request, we test the conditions of context of environment to find the best conditions to keep the power of the mobile signal (e.g. number and positions of persons). This data is going to be stored into a database. Then we are going to apply Data mining approaches (e.g. Decision trees) on the database to extract the information (features) and use the result to select the most appropriate position of the access points in the building. In this process we use Database Sqlite to record the information in the smart phone (server in distance for handling the data using data mining algorithms).

2. Second stage: We focus in the stage on the practical issues. We depend on the application's different functions which could be used for sending the requests to the server of OWLPS system. This process could be done through two ways of connection (UDP request or Http request). Then the person position could be shown in two different ways (Google Map, 3D Model) using Sketchup or OPENGL Library of android phone. Then we synchronize the smart phone database using the server in distance, and store the information using web service to find the nearest access points to be used as crossing points in order to deliver the request to the server.

## III. CONTRIBUTION

The principle of a specification is to define the expected elements and constraints. This part will define the overall context of the project, then the general architecture of owlps system functionalists of mobile phone,Test and results of tests, and Finally, the technical constraints.

### A. Background

This application mobile (smart phone) can be used to make entry into the server using webservice to be able to send the request users and applications get the position. The context of the scenarios:

- 1) Make registration of users in the database (Microsoft SQL Server Denali CTP3) in the main server (IIS) using webservice and registration will be performing with LINQ.
- 2) Send to system standard request to detect the user' position with different values for (number of packets

and delay between every packet,number of request,delay between every request).

- 3) send auto-calibration request ( in automatic and Manuel way) for calibrated the grid of fingerprints) with different values for (number of packets and delay between every packet,number of request,delay between every request).
- 4) Send the result to the server side of users.
- 5) Synchronize the information between mobile phone and server a distance(update information )
- 6) build database sqlite for mobile android for register the information of mobile phone and user(last position and values of sensors)
- 7) Server in language c for HTTP requests
- 8) Client in language java script and java for HTTP requests(Standard ,auto-calibration,Http request for image typology,Http request for loading file ).
- 9) interface android for getting the position and rotation and other information of mobile android and save the information in file XML.[6][7]

1) *Architecture General:* This part will describe the general architecture of the project, with different General steps and messages needed and expected functionality. Figure (1) represents the general diagram of the application architecture that will sent by mobile phone to the Access Points and Access Points will send the request to server aggregation after the addition of other information Access point mac address and the time the request arrived at Access Points and after the aggregation server will assemble the data and return data to the main servers that request and the receiver detect the users's position and the result after returning to the server.In our architecture we have server a distance for make synchronization between mobile phone and this server for update the information of environment and the information of mobile phone the technique used to add information in the packet sent is called the interface radiotap.[8][9]

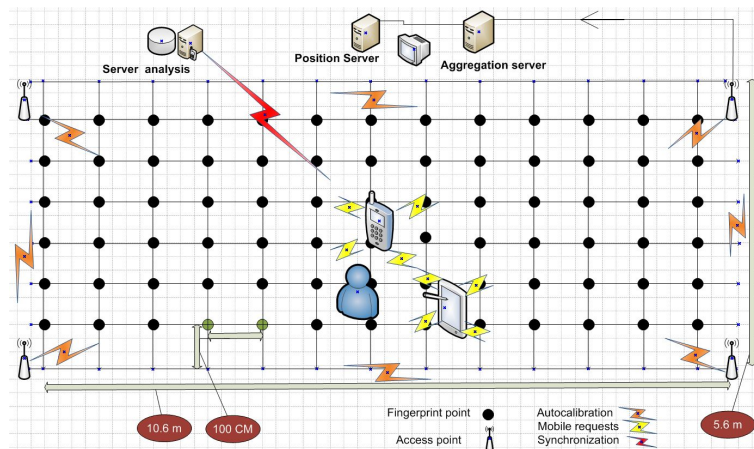


Figure 1. Conception general

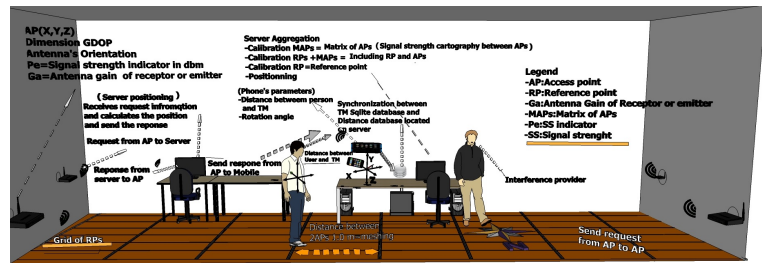


Figure 2. Conception modeling

#### IV. Technical constraints

This section will describe the various technical constraints of the project on the design methodology.

##### Design Method

This project realized by the Extreme Programming methodology (XP) is a method Agile project management information tailored to smaller teams with changing requirements. She pushes to the limit of simple principles. Agile method that is to say (Methods Agile practices are groups that may apply to various types of projects, but limited instead currently in development projects in computer science (software design). The Agile methods are intended to more pragmatic than traditional methods. They imply maximum of the applicant (customer) and allow a greater responsiveness to his requests. They designed to satisfy real customer needs and not under a development contract).[10][11][12]

##### Libraries and plugins

- 1) K2SOAP library:

KSOAP is a "lightweight and efficient SOAP library for the Android platform." SOAP is a protocol used for building web services, and which defines a format used for the exchange of messages that uses XML as base.KSOAP is a SOAP API based on kXML,where kXML is a lightweight Java-based XML parser designed to run on limited, embedded systems such as personal mobile devices. It is a pull parser which means it reads a little bit of the document at once So, KSOAP was specially was designed or developed to deal with small embedded device like mobile devices. While kSOAP doesn't support WSDL, it does make calling a service relatively painless. Only two objects are required: the SOAPObject and Http-Transport. The power of SOAP is its flexibility, which comes from SOAP's foundation in XML, and its ubiquity, which comes from the use of HTTP as a primary transport mechanism. Any modern language that works with the Internet is likely to support both XML and HTTP, and Java is no exception. There are, in fact, many good toolkits for using SOAP with Java and Apache Axis is one of the most popular. Choices abound for the developer working with J2SE or J2EE.

- 2) Green Droid library:  
Most Android users may have recognized more and more common UI elements in Android applications these days. Widgets like ActionBar and QuickActions are a great approach to standardize the way Android applications are used for easy navigation. Google posted some design proposals and took the Twitter for Android app as a great example for Android design patterns.
- 3) AChartEngine:  
AChartEngine is a charting library for Android applications. It currently supports the following chart types: line chart area chart scatter chart time chart bar chart pie chart All the above supported chart types can contain multiple series, can be displayed with the X axis horizontally (default) or vertically and support many other custom features. The charts can be built as a view that can be added to a view group or as an intent, such as it can be used to start an activity. The library is currently at the 0.3.0 release. The 0.3.0 was the first one with features based on the feedback from the library users. New chart types will be added in the following releases of the library
- 4) ADT Plugin:  
Installing the Eclipse Plugin Android Development Tools (ADT) is a plugin for the Eclipse IDE that is designed to give you a powerful, integrated environment in which to build Android applications. ADT extends the capabilities of Eclipse to let you quickly set up new Android projects, create an application UI, add packages based on the Android Framework API, debug your applications using the Android SDK tools, and even export signed (or unsigned) .apk files in order to distribute your application.
- 5) Cell Object plugin:  
The tools are open source. If the tools are useful, please consider to donate! Your donations keep me motivated to develop the tools further. Used for manage Android sqlite database from eclipse IDE.

### **Programming languages:**

- 1) Linq: Language Integrated Query (LINQ, pronounced "link") is a Microsoft .NET Framework component that adds native data querying capabilities to .NET languages, although ports exist for Java[1], PHP and JavaScript. LINQ defines a set of method names (called standard query operators, or standard sequence operators), along with translation rules from so-called query expressions to expressions using these method names, lambda expressions and anonymous types. These can, for example, be used to project and filter data into arrays, enumerable classes, XML (LINQ to XML), relational databases, and third party data sources. Used for Server a distance for Web service code in VS plate form.
- 2) CSharp:  
is a multi-paradigm programming language encompass-

ing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines. It was developed by Microsoft. Used for Server a distance for Web service code in VS plate form.

- 3) Java:  
is a programming language originally developed by James Gosling at Sun Micro systems (which has since merged into Oracle Corporation) and released in 1995 as a core component of Sun Micro systems' Java platform. The language derives much of its syntax from C# and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Used for android application in eclipse plate form.
- 4) c:  
is a general-purpose computer programming language developed between 1969 and 1973 by Dennis Ritchie at the Bell Telephone Laboratories for use with the Unix operating system. Used for Server HTTP .
- 5) Java script:  
is a prototype-based scripting language that is dynamic, weakly typed and has first-class functions. It is a multi-paradigm language, supporting object-oriented,[5] imperative, and functional[1][6] programming styles. JavaScript was formalized in the ECMAScript language standard and is primarily used in the form of client-side JavaScript, implemented as part of a Web browser in order to give enhanced user interfaces and dynamic websites. This enables programmatic access to computational objects within a host environment. Used for android request Http in eclipse plate form..
- 6) jQuery:  
jQuery can be used to develop highly interactive web applications. In this jQuery tutorials series we will show you how to use jQuery to develop nice ajax based applications. The jQuery is cross-browser JavaScript library and it helps the programmer's to develop cross-browser application easily. There are many available for the jQuery for different functionality. You can write your own plugin and then use it with jQuery. Used for android request Http in eclipse plate form.
- 7) ASP.net web Service:  
ASP.NET is not just the next version of ASP; it is the next era of web development. ASP.NET allows you to use a full featured programming language such as CSharp (pronounced C#) or VB.NET to build web applications easily. One great feature of ASP.NET is Web Services. ASP.NET Web Services support clients using HTTP-POST, HTTP-GET and SOAP protocols to invoke methods exposed, depends on your specific

requirement you choose one method over the others. The main difference between HTTP-GET or HTTP-POST and SOAP is the data types supported by SOAP is much richer because SOAP used XSD schema to represent complex data types. Web services mean that you can literally have several pieces of your application on different servers all around the world, and the entire application will work perfectly and seamlessly. Web services can even work with normal .NET Windows applications.

Used for Web service(server side)

#### Plate forms:

1) Eclipse:  
is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. It is written mostly in Java. It can be used to develop applications in Java and, by means of various plug-ins, other programming languages including Ada, C, C++, COBOL, Haskell, Perl, PHP, Python, R, Ruby (including Ruby on Rails framework), Scala, Clojure, Groovy and Scheme. It can also be used to develop packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others. The initial codebase originated from VisualAge.

2) Microsoft Visual Studio:  
Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop console and graphical user interface applications along with Windows Forms applications, web sites, web applications, and web services in both native code together with managed code for all platforms supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight.

#### 3) Database:

1) Microsoft SQL Server:  
is a management system database (abbreviated as DBMS or RDBMS for "management system relational database") developed and marketed by Microsoft.

2) Sqlite:  
is a relational database management system contained in a small ( 275 kB) . In contrast to other database management systems, SQLite is not a separate process that is accessed from the client application, but an integral part of it. Android default Database engine is Lite. SQLite is a lightweight transactional database engine that occupies a small amount of disk storage and memory, so it's a perfect choice for creating databases on many mobile operating systems such as Android, iOS.

Things to consider when dealing with SQLite:

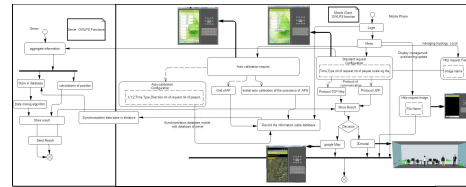


Figure 3. Activity diagram

- a) Data type integrity is not maintained in SQLite, you can put a value of a certain data type in a column of another datatype (put string in an integer and vice versa).
- b) Referential integrity is not maintained in SQLite, there is no FOREIGN KEY constraints or JOIN statements.
- c) SQLite Full Unicode support is optional and not installed by default.

### V. The Implementation

1. on our stage we tried to make tools for test the OWLPS system and this tool was an application for Smart phone(Android)using different type of request and different conditions for using the system and testing and the system in different ways and make request for position explicit and implicit: first we developed database using Sqlite Database and in this database we have these tables :

- 1) for the typology of the building
  - a) Building information table(Property, Address, Date, Description)
  - b) Department information table(Property, Address, Date, Description ,Building ID)
  - c) Cluster information Table(Property, Address, Date, Description ,Cluster ID)
  - d) Points of Building (X, Y, Z, Building ID, Reference point)
  - e) Points of Department(X, Y, Z, Department ID, Reference point)
  - f) Points of Cluster(X,Y,Z, Cluster ID, Reference point)

#### 2) For request types:

- a) request Standard:  
(number of packets, total number of packets , number of request, number total of request, mac address, IP address, Port)
- b) Auto Calibration Request: (number of packets, total number of packets, number of request, number total of request, mac address, IP address, Port, X, Y , Z(Coordinate of terminal Access point), Direction)
- c) Terminal mobile table: (Information for all sensors in android which we can use for analysis the effect of position of android ,its rotation ,temperature of room, humidity of room, Distance from the person and mobile phone, Acceleration of person holding mobile phone)

- d) User Information (for registration operation(Mac address, IP address, First Name , Last name, Password)
- e) Access points information which we can use them for specific calculation of the system: (Static, Direction, Situation, Mac Address, IP address , X ,Y , Z (Coordinate of terminal mobile of Access points) ,Gain emission, Gain Reception, Puissance emission , port )
- f) Access Points next which contain the information of the nearest access point :  
(ID Access Point , Mac Address1, IP address1 , X1 ,Y1 , Z1 ,Gain emission1, Gain Reception1, Puissance emission1 , port , Gain emission 1 , Gain Reception1, Puissance emission 1 )
- g) Terminal mobile next which contain the information of the nearest TM : (ID Access Point , Mac Address1, IP address1 , X1 ,Y1 , Z1(Coordinate of terminal mobile))
- h) Historic of user in this table we register the last position coordinate of user in table: (X, Y , Z, User ID)
- i) Auto Calibration Table: (Mac Address1, IP address1 ,Gain emission1, Gain Reception1, Puissance emission1, Mac Address2, IP address2 ,Gain emission2, Gain Reception2, Puissance emission2 ,RSS)

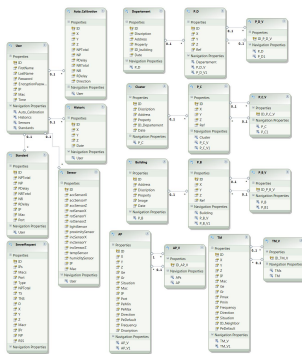


Figure 4. Entity Designer Diagram

- 3) Topology Functionality: For achieve these functionality we use Google map APi(Google Maps (formerly Google Local) is a web mapping service application and technology provided by Google, that powers many map-based services, including the Google Maps website, Google Ride Finder, Google Transit,[1] and maps embedded on third-party websites via the Google Maps API It offers street maps, a route planner for traveling by foot, car, bike (beta), or public transport and an urban business locator for numerous countries around the world. Google Maps satellite images are not updated in real time; they are several months or years old. Google Maps uses a close variant of the Mercator projection, so it cannot show areas around the poles. A related product is Google Earth, a stand-alone program which offers more globe-viewing features, including showing polar

areas.) for our application in which we can detect the position and show the position using marker after we convert the point from relative to absolute points .

- 4) Synchronization with server a distance functionality :
  1. The main goal of these function is to offer option for register the data in server a distance for make analysis and for tracking purpose
  2. Update information always
  3. tracking purpose for update the information of Changes which take place environment.
 Function of convert Relative point(with relation to reference point) to absolute points for show these point in Google Map.
- 5) Standard Request: we achieve this function for getting the position from the server
- 6) Auto Calibration Request: we achieve this function for getting calibrate the system of owlps in manually or dynamically.
- 7) Image request: we achieve this function for getting Image of typology
- 8) Image request: we achieve this function for getting File CSV of system OWLPS which contains the information of system (auto calibration , rss information, AP sending MAC address and AP receiving Mac Address ,Time Stamp in NS and S, Coordinate and direction,Client MAc address )
- 9) Sensor modeling functionality: we achieve this function for getting the different rotation and position of mobile terminal and modeling it using server and interface android and save file xml for these values
- 10) Server HTTP in C: for sending and receiving http request.
- 11) HTTP client.
- 12) calibration request HTTP.

## VI. Android Sensors

: In our functionality we used sensors of android to get information from the context of the environment like temperature sensor and get information of the different mode and position of android like rotation,vertical ,horizontal ,distance from the object who used it and so on . different types of sensors android offer:

- 1) Accelerometer sensor :  
A three-axis accelerometer sensor that returns the current acceleration along three axes in m/s<sup>2</sup>.
- 2) Gyroscope Sensor :  
A gyroscopic sensor that returns the current device orientation on three axes in degrees.
- 3) Light sensor :  
An ambient light sensor that returns a single value describing the ambient illumination in lux. A light sensor is commonly used to dynamically control the screen brightness.
- 4) Magnetic field sensor :  
A magnetic field sensor that finds the current magnetic field in microteslas along three axes.



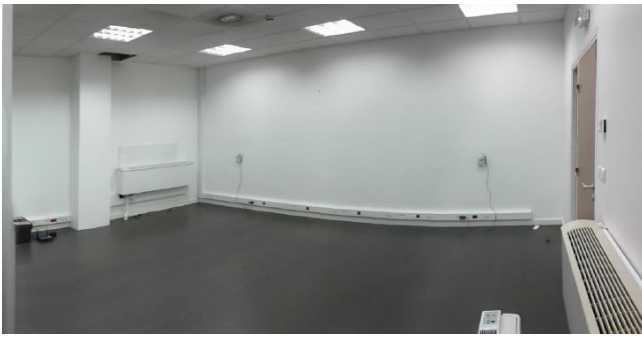


Figure 5. Room plan in numerica building

5) Orientation sensor:

An orientation sensor that returns the device orientation on three axes in degrees. The orientation sensor is explored in greater detail later in this chapter.

6) Proximity sensor:

A proximity sensor that indicates the distance between the device and the target object in meters. How a target object is selected, and the distances supported, will depend on the hardware implementation of the proximity detector

**VII. Test and experiments**

Our proposed method can work well with most of indoor location systems. However, it also has its limitation. As described before, our proposed method first determines of several scenarios and study the and see the results of different scenarios for that we make the tests in Big room in numerica building in Montbeliard the size of this room is 10.60 m long width 5.80(10.60X65.80) and this figure explain plan of this this experimental environment. We see in the room the grid of fingerprinting and the distributions of APs and the coordinates of each AP relative in three axis x-axis , y-axis and z-axis, whose coordinates are shown in table below. We have four Access points and their position . in this table

Table I  
TABLE OF GLOSSARIES

AP	X	Y	Z
Ap1 Position	4.80000	10.60000	1.00000
Ap2 Position	1.00000	10.60000	1.00000
Ap3 Position	4.80000	0.00000	1.00000
Ap4(Ref) Position	0.00000	1.00000	0.00000

we suppose Ap4 is the reference point. There are equipment and tools we have used for tests :

- 1) Nexus Samsung: Nexus S is the next generation of Nexus devices co-developed by Google and Samsung. It is the first smartphone to use the Android 2.3 "Gingerbread" OS. The most significant reason of choosing this device is the large number of built-in sensors into it. At the project start Nexus S was the only Android

smartphone available on the market that included and supported a gyroscope.

- 2) Personal computer EEE-PC 1001-PX serving as a client PC with a core Linux 3.0.0-12 and carte Wifi Atheros AR2427.
- 3) Access point Fonera : - Five Wi-Fi APs Fonera 2.0 running the embedded Linux distribution OpenWrt and equipped with an USB port, an Ethernet WAN port, an Ethernet LAN port and with a 1.5 dBi antenna. The Wi-Fi chipset are Atheros AR2315 running at 180 MHz, 32 MB RAM, 8 MB storage.
- 4) The aggregation and positioning software modules are both installed on an EEE-PC 4G running Debian GNU/Linux and with an Atheros AR5001 Wi-Fi card.

**A. Positioning Algorithms Implemented**

- Nearest-neighbour in Signal Strength (NSS), based on RADAR [3], which is a simple cartography-based algorithm.
- Trilateration using the propagation formula proposed by Interlink Networks in [4].
- Trilateration using the FBCM [2, 5] (Friis-Based Calibrated Model ), which adapts the propagation formula to better match the deployment area's characteristics, thanks to a minimal calibration.
- Basic FRBHM [6, 5, 7] (FBCM and Reference-Based Hybrid Model), that is a combination of the NSS and the FBCM which allows to adapt dynamically the propagation formula to the characteristics of the room where the mobile terminal is supposed to be.

**B. Commands of system**

With these command we can use OWLPS system for for more information see reapot technique.

- 1) change frequency of auto-calibration
  - 2) change frequency of listeners
  - 3) lunch server
  - 4) lunch client
  - 5) login to all access points
- 1) Client for change the parameters of the client (Fonera ).
  - 2) Aggregator commands for lunching aggregator.
  - 3) Access points commands

**VIII. SCENARIOS OF TESTS**

- This experience is the same of first but the client At waist level and no human operator in the room(five points in the room we have to mesure the ss four points in each corner of the room and the another point in the center of room )we send requests and the delay between each request is 500ms and for each request there are 20 packets and the delay between each packet is 10ms.
- This experience is the same of first but the client on the floor no human operator in the room (five points in the room we have to measure the ss four points in each corner of the room and the another point in the center of room )we send requests and the delay between each request is 500ms and for each request there are 20 packets and the delay between each packet is 10ms.

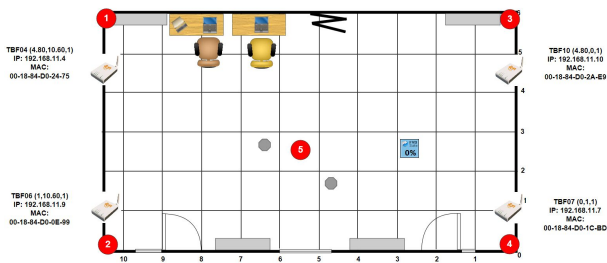


Figure 6. Scenario 1

- This experience interest in two measurement points (one in the lower left corner of the room and the other center). Measures should be caught in two directions: one facing the opposite wall and the other located 45 degrees from the previous position. The customer will be held by a human operator and it will be tested in three different positions: Horizontal, Vertical. The program will be run one minute for each position and orientation. we send request for one minute and the delay between each request is 500ms and for each request we send 20 packet and the delay between each packet is 10ms.

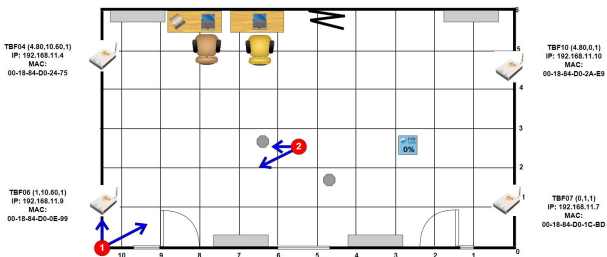


Figure 7. Scenario 3

- this experience is the same of first two experience but the human is moving from every point to other and the program is not stopping.

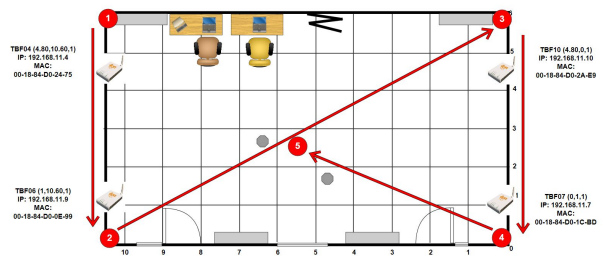


Figure 8. Scenario 4

- Launch auto-calibration on the aggregation server for 10s and then restart the server without self-calibration and make measurements of the central point a minute as packet transmission frequency: 1, 5, 10, 20, 40, 500 ms.

- Using new measurement points (see diagram 9). For each point of measurements to test three different heights: floor, hip, ear. A partition will be drawn to the center of the room and no human operator in the room. The program is started one minute on each measurement point and position.

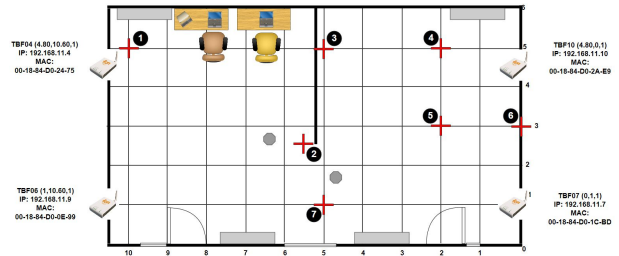


Figure 9. Scenario 6

- Same experience as the previous (6) VIII and see diagram 9 but by adding a human operator located 50 cm south of the client and we will consider only one level, that of the hip.
- Same as Experiment 7 VIII except that it adds a human operator to a fixed point of the room.
- in this experiment we Start the server aggregation with auto-calibration for ten seconds and then restart it without self-calibration and take measures at the central point for one minute with different packet sizes: 64, 128, 256, 512, 1024, 1450.
- Repeat the scenario 1 see ?? and diagram ?? by changing this time the frequency of calibration as following values: 5, 10, 15, 20, 25 ms.
- During this we repeat scenario 4 see ?? and diagram 8 with a frequency-calibration different between each Access Point. values to be tested: 100ms (-n10 -t8), 250ms (-n16-t14), 500ms (-n20-t24), 1000ms (-n39-t25).
- Take the measurement points of the scenarios numbers (3 ??, 4 ??, 5 VIII and (see diagrams 3, 7, 4, 8, 5) and make measurements with an orientation antennas of different AP. Firstly antenna disposed horizontally towards the opposite wall. Second antenna horizontal but always arranged so that each antenna points toward another AP. see diagram see diagram

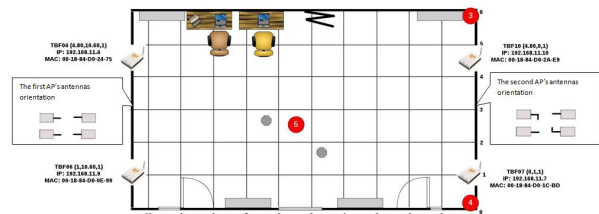


Figure 10. Scenario 12

- In this experiment We make the same scenario 1 ?? but for Take the measurement for point 1 and test of room temperature.
- In this experiment We make the same scenario 13 but for Take the measurement for point 1 and test of room humidity.
- *Experiment 15:*  
During this experiment Repeat scenario 1 ?? but replace the client terminal by a human operator. Launch the auto-calibration during the experiment to to see the importance of human presence in a room.
- During this experiment Repeat scenario 4 ?? but replace the client terminal by a human operator. Launch the auto-calibration during the experiment to to see the importance of human presence in a room.

### IX. OPERATIONAL METHODOLOGY

Wi-Fi Access Points (WfAPs for short) are configured to catch the attenuation of signal from a message emitted by WfCD. The WfCD emits 20 or more (depend on Scenario experiment) serialized explicit positioning requests (ePR for short). The time delay between 2 ePR is parameterized depend on scenario experiment for example 10ms for scenario 1. The WfCD reproduces sending 20 serialized ePR , during 1m. between each serie at the same PP, the time delay is parameterized to 500 ms.

### X. RESULTS

Our goal in this experiment is to test the impact of auto-calibration in presence of one or two person , presence of door or wall ,different client height and client orientation. we first find that RSS is not stable of one location each time because of path loss,Shadowing(absorption, reflection, scattering, and diffraction),multipath,long term fading. for example in first experiment in empty room without human factor and client at waist level . RSS from client 48:5d:60:28:37:8d (first point)to AP 00:18:84:d0:0e:99 during position request RSS from client (first point) 48:5d:60:28:37:8d to AP

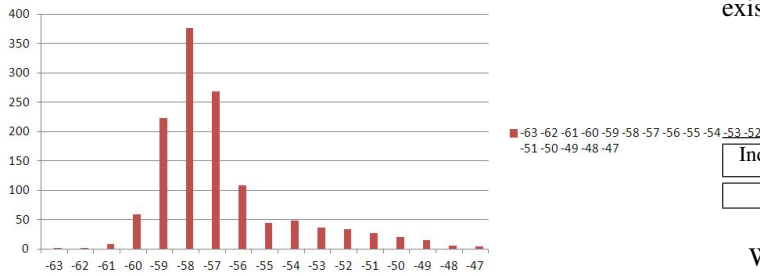


Figure 11. RSS changes in SC1 AP1

00:18:84:d0:1c:bd in scenario 2 we client on the floor the RSS change also when height of client on floor (RSS from client 48:5d:60:28:37:8d (first point)to AP 00:18:84:d0:0e:99 during position request). ?? in experiment 4 our goal is to know the impact of change the client his place. we measure the auto-calibration initialized when client in point 1 and

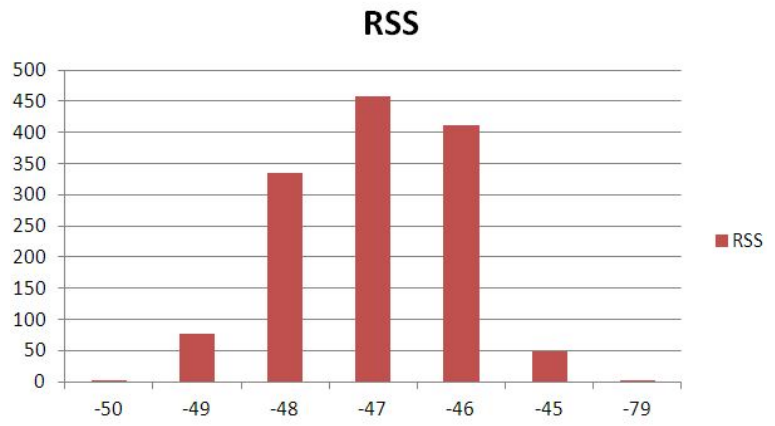


Figure 12. RSS changes in SC1 in AP4

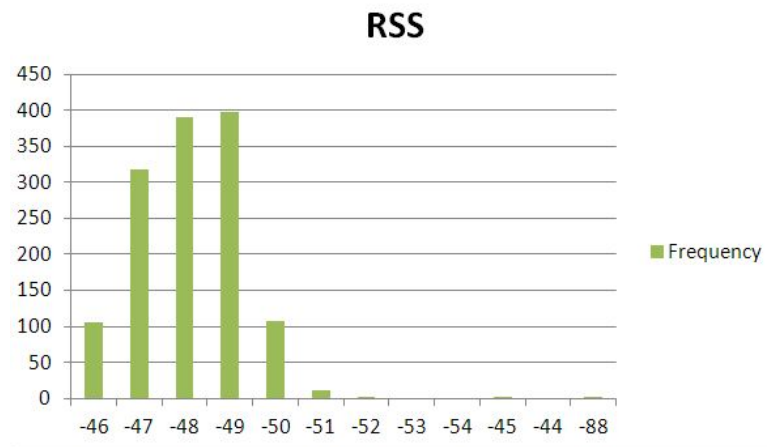


Figure 13. RSS changes in SC2 client on floor

when client in point 5 8 and and we calculate Index of Friis equation . For scenario 6 we make the same operation but for discovering the impact in door . For scenario 7 we make the same operation but for discovering the impact of door exist and two persons in the context .

Table II  
TABLE OF RESULTS

Index Friis	Person in room	person moving	door	two person and door
N	3.356	3.034	3.4596	3.3051

With these result we can conclude that

- 1) Instability of RSS in empty room and when human change his position.
- 2) Instability of auto-calibration in presence of human and when environment change(door open/close).
- 3) Estimation error is between 1.5-2.5 min OWLPS system

we can not predict whether there are any interference factor between the closet AP and TM and for that choose the most stable APs to calculate the position .



## CONCLUSION

In this work, we have presented an experimental system for position(OWLPS) based on RSSI and we make an application android helpful for test this system using model propagation Friis . The result of the experiment shows that in the indoor environment, the degree of accuracy of this position system is around 1.5-2.5 meters. This demonstrates this system has a high degree of accuracy, which is quite meaningful for future research work. There exists potential to increase the position accuracy. Thus the future work will be focus on improving position algorithm training or implementing different position algorithm on this platform to choose the most stable APs aren't on line of sight. In our experiments we focus on a lots of factor to measure the impact of factor and the changes of environment:

- 1) Impact of auto-calibration;
- 2) Impact of human presence;
- 3) Impact of environment change like wall or opened/closed door;
- 4) Stability of RSS in human presence.

In the end we try to show the impact of factors in OWLPS system and use specific APs to calculate the position without time consuming and we wait for result from Matteo for make comparison. In the future work will be focus on improving position algorithm or implementing different position algorithm on this platform and extract similarity function to estimate the position accuracy.

## REFERENCES

- [1] Matteo Cypriani—Adrien Henriet—Philippe Canalda and François Spies. Étude de l'interception et du positionnement de trafic wi-fi dans un environnement hétérogène.
- [2] Matteo Cypriani—Frédéric Lassabe—Philippe Canalda and François Spies. Open wireless positioning system: a wi-fi-based indoor positioning system.
- [3] Matteo Cypriani, Philippe Canalda, Soumaya Zirari, Frédéric Lassabe, and François Spies. Open wireless positioning system. *LIFC-Laboratoire d'Informatique de l'Université de Franche Comté, Technical Report RT2008-02*, 2008.
- [4] Matteo Cypriani, Adrien Henriet, Philippe Canalda, and François Spies. Étude de l'interception et du positionnement de trafic {Wi-Fi} dans un environnement hétérogène. In *C&ESAR'09, Computer and Electronics Security Applications Rendez-vous-Session Réseaux sans fil*, pages 4–19, 2009.
- [5] Matteo Cypriani, Philippe Canalda, and François Spies. Owlps: a self-calibrated fingerprint-based wi-fi positioning system. In *International Competition on Evaluating AAL Systems through Competitive Benchmarking*, pages 36–51. Springer, 2011.
- [6] Matteo Cypriani, Frédéric Lassabe, Philippe Canalda, and François Spies. Wi-fi-based indoor positioning: Basic techniques, hybrid algorithms and open software platform. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, pages 1–10. IEEE, 2010.
- [7] Matteo Cypriani, Philippe Canalda, François Spies, and Ancuta Dobircu. Benchmark measurements for wi-fi signal strength-based positioning system. In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, pages 1–8. IEEE, 2012.
- [8] M Cypriani. Géopositionnement wi-fi autocalibré en milieu hétérogène. *Université de Franche-Comté*, 2012.
- [9] M Cypriani, Ph Canalda, and F Spies. Performance evaluation of a self calibrated 3d wi-fi fingerprinting positioning system. In *Communications and Electronics (ICCE), 2012 Fourth International Conference on*, pages 274–278. IEEE, 2012.
- [10] Frederic Lassabe, Philippe Canalda, Pascal Chatonnay, Francois Spies, and Oumaya Baala. A friis-based calibrated model for wifi terminals positioning. In *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a*, pages 382–387. IEEE, 2005.
- [11] Frederic Lassabe, Philippe Canalda, Pascal Chatonnay, and Damien Charlet. Refining wifi indoor positioning renders pertinent deploying location-based multimedia guide. In *20th International Conference on Advanced Information Networking and Applications-Volume 1 (AINA'06)*, volume 2, pages 126–132. IEEE, 2006.
- [12] Frédéric Lassabe. *Géolocalisation et prédiction dans les réseaux Wi-Fi en intérieur*. PhD thesis, Université de Franche-Comté, 2009.