Cooperative Mobile Positioning

Data Fusion by

Support Vector Machines (SVMs)

Philippe LAURENT
07gr1124@es.aau.dk

June 2007, Aalborg University

Aalborg University
E-Studyboard
Mobile Communications 10th semester

TITLE:
Cooperative Mobile Positioning, Datafusion by Support Vector Machines (SVMs)

THEME:
Mobile Radio Communication

PROJECT PERIOD:

1st February – 7th June 2007

PROJECT GROUP:
gr 1124

GROUP MEMBER:
Philippe Laurent

SUPERVISORS:
Simone FRATTASI
Joao Figuerias

Number of duplicates: 3

Number of pages in the report: 61

Abstract:
Emerging location based applications like security, monitoring, tracking, emergency and others require accuracy in localization information anytime and anywhere. This represents a great challenge for researchers and industry. We investigate a data fusion based on SVM. Previous works have already been done. The innovation in this report comes from the RSSI measurements that are the SVM input, and from the indoor (with AP, computers and MS) and specific outdoor (only AP and MS) environments considered. Algorithms were implemented to improve the MS position estimations (SVM output). We obtain the best results in indoor with 70% (60% in outdoor) of MS well predicted thanks to computers and MS cooperation. More cooperative algorithms can be associated with the previous ones to reach peak performance.

Contents

Acknowledgement

I would like to express our gratitude to our supervisors Simone Frattasi and João Figueiras for their valuable guidance throughout this project and all those people who directly or indirectly helped us to finish this project.

Philippe LAURENT

List of abbreviations

1D	1 Dimension
2D	2 Dimensions
3D	3 Dimensions
3G	3^{rd} Generation
4G	4^{th} Generation
AOA	Angle of Arrival
AP	Access Point
BS	Base Station
CSVM	Cooperative Support Vector Machine
GPS	Global Positioning System
ICA	Indoor Cooperation Algorithm
LOS	Line of Sight
LSVM	Lagrangian Support Vector Machine
MRA	Maximum Response Algorithm
MS	Mobile Station
NLOS	Non-Line of Sight
OCA	Outdoor Cooperative Algorithm
RSSI	Received Signal Strength Indication
SVM	Support Vector Machine
TDOA	Time Difference of Arrival
WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network

List of Figures and Tables

1. Introduction

Until now a lot of work regarding geolocalization has been done. In wireless networks the goal was to localize as accurately as possible a sensor or a mobile station. Identifying the position of sensors nodes or mobile stations is really interesting and useful for the resource allocation and also to manage the network in term of efficiency.

Several techniques have been implemented. They deal with different kind of information TDOA, AOA or RSSI. Let us take GPS where TOA measurements are considered. This technology does not fit when we consider urban environment because of the presence of buildings. Moreover, the cost is not negligible due to the high battery consumption.

Other work has been done using the same kind of data as input. From those they can estimate the location of the sensor nodes. However, these techniques are subject to noise and to the cost of the localization. Indeed, to estimate mobile locations for instance we need BS which have to deal with several mobiles inside the coverage area.

1.1 Related work:

Regarding localization using SVM there are 2 different interesting works which have already been done [1] [2].

The 1st work deals with RSSI signals and AP dispatched in the area considered. An SVM process is running in order to determine MS location according to the RSSI information. The output of this system, namely the estimation of the MS positions depends on a weighting process in which each MS position depends on his surrounding neighbors (AP). Probability calculations and weighting functions attributed to the MS's neighbors give a predicted position.

The 2nd work concerns a dispatch of a certain amounts of AP in a square area running SVM according to hyperplans (more explanations will be found in the SVM part 3.3). The project doest not handle RSSI measurements but distances between the AP. Moreover the environment considered does not include any obstacle (walls, buildings).

1.2 Thesis contribution:

In our project, we deal with RSSI data as in the 1st existing work but otherwise than determining MS positions considering mathematical probabilities and dealing with a regression function our work is based on classification of data.

Some of our Scenarios are based on the 2nd existing work in the sense that we classify data but we do not distance as input. In fact 2 signals received by a MS from 2 AP which are at the same distance to the MS can have 2 different RSSI. So our project deals with on more constraint. Moreover, we decide to use a more realistic environment. Finally, the number of APs seems too high in the existing work to fit with a real practical need.

We decide to take into consideration WLAN technology. In fact, we consider 2 different Scenarios. The first one is based on an outdoor environment in which w only consider LOS transmissions. The second is an indoor Scenario where both of LOS and NLOS transmissions are considered.

Above is the organization of the report:

- Chapter II explains the project description, which includes the Scenarios, the problem definition, the scope of the project and finally the necessary assumptions.

- Chapter III deals with the background theory regarding fundamentals of positioning, localization according to hyperplans and also SVM.

- In Chapter IV, we will talk about the protocols and network management, namely who runs SVM, transmit the data, and how to manage several APs in the same area.

- In Chapter V, all the algorithms implemented, namely LSVM, MRA, OCA and ICA are explained

- Chapter VI shows the simulation models, the results and the discussion about them.

- Finally, Chapter VII concludes the project by presenting a sum-up of the work that has been done plus an overview of the future work.

2. Project Description

2.1. Introduction

We use cooperative mobile localization in this project to obtain a better localization. Using ad-hoc system and RSSI measurements will permit us to get better accuracy regarding the calculations of the positions. But we need to devise an accurate and efficient technique for fusing data.

SVM has proved over the last 10 years its efficiency in solving classification problem [3] thanks to a strong mathematical theory and also the fact that it is adaptable to several kinds of data which has to be classified (text, image) [4]. From RSSI measurements in ad-hoc network and thanks to SVM we will be able to determine locations.

2.2. Scenario

In this project we consider 2 different Scenarios:

Outdoor Scenario:

We consider an infrastructure link between the AP themselves and also the same infrastructure link between the mobiles and the AP.

So we consider 1 communication possible:

- Short range – WLAN (802.11g) between the access points and several mobiles connected thanks to infrastructure-based (AP->MS) and infrastructure-less (MS->MS) communications for the Scenario 1.

All the AP are placed in the cell such that each mobile can communicate with the other mobiles. Let us consider that we have n different AP. We assume that the positions of k different APs ($k<n$) are known in order to determine the location of the (n-k) mobiles.

The environment and the communication are simulated in Matlab. Moreover the calculations of the location are also done according to Matlab and Borland C++.

Below comes a diagram (Figure 2.2.a) that represents the outdoor Scenario considered in the project:

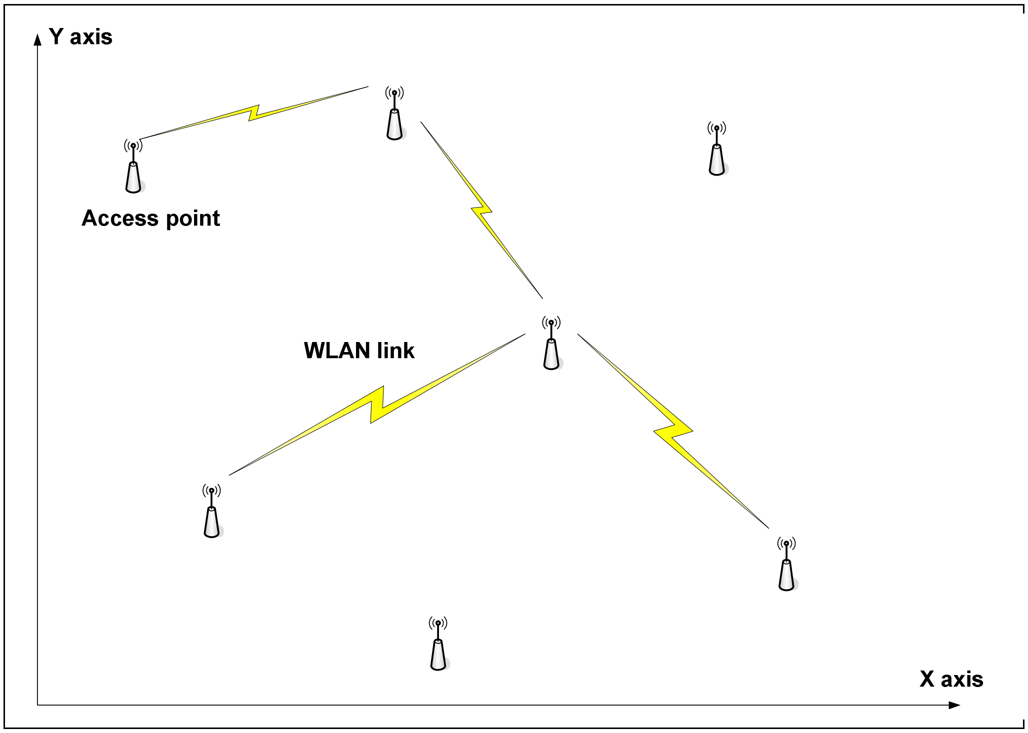

Figure 2.2.a: Outdoor Scenario with AP

In Figure 2.2.a several access points are dispatched all over the 2D field.
Each of them belongs to the WLAN network whose links are represented by the yellow flashes. Therefore, every access point can communicate with all the others within this network.

The position of all the APs is fixed so we assume that they are not moving in time. We will refer to them as beacon nodes.

The second diagram (Figure 2.2.b) represents the outdoor Scenario including some mobiles whose locations are unknown.
The goal is to determine their position according to the signals they receive from the beacon nodes. This is why we consider this WLAN network which permits to every beacon nodes and other nodes to communicate.

More explanations regarding the localization will be found in the part 3.2 of the report.

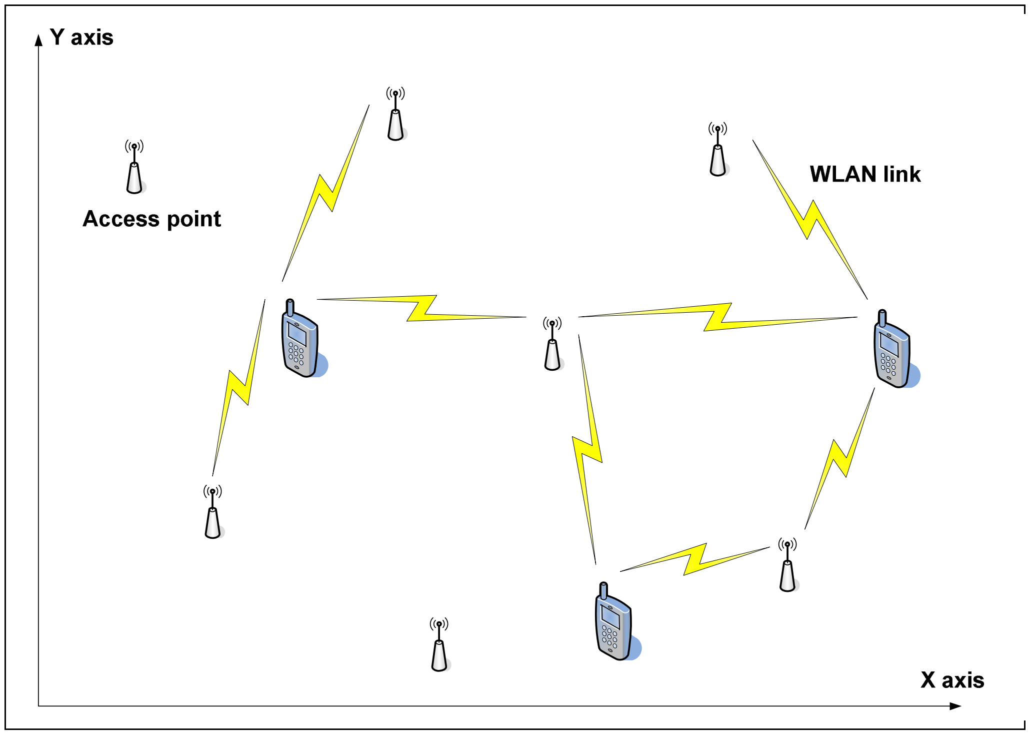

Figure 2.2.b: Outdoor Scenario of the project with mobiles

Advantages of this Scenario:

- We use an existing network with APs and mobiles.
- Once the AP are placed they are staying in the same place and the calculations regarding them only have to be done once. More explanations will be found in the Chaper 5 which describes LSVM algorithm.
- If at least one MS has been localized, then we can use it in order to localize a new mobile user moreover than taking into account the existing network (cooperation). The accuracy will be better (a Kalman filter can permit us to reach this better accuracy).
- No obstacles means LOS link.

Disadvantages:

- A lot of AP have to be dispatched within a certain range such that each of them can communicate with the others. We will see the minimum number of beacon nodes needed to get correct estimations in the Chapter 6 which describes the results obtained.

- It costs a certain amount of money to implement access points compared to the second Scenario in which we use existing desktops.

Assumptions:

1. We consider *n* AP randomly dispatched or fairly placed in the outdoor environment we consider.

2. The APs locations are know, only the MS ones have to be determined.

3. The communications between the different APs and between the AP and the MS are done thanks to respectively an infrastructure-based and infrastructure-less links.

<u>Indoor Scenario:</u>

We consider in that Scenario a floor where has been added a set of 3 AP fairly dispatched. The latter is linked to a hub and then to a server such as shown in Figure 2.2.c) which represents the Scenario. We also consider a set of computers which are placed in the middle of each room of the floor.

The aim is to predict in which room or where in the corridor the MS is.

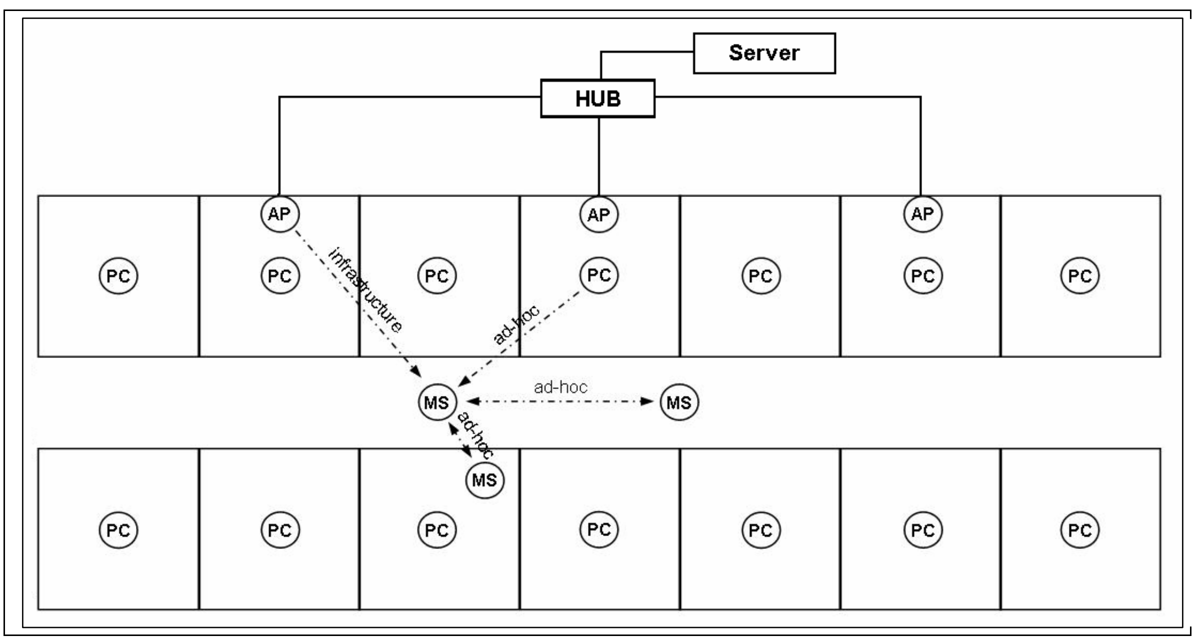

Figure 2.2.c: Indoor Scenario of the project

In Figure 2.2.c is presented the floor considered with 14 different offices equally distributed on both sides. Namely, there are 7 offices on each side. A corridor whose length is 70m, and width 5m separate the 2 set of rooms. We assume that each room is a 10-by-10m square. Figure 2.2.d sums-up the dimensions of the floor considered in this Scenario.

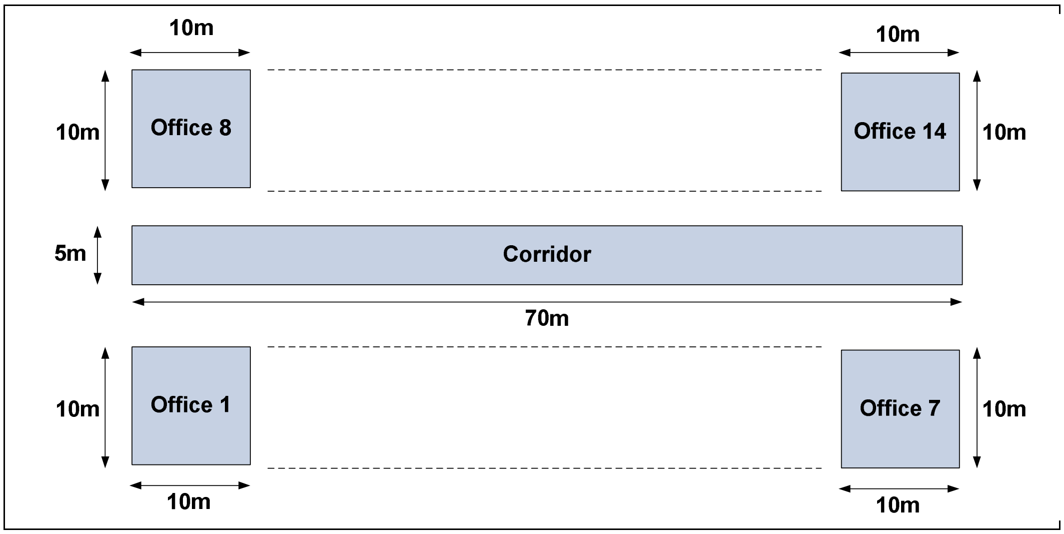

Figure 2.2.d: Dimensions of the floor for Scenario 2

Figure 2.2.e presents the indoor Scenario with AP and laptops or desktop computers and also the type of communication used.

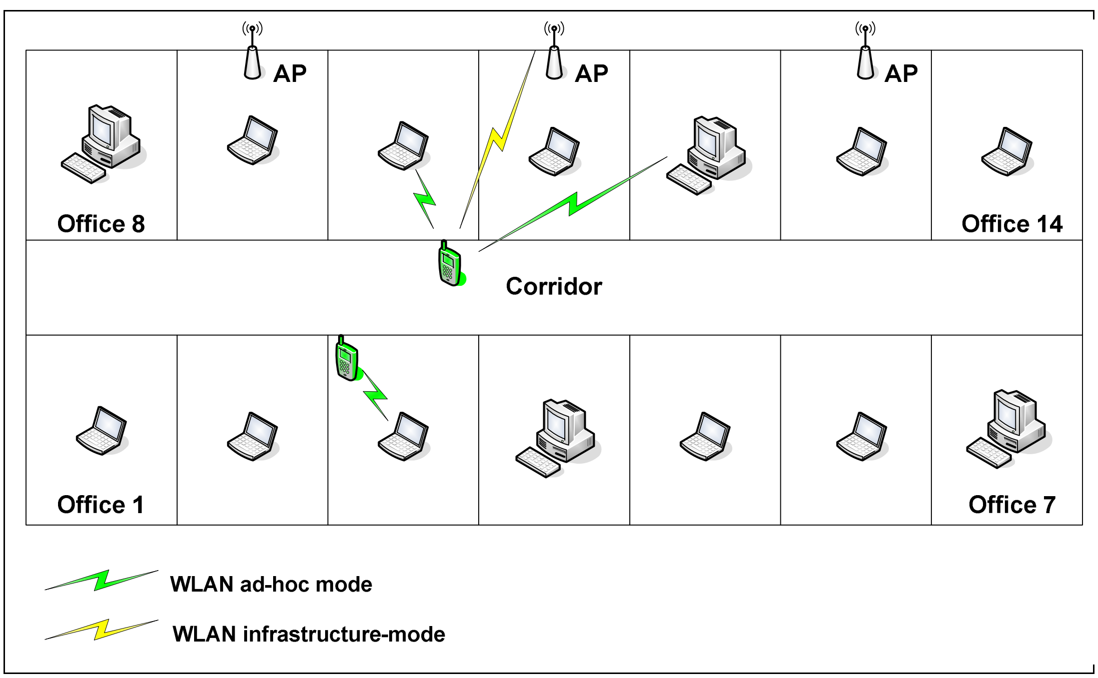

Figure 2.2.e: Indoor Scenario with APs and laptops or desktop computers

Advantages of this Scenario:

- We use desktop computers, laptops and the APs normally placed in each room or office of a building (university, company, etc).

- Thanks to the fact that the environment is not varying (the AP and computers are fixed) we will see in the Chapter 5 (SVM description) that it will permit us to save time in the localization process.

Disadvantages:

- Due to indoor environment the RSSI could not be reliable in some cases due to obstacles (e.g: doors, walls, etc). So we have to take into account it in the model we use.

- We need a sufficient number of desktops to estimate reliably the location of a mobile user.

Assumptions:

1. We consider 3 access points located in 3 different offices as shown in Figure 2.2.c.

2. The computer positions are known and do not move along the time.

3. The communications between computers and mobiles are in *ad-hoc* mode.

The simulations are done in matlab and C++ according to Matlab software and Borland C++ 5.5.

2.3. Problem Definition

1st Scenario:

According to the real distances between every AP we calculate the RSSI measurements. This is done thanks to the pathloss model. So let's take an example. If we consider 30 AP, every AP is going to receive 29 RSSI values (30-1). The input data for our SVM will be this RSSI information. The SVM is going to run once in order to output the calibration parameters (describing the system) called support vectors.

The latter, in a second time, will be associated to the RSSI information received by the MS from the APs in order to localize it. This association will be done by SVM again which is going to be run, therefore, a 2nd time.

2nd Scenario:

One AP will act as a head AP which means that in reality all the information will be gathered by it. First of all, we use the real distances between all the beacon nodes in order to calculate the RSSI information received by each of them. As in the previous Scenario the RSSI measurements come from the pathloss model which corresponds to an indoor environment for this Scenario. So if they are 20 beacon nodes, each of them is going to receive 20-1 = 19 RSSI measurements. Second, the latter is stored in order to work as input for the SVM. Based on this information SVM runs for the 1st time and outputs support vectors which corresponds to parameters of the system which depends on the number of beacon nodes, the way there are dispatched…Then when a mobile is coming into the floor it receives 20 RSSI data coming from all the beacon nodes. This information and the support vectors obtained previously are input in the SVM which runs for a 2nd time in order to determine the location of the mobile.

Procedure for both Scenarios which introduces cooperation:

We can use the estimated positions of some MS to find another one's. We will see if the use of cooperation (taking account estimated positions to find unknown ones) improves or not our results.

2.4. Scope of the Project

The aim of this project is to obtain a very accurate location for a mobile depending on the RSSI measurements from beacon nodes (APs, computers or other mobiles if we consider cooperation). In order to do so we must have a certain number of beacon nodes. Our simulations will show some results under different assumptions (number of beacon nodes for instance) for SVM.

The different steps followed in this project are:

- To set the mobile in a certain range of the cell and determine its RSSI values using the relevant communication in the networks (*ad-hoc*, *infrastructure*).

- Describe the Support Vector Machine model and configure the one which fits the best to our system and determine its parameters.

- Implement an algorithm using cooperation which will work for Scenario 1 and 2. It will work in addition to SVM and the goal is to compare it with SVM.

- Implement an algorithm which will use cooperation in a different way than the previous one and which can fit outdoor and indoor environment in an efficient way (Scenario 1 and 2).

2.5. Assumptions

We assume in this project that the positions of the beacon nodes (including the base station whose position is known in the outdoor environment) exist. From then we calculate the locations of the unknown mobile users.

1st Scenario

- All the MS and access points are in LOS with each other
- The links AP->AP and AP->MS are infrastructure ones
- The link MS->MS is an *ad-hoc* one
- We do not consider multipath
- We consider shadowing

2nd Scenario

- The links AP->AP and AP->MS are infrastructure ones
- The link MS->MS is an *ad-hoc* one
- We consider LOS and NLOS
- We consider shadowing
- We do not consider multipath

3. Background theory

3.1. Fundamentals of positioning

Positioning techniques:

We present a group of positioning techniques in this section. Basically, we can differentiate 3 different categories:

- Network-based:
  the AP performs both position measurements and computation of a location estimate.

- MS-based:
  the MS performs both position measurements and computation of a location estimate

- Mobile-Assisted:
  the MS provides position measurements to the network for computation of a location estimated by the network. The network may provide assistance data to the MS to enable position measurements and/or improve measurement performance.

In this project the Mobile-assisted method will be adopted. Above is the explanation of the TDOA technique and RSSI technique. The latter is the one we deal with in this project:

- TDOA technique:
  TDOA can be estimated by doing the cross-correlation between 2 different signals received by the MS from 2 different BSs. This technique will not be used in this project because we have to establish a connection between an AP and a MS or between 2 AP if we want to measure TDOA values. With RSSI technique, only the SSID and the RSSI measurements are required

- RSSI technique:
  With short distances this technique is the one which fits best. We based our calculations on the RSSI (in dB or dBm) received by one AP or a mobile from another AP or another mobile (in case of cooperation). The advantage of using this technique results in that being connected is not an essential need.

3.2. **Localization**

There are several ways to localize one MS. At the output the aim is to obtain an estimated position of the MS in 2D for instance. Some techniques permit us to get directly an estimated location with 2 coordinates. Others give us an answer according to an area [2]. Namely we have an answer to the question: “Does this mobile belongs or not to this area”.

Now if we think about the interest of this method, first of all we can divide quite easily any cell, field into n several areas (A1, A2…An) in order to determine if our mobile belongs or not to A1, A2…till An.

The Figure 3.2.a shows a subdivision possible of one area into 4 equal rectangles.

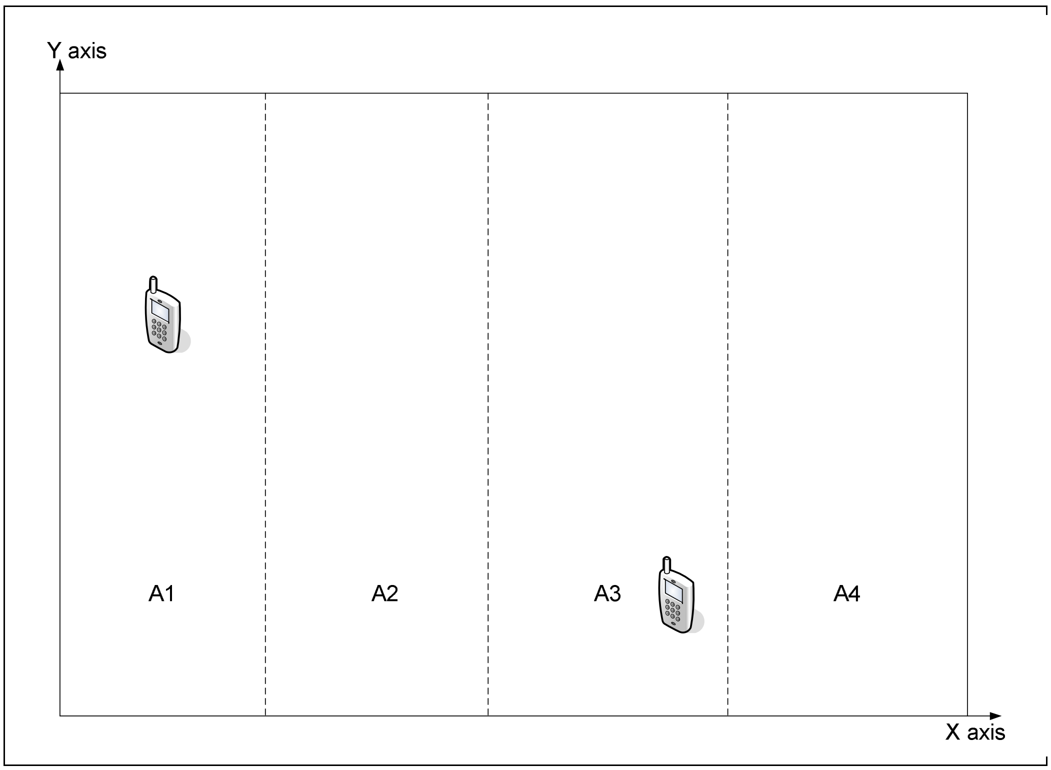

Figure 3.2.a: Localization using subdivided areas along the x-axis

The mobile 1 belongs to the area A1 and the mobile 2 to the area A3. This cutting out along the x-axis can also be done according to the y-axis as well as the Figure 3.2.b shows:

Figure 3.2.b: Localization using subdivided areas along the y-axis

In order to determine to which band (A1, A2 or B1 for instance) each MS belongs to we need of course a procedure and algorithms that can work under the constraints of Scenario 1 and 2 explained in Figures 2.2.b and 2.2.d.

The procedure we select is the one which corresponds the best to that kind of classification (classify the MS positions) according to signal strength measurements [1]. The procedure is Support Vector Machine.

3.3. Support Vector Machine

Introduction:

This method was invented in 1995 by Vapnik [5]. SVM is composed of SVC and SVR. The former concerns classification while the latter is a regression procedure. In this project we only use SVC, namely SVM as a classifier. SVC is a classification standard method which permits to classify some data thanks to some complex functions and calculations. We will see in the following section the theory behind SVM.

So it is a matter of 2 states (binary) classification [5]. Indeed at the input of an SVM we find real data which are going to be compared to previous collected data. A binary Figure is coming up at the output of this process. With that method we compare some values in order evaluate the degree of similarity between some of them.

Theory behind SVM

Among the kernels methods, coming from the statistic learning theory of Vapnik, Support Vectors Machines are standard methods which are the most famous. It consists of a binary classification within a supervised learning. Because it is a 2 states classification matter this method calls a learning dataset in order to learn the parameters pf the system. It is based on the use of a function called “kernel” that permits an optimal data separation.

As we can see on from Figure 3.3.a the goal is to separate the collected data according to a straight line, in order to create 2 groups. The straight line is called a *hyperplan*. The name of the closest points to the hyperplan is *support vectors*.

We consider an area E, that belongs to the cell considered. The binary classification the SVM uses is done according to the fact that the data belongs or not the specified zone.

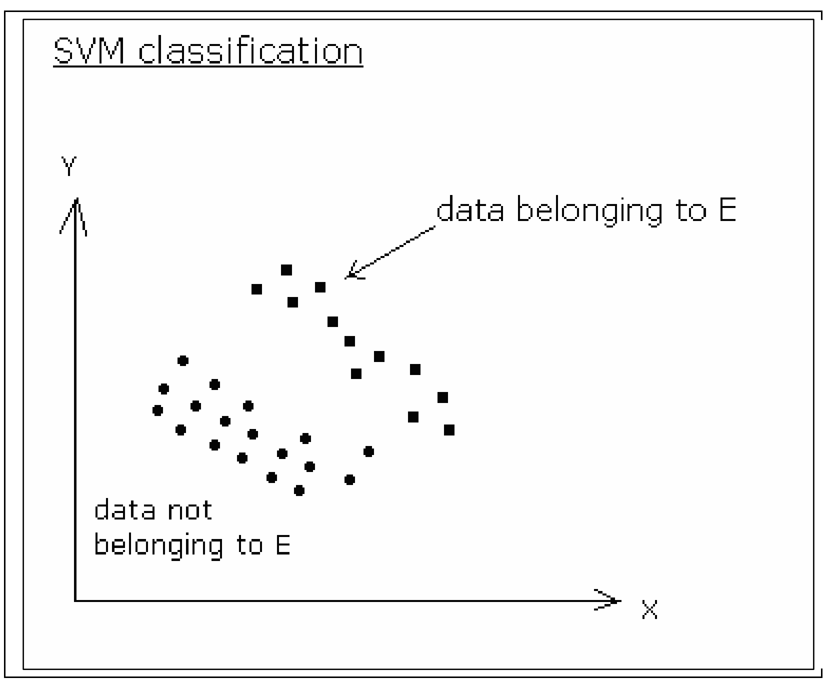

Figure 3.3.a: SVM classification of data

The data from Figure 3.3.a has to be classified according to a binary process in which we separate the one which belongs to the zone E and the other not. We are now looking at a hyperplan, that will permit us to separate the data.

Figure 3.3.b shows the second step of this classification process displaying a hyperplan. This hyperplan works as a limit between the 2 sets of data (left and right side).

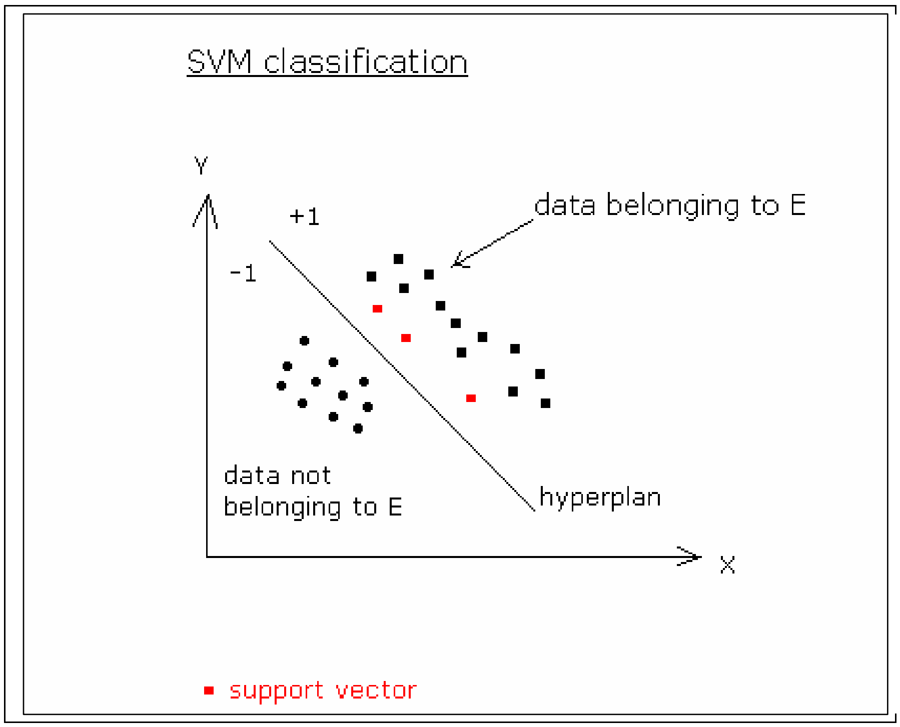

Figure 3.3.b: SVM binary classification and support vectors

The support vectors represented in red correspond to the closest points to the hyperplan belonging to E. We highlight them because they will have an important role in the classification process. However, there is not a unique hyperplan which can subdivide all the data into 2 groups. If we take the example coming from Figure 3.3.b we can see that several hyperplans can fit regarding the binary classification we would like to do.

Figure 3.3.c which presents the SVM classification, shows that several hyperplans can correspond to the one we want.

Thereafter, we will see how we can manage that consequent amount of hyperplans in order to select one and only one of them.

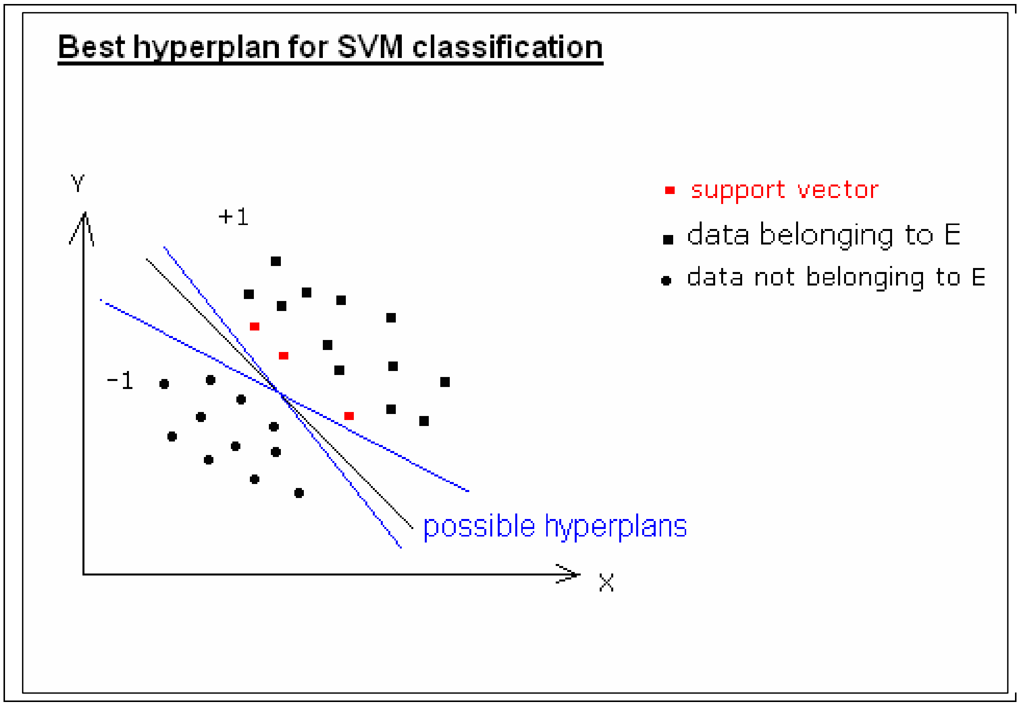

Figure 3.3.c: SVM classification thanks to the best hyperplan

All of them can separate the data according to the same 2 groups. So the goal from now is to find the best hyperplan, also called “optimal hyperplan” [6]. To translate that "optimal hyperplan" expression into geometrical aspects we can talk about maximizing the margin between the hyperplan and the support vectors. That will permit us to find the optimal one.

Previously in Figure 3.3.c we assume the hyperplan is a straight line but this is only true if it is a linear model. Indeed if some data which belongs to one group are located between other data belonging to the other group then a linear classification is not possible. In that case we need an unlinear model.

Unlinear classification:

We separate data thanks to one more dimension for instance. So, the goal is to evaluate the data according 3 dimensions and not only 2 such as the following example in Figure 3.3.d, which represents a kernel classification in 3 dimensions, show us.

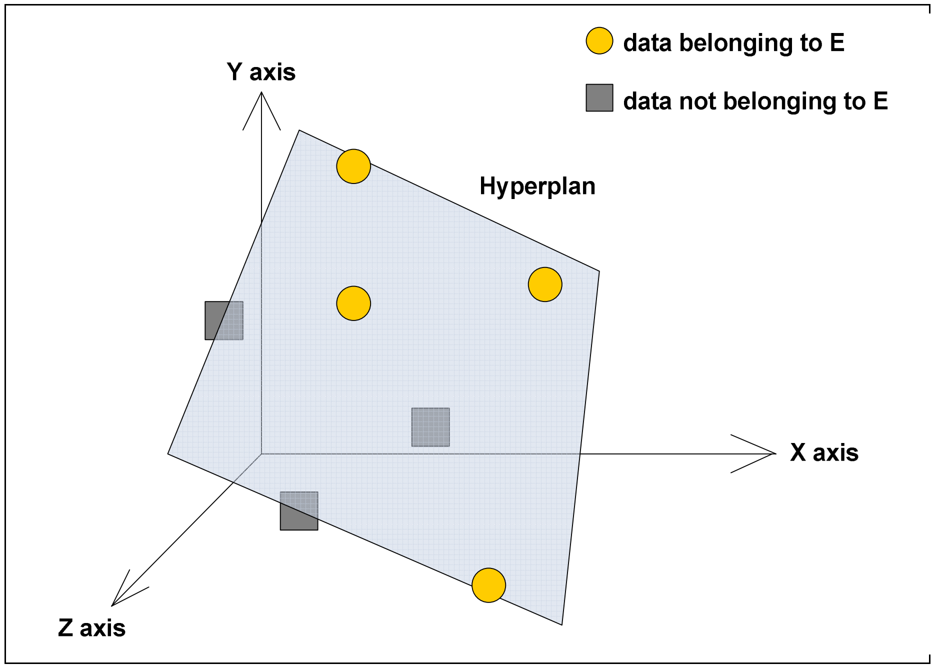

Figure 3.3.d: Classification using Kernel of n+1 (=3) dimensions

The classification process described in Figure 3.3.d is done by a kernel function which can be a polynomial, Gaussian or also Laplacian function.

Figure 3.3.e shows a classification example using a polynomial kernel.

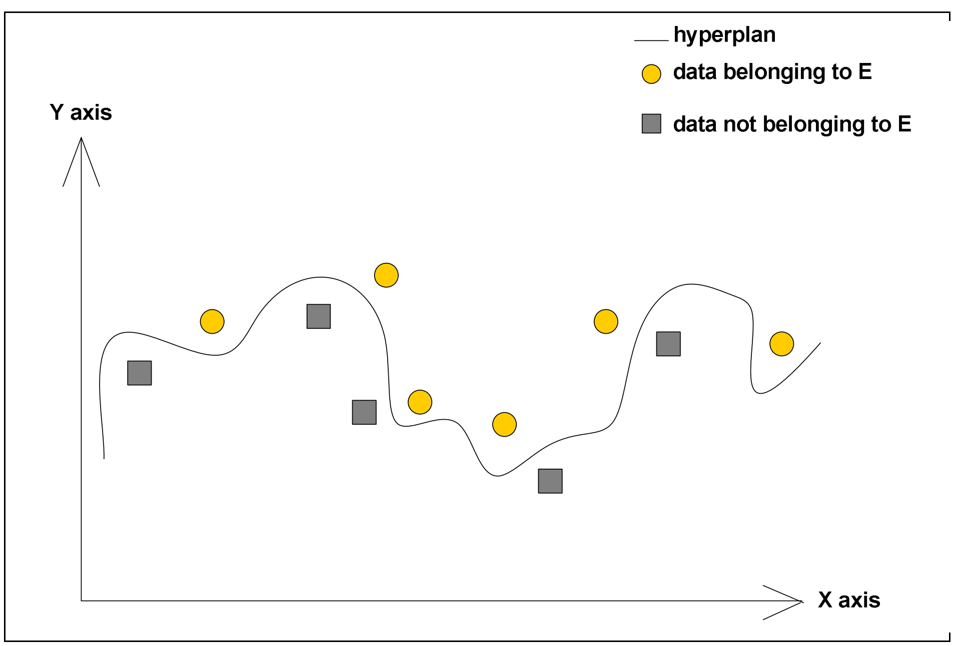

Figure 3.3.e: Classification thanks to a polynomial kernel function

3.3.1. Training

In the training phase we consider a dataset composed of signal strength measurements issued from the different beacon nodes. The goal is to compare these 2 different data for every couple of beacon nodes in order to evaluate the parameters of the SVM such as shown in Figure 3.3.1.a.

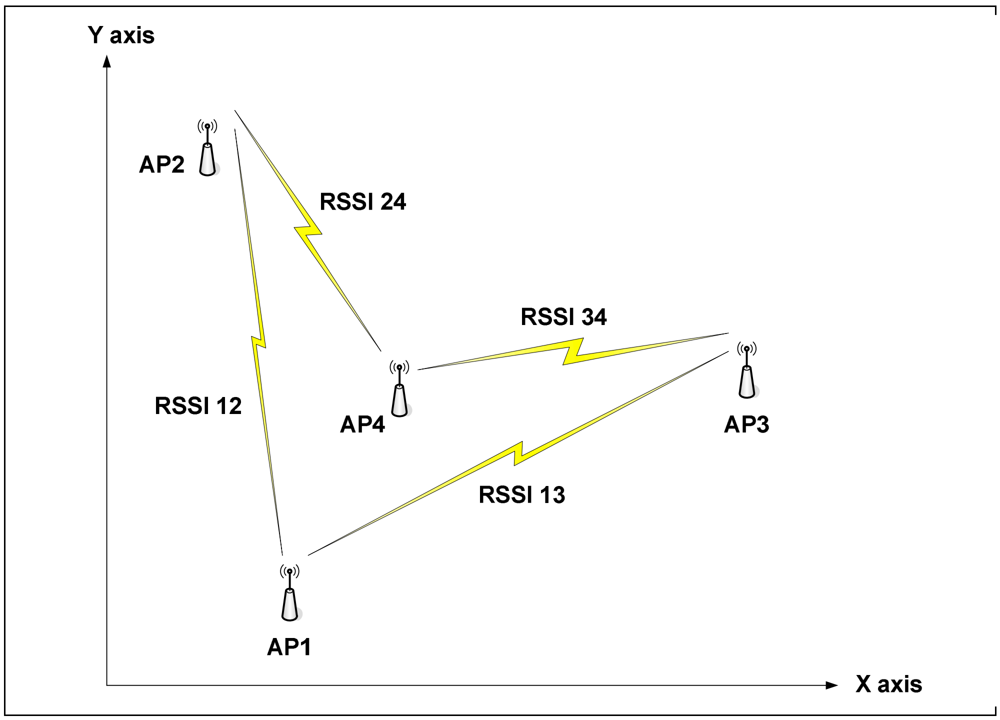

Figure 3.3.1.a: RSSI measurements for the training phase dataset

Figure 3.3.1.a assumes that all the RSSI information can be obtained. If it is not the case one solution will be followed. It consists of taking 2 mobiles, called MS1 and MS2. They are placed at the same position as the AP. From then MS1 is going to check such as a sniffer the RSSI received from the MS2. Moreover the SSID of MS2 will also be received with the RSSI so both data can be gathered. We repeat the same procedure for all the links where the RSSI cannot be calculated

All the RSSI measurements collected are useful for the input of our SVM. We take into account them for the training phase that is the first part of the SVM procedure. Thereafter, in the classification phase other RSSI values are going to be gathered. Section 3.3.2 of the report goes more deeply in this classification part.

Let us see now the algorithm use in order to classify the training data, it is explained in Figure 3.3.1.b

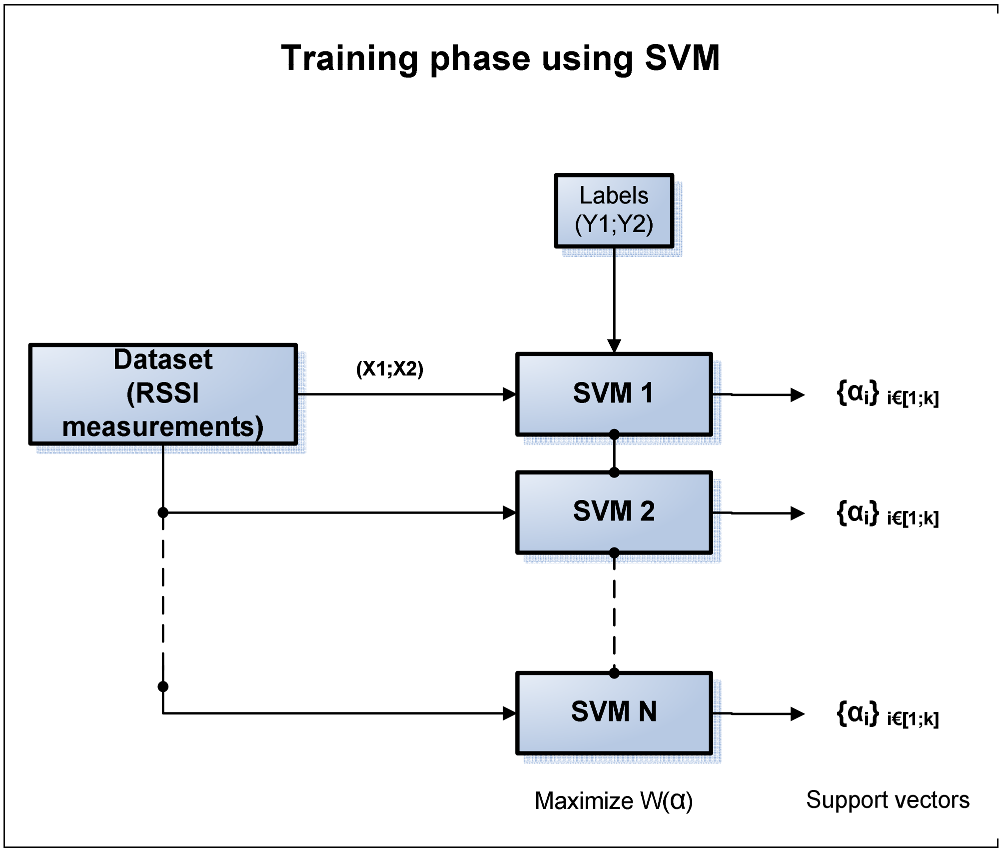

Figure 3.3.1.b: SVM Training phase

First, we have some input called X1 and X2, which have to be compared in order to do a classification. These data could be TDOA or RSSI. In our project we only take into account RSSI measurements. Second, the SVM intervenes in the process in order to do the training. It means to look at the data at the input such as for instance, some distances between 2 nodes and evaluate the support vectors as explained in the theory of SVM (section 3.3). So every SVM will deal with X1 and X2 but also with Y1 and Y2 which are binary values equal to -1 or 1.

The number of SVM depends on the number of hyperplans according to both x-axis and y-axis. Namely if we consider a field divided into *p* different bands (*p* integer >=1) according to the x-axis there will be *p* SVM calculations for the training phase. Regarding now the value of the label Y we see at the top of the Figure 3.3.1.b, it indicates if the beacon node belongs or not to the hyperplan considered. The goal of the SVM is to maximize a function called $W(\alpha)$ in order to obtain support vectors which act like some limits in the classification process.

All this training phase is detailed according to the example we took in the part 5.1.a titled “Training phase”.

3.3.2. Classification

The classification is done taking into account the same kind of data as the input. Namely we deal RSSI measurements coming from the beacon nodes in the training phase. In the classification one RSSI measurements are also considered but they are issued from the links between all the beacon nodes to the MS. The difference between the classification and training phase, regarding the input, results in the first 1^{st} term.

In the training phase we compare values such as RSSI measurements between nodes whose position are known. It is therefore a sort of calibration. In the classification phase we are interested in finding out the position of one node according to its RSSI values with respect to the other nodes whose positions are known (see Figure 3.3.2.a).

To sum-up we have first a calibration regarding some nodes whose location is known in order to find the features (support vectors) of the SVM system. Thereafter, we can apply this SVM system one more time in order to localize a new mobile coming up in the network.

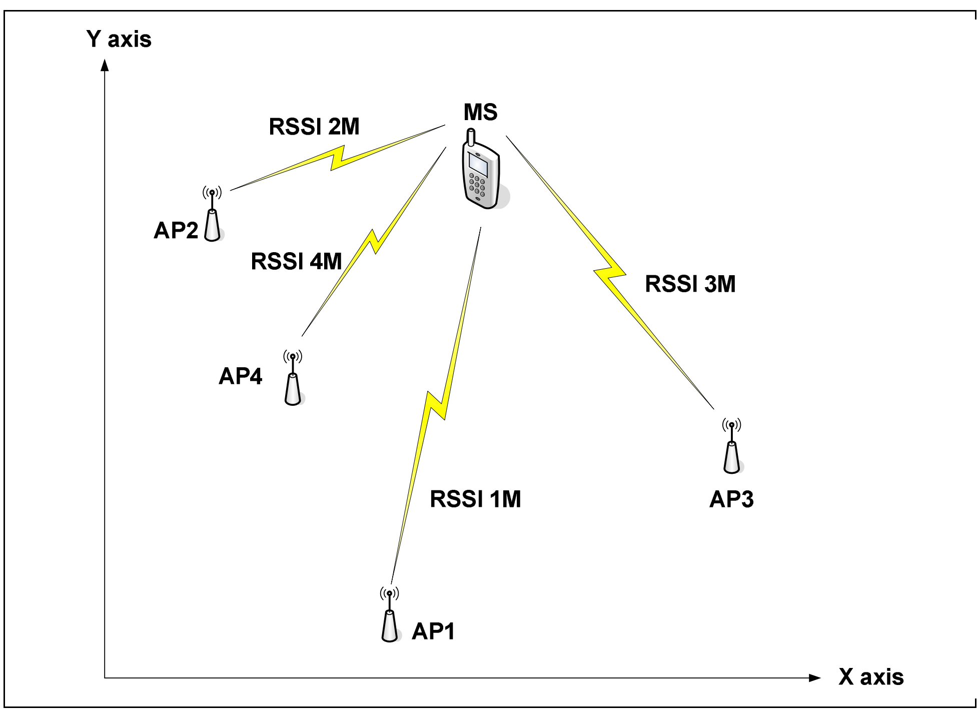

Figure 3.3.2.a: Input data for the classification phase

The RSSI measurements issued from every AP->MS link for each new mobile are gathered and are used in the classification procedure such as input data for the SVM.

The entire classification process is described in Figure 3.3.2.b:

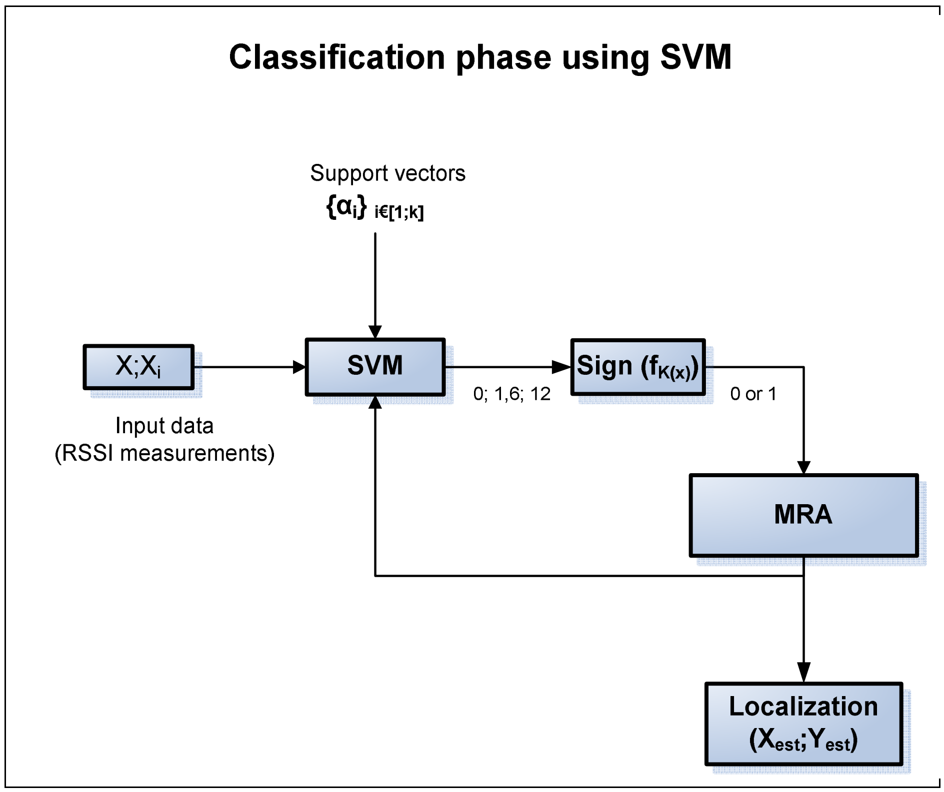

Figure 3.3.2.b: Classification process using SVM

SVM deals with the input data as explained above and also the support vectors in order to classify new data (for instance, to localize a new mobile user in the network). The support vectors work as limits in the classification. It is the reason why we use the “sign” function which outputs a binary value 0 or 1. Indeed if the difference of signal between the mobile user and a node A is almost the same as the difference of signal between the node A and another node C then the output of the “sign” function will be 1.

The Maximum Response Algorithm (MRA) is an algorithm which is going to be used in this project in order to localize the new MS which belongs to the network as efficient as possible. More explanations will be found in section 5.1.b.

4. **Protocols and network management**

4.1. SVM calculation

The 2 phases included in the SVM process, namely the training and the classification phase, are run by one and only one AP, that is considered as a “head AP”. Indeed it is more relevant to have one “leader” to manage our WLAN network.

Its job is to gather all the RSSI information coming from all the other APs (training phase) in order to run the SVM once to outputs the support vectors. Then, every time a MS is coming up in the network the head AP sends it the SVM parameters. Afterwards, the MS has to do the calculations itself in order to know where it is.

In each Scenario that we will present in this report there will not be any IP connection between APs and APs. Otherwise there must be a connection between either APs and MSs or computers and MSs (2^{nd} Scenario) because the head AP owns the parameters of the system after the training phase. Moreover the RSSI values are only known by the MS. Therefore the best solution will be for the head AP to send the features of the system (support vectors) to the MS in order for him to do the 2^{nd} classification phase and find his location. This solution will avoid to the MS to send all the RSSI values obtained from the beacon nodes, to wait for the calculations and the answer from the head AP.

4.2. Interference problem

Another issue must be raised. How to deal with several AP in the same area? The answer is 3 taking into account the 802.11g standards. The use of the channel 1, 6 and 11 permits to avoid overlapping but it narrows our Scenario.

To deal with that problem we can order the n AP considered from 1 to n in order not to transmit at different time slot. This will solve the interference problems. Moreover the training phase is only done once so this solution will rather be interesting during the classification phase.

5. Algorithms

We have implemented several algorithms based on SVM. In this section all of them will be presented. First of all we describe LSVM and MRA (training phase of LSVM) algorithms that will be used for the simulations. Then CSVM, which consists of SVM added with a cooperative algorithm, will be presented. The last section concern OCA (for Scenario 1) and ICA (for Scenario 2) that use, as CSVM, SVM as a first step and then deal with cooperation between APs and MS for OCA and APs, computers and MS for ICA.

5.1. LSVM and MRA

LSVM is a Support Vector Machine technique using lagragian coefficients. It is the fastest technique for training SVM [7]. Concerning the testing phase we use the MRA algorithm that permits to avoid running several times the training phase.

The environment:

We consider a 2D-grid geographic area with a certain number of cases (possible square area locations). Along the x-axis and y-axis we choose to have a longer *d* which will vary along our simulations. We take *M* different locations possible along each axis. So higher the value of *M* is, better will be the accuracy of the localization (see Figure 5.1.a):

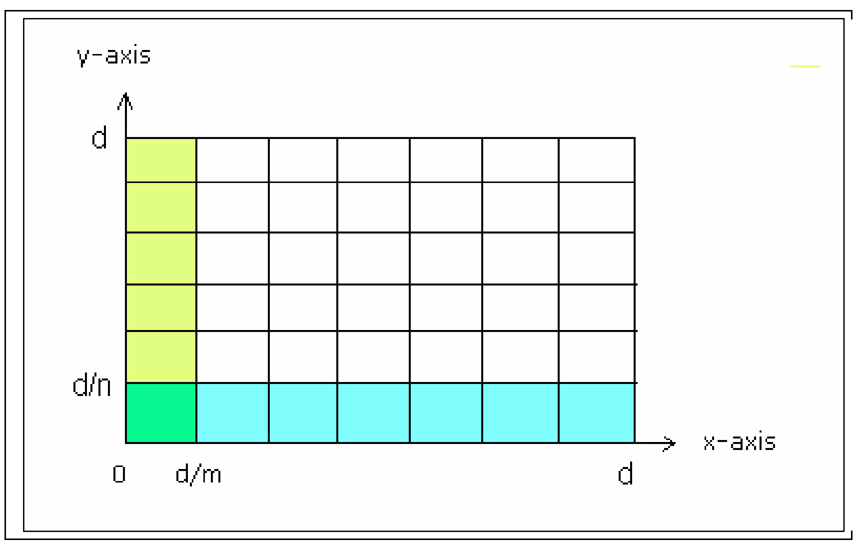

Figure 5.1.a: 2D-grid *n* x *m* dimensions

5.1.1 Training phase:

Firstly we consider several access points whose positions are known. Let us assume that we have *k* different ones. They are randomly placed in the [0, D] x [0, D] grid. The classification corresponds to a 2 classes-binary classification. The point belongs to a set *E* or not. We consider training data points, which correspond to beacon positions. *K* data have to be taken into account; namely $x_1, x_2 ... x_k$. We also consider *k* different labels, namely binary value [-1; 1]. y_1 = 1 means x_1 belongs to the set E. In the previous example E stands for one of the square shown in the previous picture.

Second we define the relevant Kernel function [8]:

$$\boxed{Kerf(S_i, S_j) = \exp(-\gamma \|S_i - S_j\|^2)}$$

where i and j are values that range between 1 and *k*. They stand for the indices of the beacon node considered. S_i is a vector that contains the *k* different shortest distances between the beacon node i and its (*k*-1) beacon neighbors. γ is a positive constant that was determined during the training phase.

Then, we have to maximize the following equation

$$\boxed{W(a) = \sum_{i=1}^{k} a_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} a_i a_j y_i y_j K(x_i . x_j)}$$

under the 2 following constraints

$$\boxed{\sum_{i=1}^{k} y_i a_i = 0} \quad (1)$$

$$\boxed{0 \le a_i \le C} \quad (2)$$

where a_i, i$\in [1; k]$ are the solutions of this optimization problem. *C* is a constant that permits to take into account the points that cannot be really well classified. Namely, the more you increase the value of *C*, the better the classification of the data will be.

Figure 5.1.b shows some examples of the same classification but using different values for *C*.

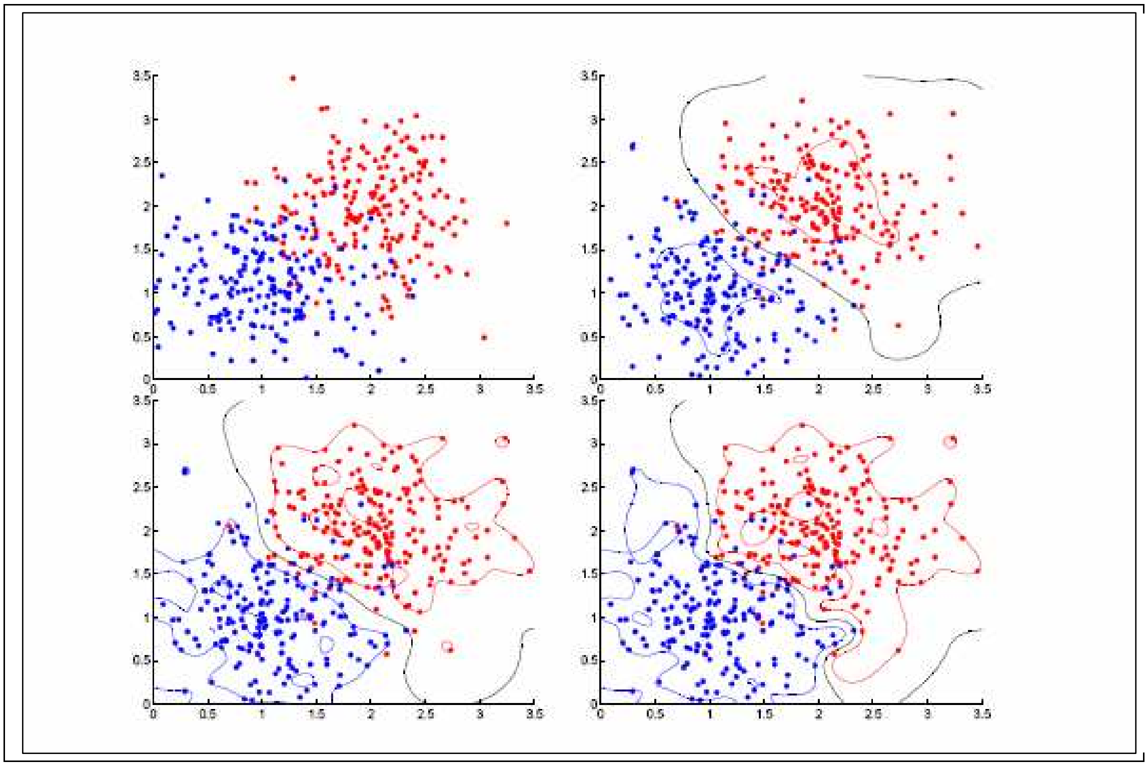

Figure 5.1.b: Results of classification under different values of C

As we can observe on the Figure 5.1.b, higher is C better will be the classification. Indeed in the 4^{th} diagram (down-right) all the red points which stand for values y = +1 (the values y = -1 are not in the set) are gathered. The goal is obtain the SVM model information, namely these a_i coefficients. The system model is then applied to the mobiles in the cell whose position is unknown. The aim of this process called the classification phase is to localize them.

5.1.2 Classification phase: MRA algorithm

Now each mobile in the cell whose position is unknown has to gather the k different RSSI measurements it receives. Let us say we consider N different mobiles. Once every mobile gets this information it applies the SVM in order to localize itself. The algorithm used is called “Maximum Response Algorithm”.

This algorithm is described in the following diagram from the input “support vectors” and also the data X and Xi. The diagram is the Figure 5.1.c.

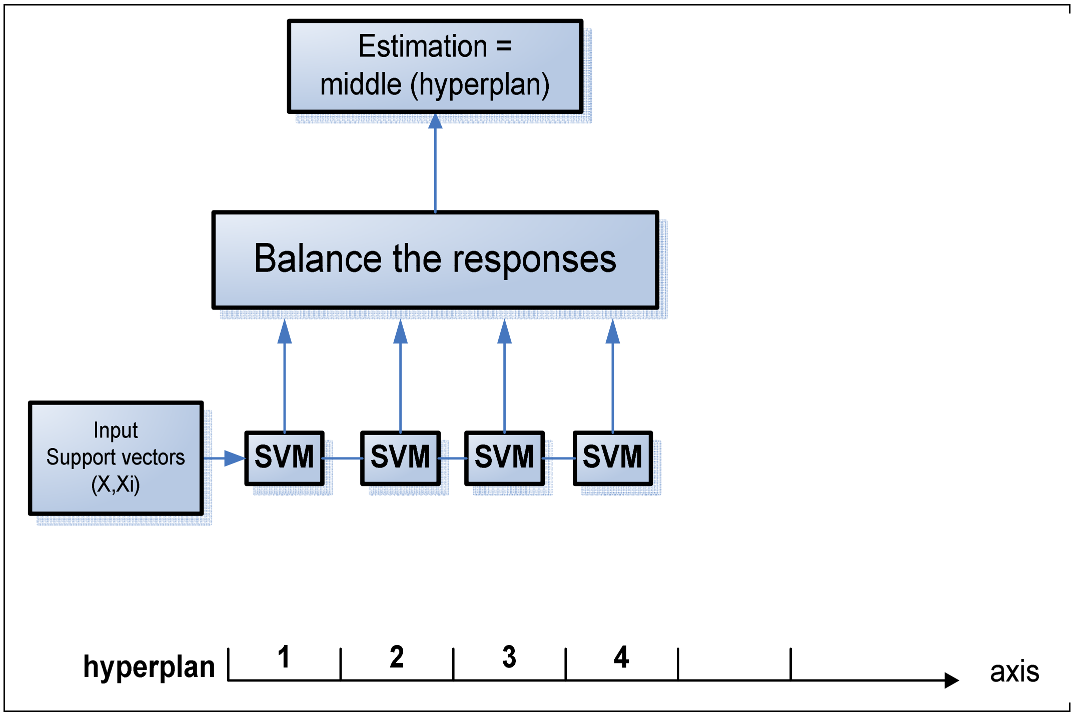

Figure 5.1.c: Maximum Response Algorithm

We just consider one axis but the procedure has to be repeated according to the other axis (to get an answer for *x* and *y*) as well. The cell has been divided into several different rectangular zones (hyperplans).

According to SVM we determine if the new mobile belongs or not to each hyperplan. Repeating this process several times (for each hyperplan) permits us to get the area on the x-axis the mobile belongs to (physically the zone where the mobile is).

This binary calculation is done thanks to the following equation

$$\boxed{h_{K(x)} = \sum_{i=1}^{k} a_i y_i K(x, x_{x_i}) + b}$$

where b represents the bias, calculated in the training phase. $\{a_i\}$ are the lagrangian coefficients coming from the training phase. $K(x, x_i)$ is the kernel function, which includes the RSSI values between each beacon node and the other beacon nodes, and the ones between each mobile and the beacon points.

Localization:

At the output of the system we get a square $\frac{d}{m}$ by $\frac{d}{m}$ because we take into account the cross between the 2 hyperplans obtained. One comes from the x-axis and the other one from the y-axis. We take the middle of this square in order to determine the x and y coordinates of the MS we have to localize.

Remark: according to the value of m, we can increase or not the accuracy of the localization. Indeed, smaller is the target zone, better will be the precision.

However after all the SVM procedure it happens that some MS are not localized. This can be due to the severe constraints of the environment, to the parameters that are too selective (strong value for C). In order to solve this problem another algorithm was implemented. Its name is CSVM. The next section describes it in details.

5.2. Cooperative Support Vector Machine (CSVM)

This algorithm is based on SVM. It is applied after at least one SVM runs in order to increase if it is possible the number of predictions. We call it Cooperative Support Vector Machine because cooperation appears among the MS. Indeed, in order to localize accurately a MS, which has not been localized by the first run of SVM, we can use the MS which have been localized.

To sum up the MS well-localized are going to become Beacon nodes, namely we will include them in the second training phase of the SVM (process now called Cooperative Support Vector Machine).

In Figure 5.2.a the algorithm CSVM is described:

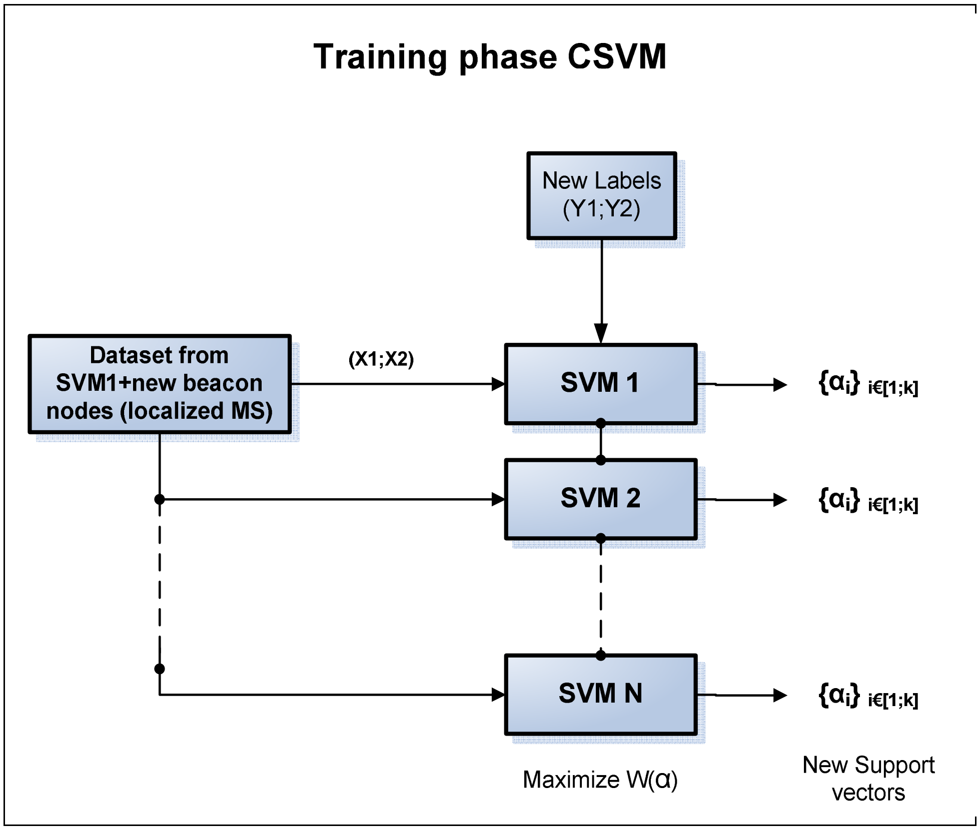

Figure 5.2.a: CSVM training phase

The SVM is going to run N+M several times; N corresponds to the number of hyperplans according to x-axis, M to the y-axis. So, there is no difference concerning the number of SVM runs for SVM and CSVM process.

However, the number of beacon nodes increase between the 2 runs (SVM and CSVM) because the MS well localized after the 1st classification phase (SVM) are now input in the dataset as shown Figure 5.2.a.

So, the training phase of CSVM will take much more time than SVM's one because we have to run twice the training phase otherwise than running it only once. Otherwise the classification phase will be faster thanks to a less amount of MS to localize. The CSVM classification phase is explained in Figure 5.2.b.

All the analysis and performance of CSVM will be done in Chapter 6 which talks about simulations, results and discussions.

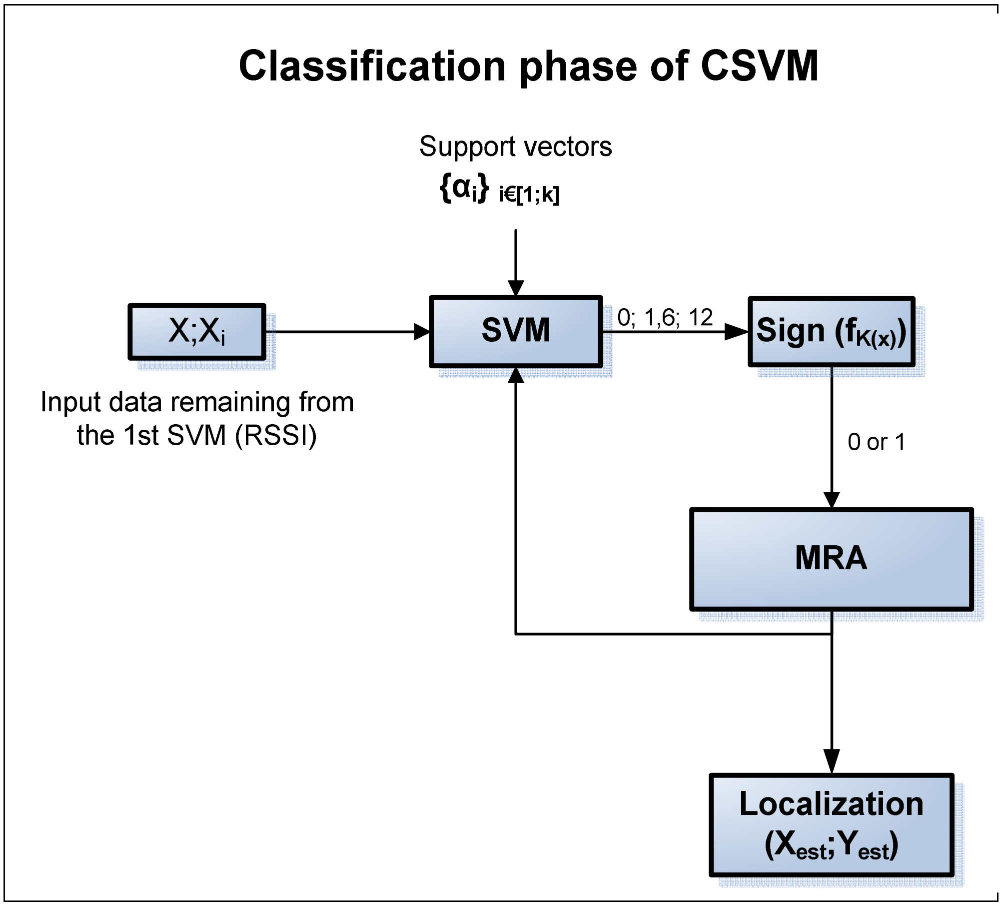

Figure 5.2.b: CSVM classification phase

5.3. Outdoor Cooperative Algorithm (OCA)

OCA algorithm is a part of the localization process in outdoor environment. Its goal is to better estimate the position of the MS after the SVM has been run once. In the simulations, a lack of predictions appears for some mobiles even using CSVM so the aim is to avoid getting no-predictions. In order to do so OCA deals with the RSSI measurements obtained from the distances. Afterwards, it assesses and stores the minimum RSSI value for each AP to MS links. For instance, if we put 50 APs and 10 MSs in the environment used in Scenario 1, OCA will keep track of 10 RSSI values, which stand for the minimum measurement for each of the 10 mobiles.

When the minimum RSSI value has been saved we are interested in dealing with all the MSs whose positions have not been predicted. This is why when there is no prediction according to the x and y-axis the OCA algorithm will calculate the estimated position taking into account the minimum RSSI value that has been stored.

Figure 5.3.a describes the OCA algorithm

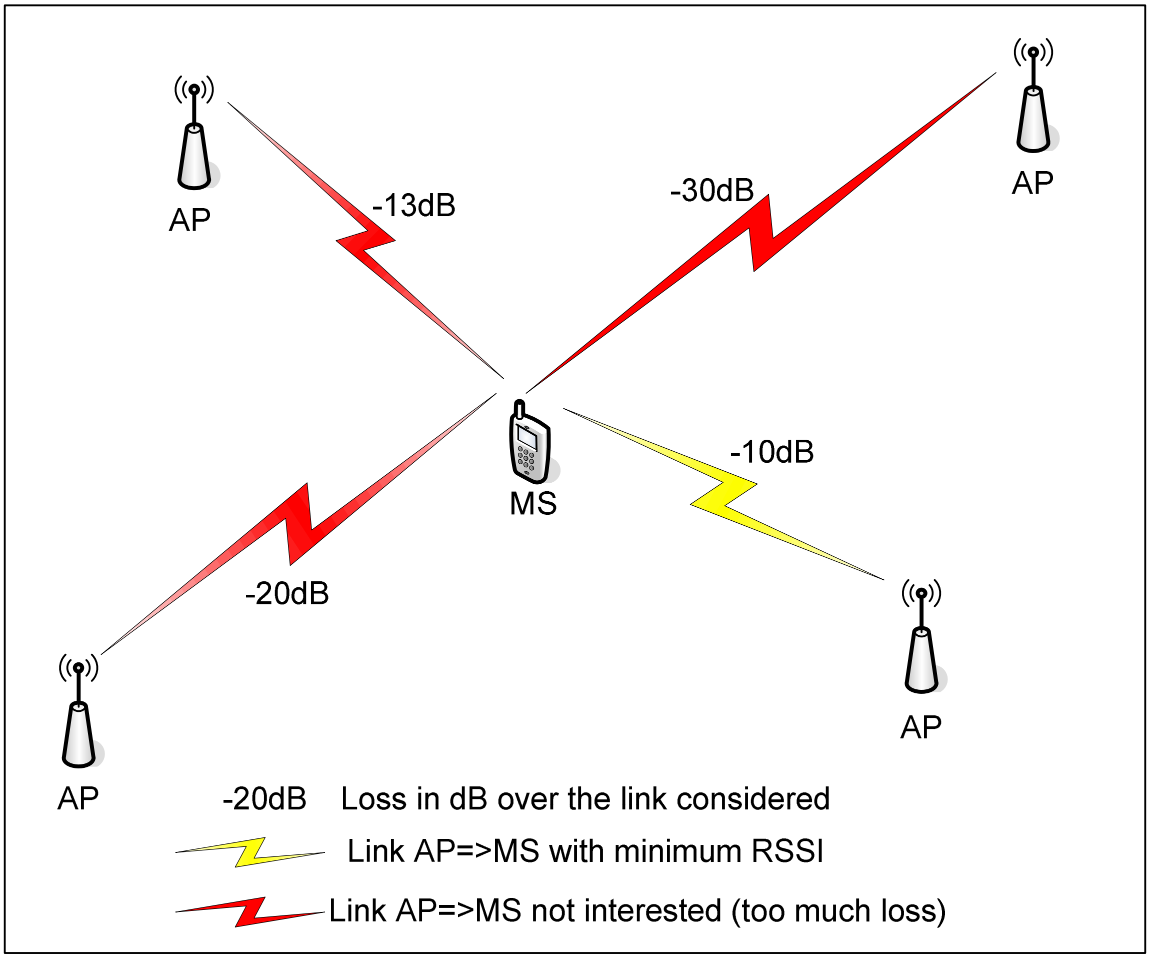

Figure 5.3.a: Outdoor Cooperative Algorithm (1st part)

From this procedure OCA will consider that the MS is located in the hyperplan where the AP is. This will give a better estimated position.

5.4. Indoor Cooperative Algorithm (ICA)

ICA works as OCA except that we use it in an indoor environment to predict the MSs' positions if there were no one. We apply it for the Scenario 2 represented in Figure 5.4.a following which gives the process of ICA:

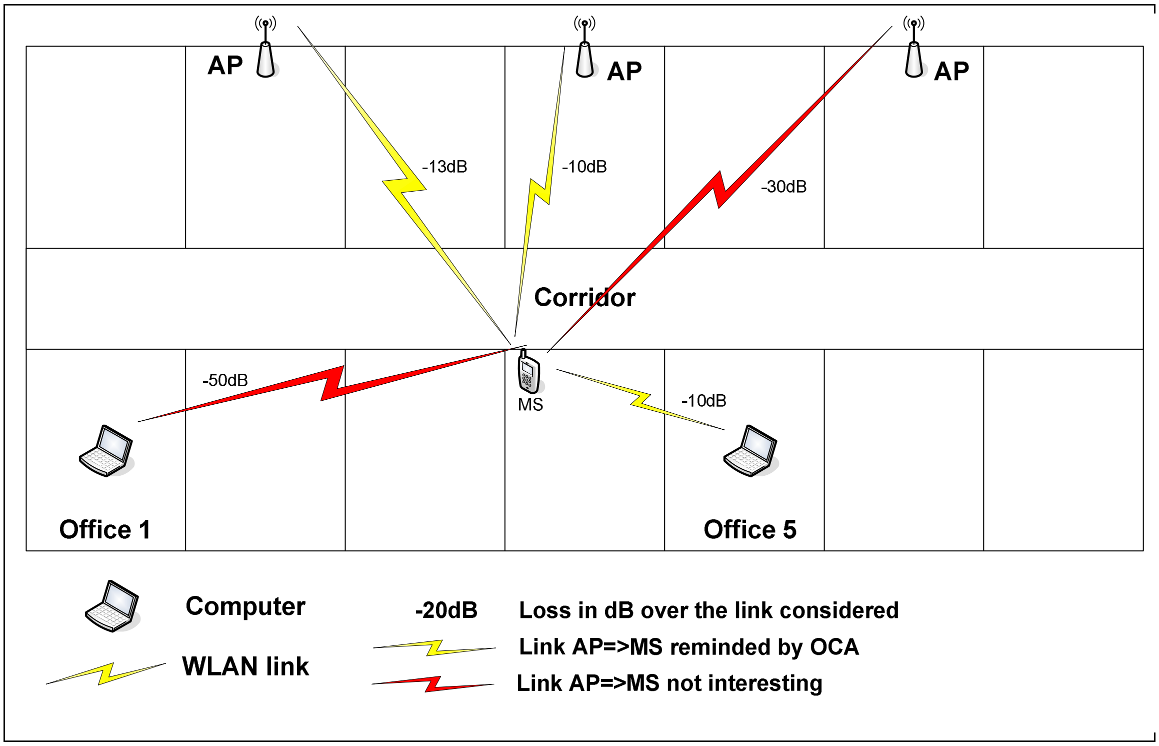

Figure 5.4.a: ICA Algorithm (RSSI comparisons)

In this Scenario what is important is to forecast that the MS considered will be in the right room. Knowing that there are 14 different offices it seems really important to localize the mobile according to the computers and the AP as well. We take the minimum RSSI value and decide to estimate the position of the MS as inside the room where the beacon node the most interested is (the one which sends the high RSSI signal to the mobile). We then choose to choose the middle of the room for the prediction.

<u>Next section: simulations and results:</u>

Next section will present the analysis of the performance of SVM first which is the main aspect of this report and which takes part in every simulation. Then we will go through CSVM in order to see the improvements done. And finally, OCA and ICA results are going to be analyzed to see their respective efficiency and understand how important their contribution is.

6. Simulation, results and discussion

6.1 Outdoor Scenarios

6.1.1. <u>**Scenario 1.A**</u>

In the Scenario, we consider a 100m-by-100m square area in which we dispatch several Access Points [2]. Otherwise than dealing with the hop-count distances our input information is the RSSI measurements from every AP to the other AP (training phase) or to the other MS.

The following diagram (Figure 6.1.1.a) presents the environment of the outdoor Scenario 1.A.

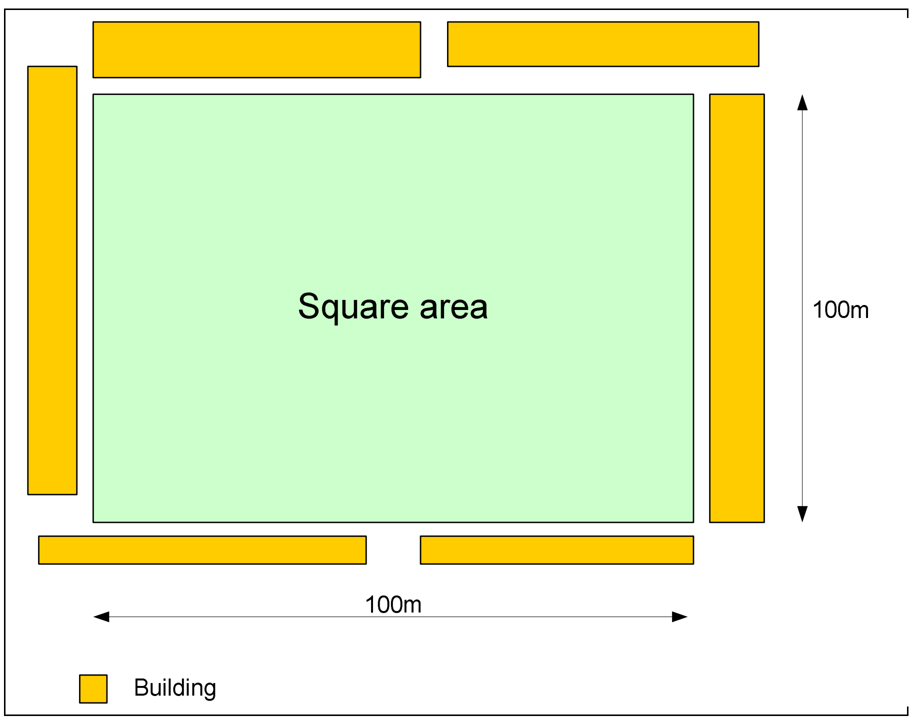

Figure 6.1.1.a: Outdoor map, Scenario environment 1.A

The area corresponds to some main areas of cities such as ones with city halls, museums (Raadhuspladsen in Copenhagen, Le Louvre in Paris). These squared places are surrounded by buildings on each of the sides. Moreover their width and length are important (around 100m). This is why this Scenario is interesting.

In order to calculate the signal strength we base our calculations on the distance between the 2 AP considered or the AP and the MS. We follow Scenario B1 presented in [9]. We make a few modifications in order to take into account only the LOS case because we assume that no building or obstacles are present in our Scenario (except in the borders).

Here is the pathloss equation:

$$\boxed{Pathloss\_(dB) = 22.7\log_{10}(d) + 41}$$

d stands for the distance between the 2 AP or the AP and the MS.

and the shadowing standard deviation:

$$\boxed{\sigma\_(dB) = 2.3}$$

<u>Randomly generated Beacon nodes</u>

We dispatch a certain amount of AP in the square area considered. In order to localize the MS accurately we decide to base our calculations on the environment described as following (Table 6.1.1.b):

Dimensions of the area (m)	x-axis	100
	y-axis	100
Number of hyperplan	x-axis	16
	y-axis	16
Number of AP generated		50
Number of simulations		20

Table 6.1.1.b: parameters of the Scenario 1.A

The Figure 6.1.1.c, below, represents the 50 AP randomly generated in the square area.

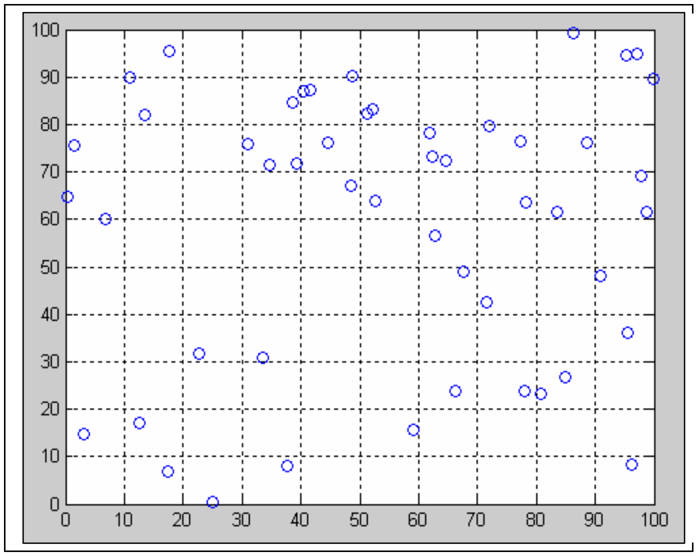

Figure 6.1.1.c: 50 AP randomly generated in a [100x100m] grid

The distance of each hyperplan is equal to $\frac{D}{M} = \frac{100}{16} = 6.25\,m$

Fixed Beacon nodes:

We decide to set the parameters of the SVM for every indoor and outdoor simulations as shown in Table 6.1.1.d

SVM type	multiclass
Kernel	RBF
Scaling interval	[0;1]
C	40
G	2

Table 6.1.1.d: SVM parameters for the Scenario 1.A

SVM type has to be multiclass (one class stands for one hyperplan) because we consider several hyperplans. The kernel function chosen is the most optimal one [6]. The SVM software choosen is libsvm [10] [11]. After the generation of the APs. We also generate the MS on the map. Figure 6.1.1.e shows an example of a complete map including AP (beacon nodes) and MS (to be localized).

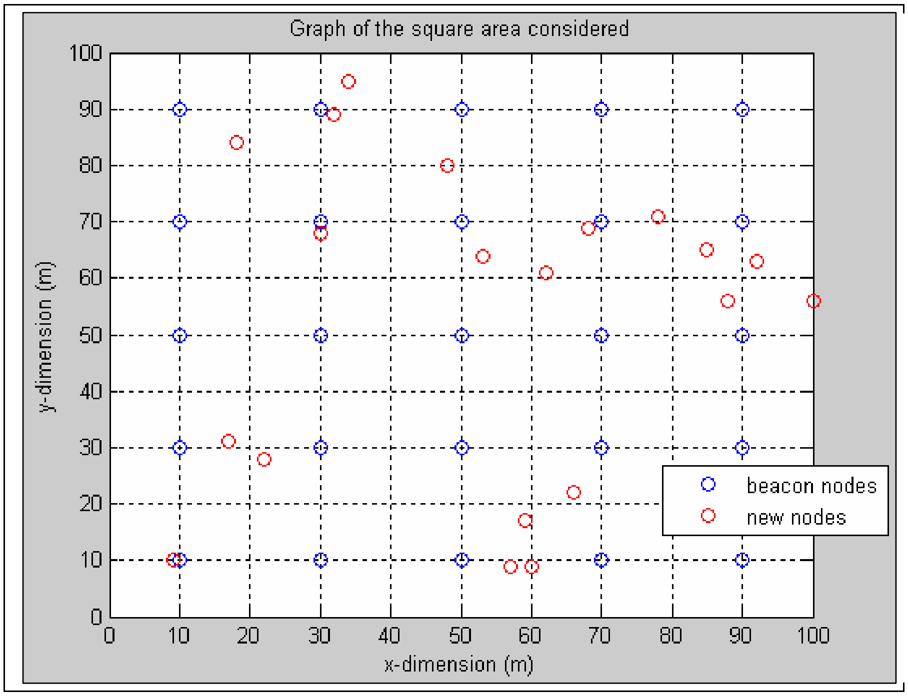

Figure 6.1.1.e: AP and MS random generations for the Scenario 1.A

In this Scenario we choose to fix the AP to a certain location in order to study the incidence regarding the localization. Namely we decide to place 50 beacon nodes such as they are fairly dispatched in the square area (see Figure 6.1.1.f)

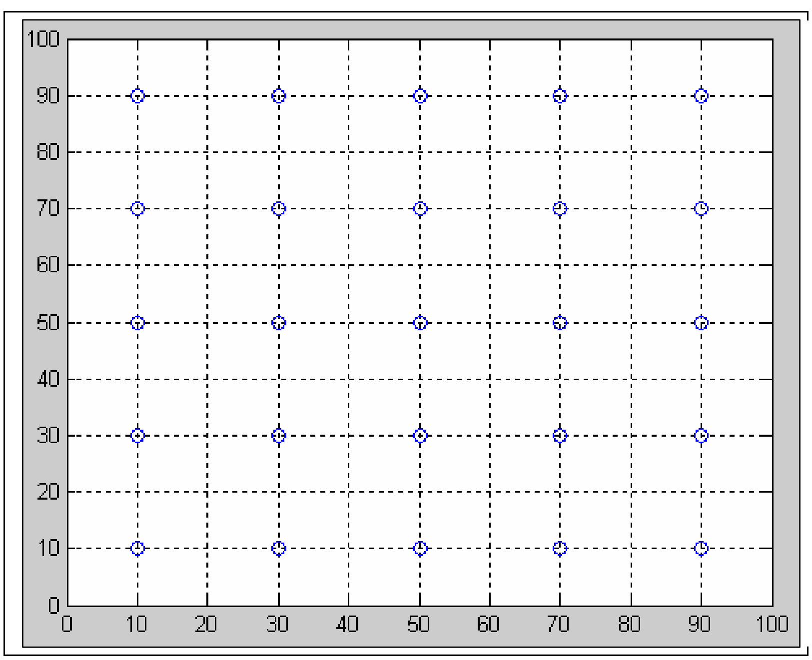

Figure 6.1.1.f: 25 AP fairly generated in the [100x100] area

6.1.2 Scenario 1.B

In this Scenario, we consider a 100m-by-50m street in which we dispatch several Access Points. Otherwise than dealing with the hop-count distances our input information is the RSSI measurements from every AP to the other AP (training phase) or to the other MS. Using this environment is important because we can find several streets in different cities which have a length of at least 100m and whose width is around 50m (shopping and main streets cities).

Figure 6.1.2.a presents the outdoor Scenario 1.B with this main street whose dimensions are 100m-by-50m.

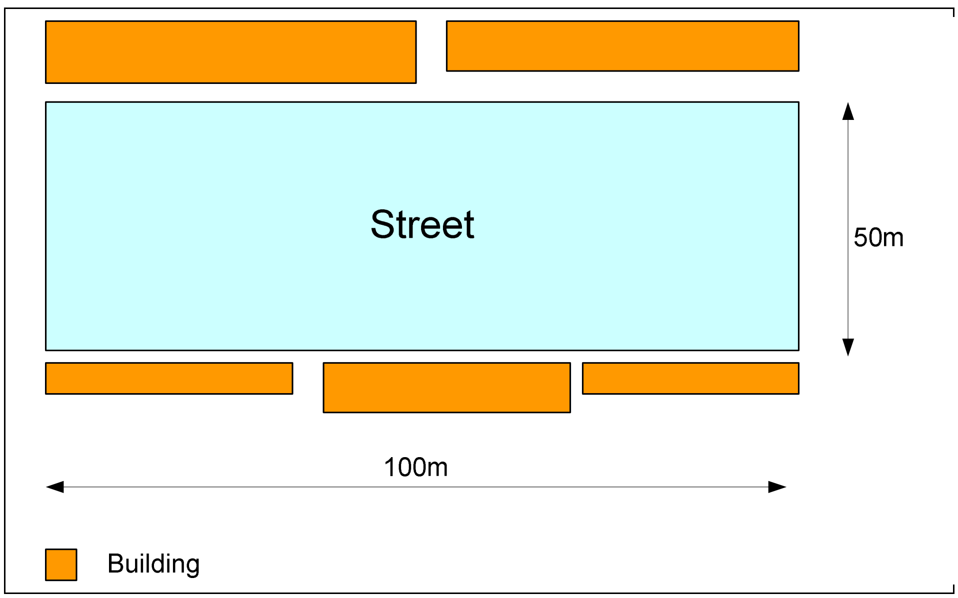

Figure 6.1.2.a: Outdoor map for the Scenario 1.B

In order to calculate the signal strength we base our calculations on the distance between the 2 AP considered or the AP and the MS. We follow the Scenario B1 presented in [9]. We also make a few modifications in order to take into account only the LOS case because no building or obstacles are present in our Scenario.

Here is the pathloss equation:

$$\boxed{Pathloss\_(dB) = 22.7\log_{10}(d) + 41}$$

d stands for the distance between the 2 AP or the AP and the MS and the shadowing standard deviation:

$$\boxed{\sigma\_(dB) = 2.3}$$

Randomly generated Beacon nodes

In this 2nd Scenario we dispatch a 25 AP in the area considered. In order to localize the MS accurately we decide to base our calculations on the environment described in Table 6.1.2.b:

Dimensions of the area (m)	x-axis	100
	y-axis	50
Number of hyperplan	x-axis	16
	y-axis	8
Number of AP generated		25
Number of simulations		20

Table 6.1.2.b: Parameters of the Scenario 1.B

We notice that we keep the number of simulations, namely 20 different ones. That means 20 different random regarding the AP. Let us see now in Figure 6.1.2.c one case in which 25 AP are generated randomly taking into account 100m of length and 50m of width for our environment.

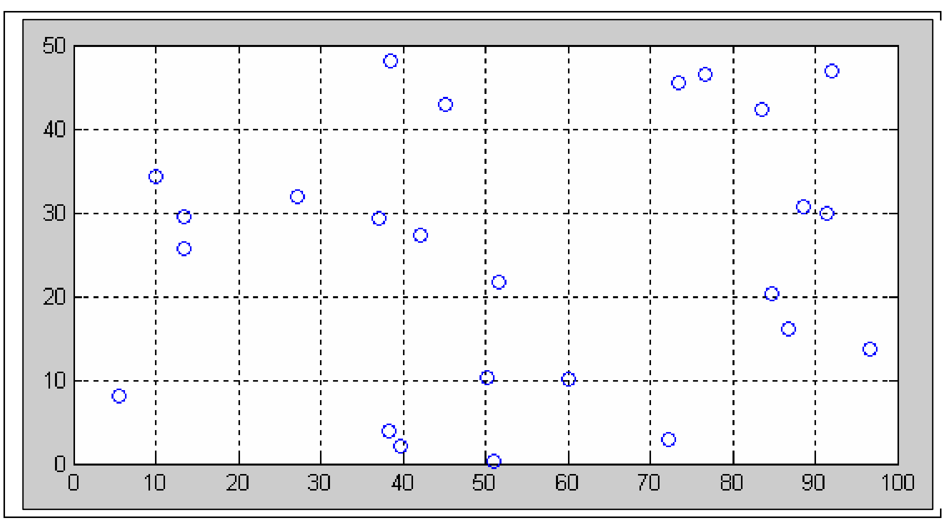

Figure 6.1.2.c: 25 AP randomly generated in a [100x50m] grid

Each hyperplan width (x-axis) is equal to $\frac{Dx}{M} = \frac{100}{16} = 6.25\,m$

Each hyperplan width (y-axis) is equal to $\frac{Dy}{M} = \frac{50}{16} = 3.725\,m$

Fixed Beacon nodes:

Here we exactly deal with the same Scenario 1.B as previously but we fixed the 14 access points such that they are fairly dispatched in the [100x50] grid.

Figure 6.1.2.d shows us the environment.

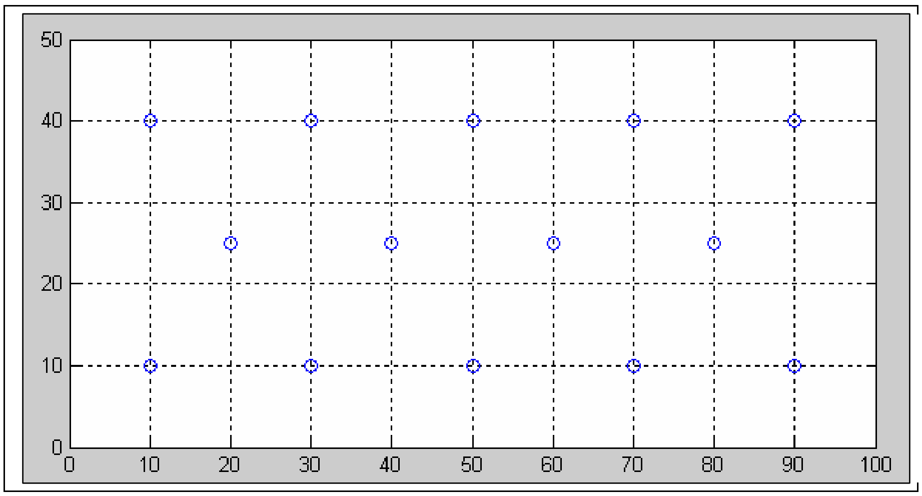

Figure 6.1.2.d: 14 AP fixed generation in a [100x50m] grid

Afterwards, 20 MSs are generated. They are going to be localized in the [100x50] m Scenario which represents the main street environment. Figure 6.1.2.e represents the Scenario.

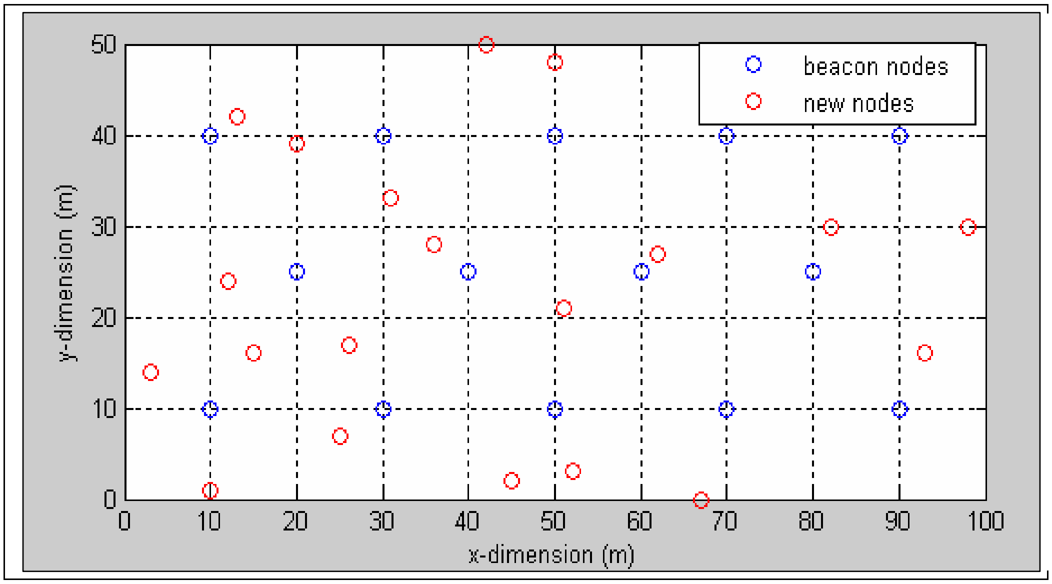

Figure 6.1.2.e: Main street area with 14 fixed APs and 20 random MSs

Relevance of these 2 Scenarios:

Hanging around in a main street of a capitol for instance permits us to focus on the different area where we can get a WiFi connection thanks to one or more access points.

The goal is to use as far as we can some networks already built in order to exploit the points to increase the localization accuracy. In a main street what is interesting is the fact that several fast foods offer now a WiFi access. Moreover, in some café, currently there is an enhancement in this sense to increase the number of clients by attracting tourists who would like to take their lunch and look through the web if they can find a show for their night or to check if a museum is open in the afternoon to avoid going there if it is closed. Then we may also use the WiFi connection from the hotels, which could be a good opportunity to decrease the cost of the AP deployment.

Finally, in several cities such as London and Paris, for example, internet providers offer to their clients a low cost fare to share their access points in order to allow users in the street with VoIP phones to communicate or browsering through the web on a patio of a restaurant to find movies performance in the surrounding cinemas. Next section describes the results and their analysis

6.1.3 Results, analysis and discussion

Scenario 1.A:

Figure 6.1.4.a presents the results for the outdoor Scenario 1.A. It compares the SVM with CSVM.

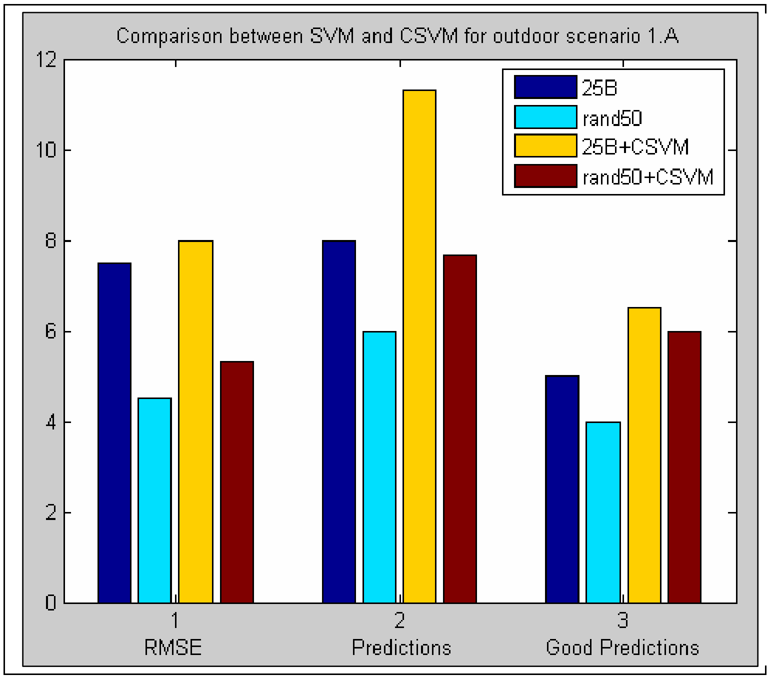

Figure 6.1.3.a: Comparison CSVM vs SVM for Scenario 1.A

Analysis:

RMSE:

The RMSE (Root Mean Square Error) is assessed thanks to the following formula:

$$\boxed{RMSE = \sqrt{(x_{est} - x_{real})^2 + (y_{est} - y_{real})^2}}$$

where x_{est} stands for the estimated position of the MS along the x-axis, y_{est} the one along the y-axis. Finally, x_{real} and y_{real} correspond to the real positions of the MS along both axis.

The RMSE for the fixed generation of AP is around 7.5m. If we compare that figure to the 4.5m standing for the 50 randomly distributed AP we note a difference of 3m. This difference can be explained because it is better to dispatch fairly the AP in the environment. Indeed the chance for the beacon nodes to get close to the MS are more important than if we consider a random dispatch where several beacon nodes can be gathered in a corner, which is not efficient. For instance using a random distribution will create empty space, namely without any beacon nodes. This will be detrimental to the MS in these empty spaces that have to be localized. The CSVM algorithm brings about a short increase around 1m because the MS estimations at the output of CSVM are based on beacon nodes (former MS) which have some errors. So the error issued from the output of the first SVM run is propagating to the 2nd SVM run which characterizes CSVM algorithm.

Predictions:
The 50 APs random distribution is not so efficient compared to the 25 APs fair dispatch in that we obtain 8 predictions for the latter and 6 for the former. This can be explained because several APs can be closed one to the other which is useful to increase the number of good predictions for a MS close to them. But if the MS is located in an empty area, SVM will not be able to predict where he is because of a lack of information regarding the RSSI measurements in this area (no AP). So some MSs far away from any beacon node cannot be predicted.
As for the CSVM algorithm it turns out that the number of prediction is soaring up (2 times more predictions for the 25 fixed beacon nodes). However CSVM is not so powerful regarding the random distribution (only 2 more predictions). So the random generation remains a handicap in the predictions.

Good predictions:
Number included in the “Predictions” the word “good” means that the MS are well localized. Namely, the distance between the estimated position and the real one does not exceed 10m. Both under SVM and CSVM the difference between the number of good predictions of the fixed AP and the randomly generated ones is not so big, around 1. Moreover, we notice 2 more good predictions for the random distribution which permits to obtain a 50% gain. So cooperation is working in that a MS well localized can work as a beacon node to find out another MS location (MS whose position was not estimated after 1st SVM procedure).

Conclusion:
Even though the difference is not so much, CSVM algorithm improves the localization in our outdoor Scenario 1.A. Over 20 MS generated it permits to obtain around one more good prediction which means 10% of increase. Regarding the difference between the fixed 25 beacon nodes and the 50 generated ones it appears that the former benefits from their fair dispatch along the grid. The localization performance is 33% better for them.

Let us compare SVM with OCA. Figure 6.1.4.b presents the results for the outdoor Scenario 1.A. It compares the SVM use with OCA ones.

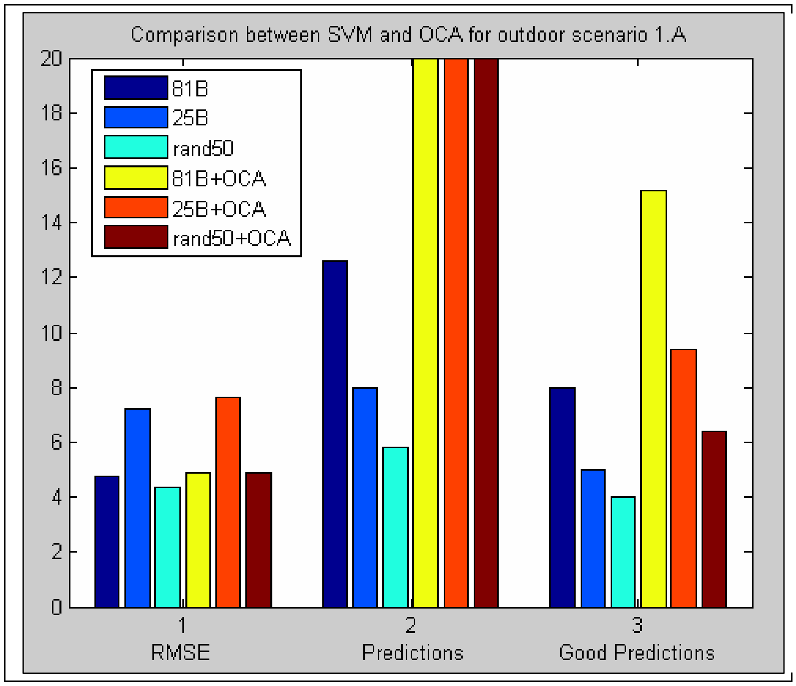

Figure 6.1.3.b: Comparison OCA vs SVM for Scenario 1.A

The difference of values of OCA and SVM are insignificant (less than 0.5m). This can be explained thanks to the fact that OCA only intervenes for the case where there is no prediction, so the predictions coming from the SVM are not modified. Moreover predictions added by OCA are good in the sense that the RMSE is not varying while the number of predictions is soaring up to reach 20. It permits us to be confident in the efficiency of OCA. First of all, thanks to OCA all the 20 MS have now an estimation of their position which is a good point compare to previously where SVM outputs a lack of information. Second and most interesting observation: for every different situation the number of good prediction is soaring up to reach 10 for the 25 fixed beacon nodes and around 7 for the random generation. The increase is about 100%. It means that OCA associated with SVM well performs by keeping the same RMSE as SVM and increasing the number of good predictions.

Conclusion:

OCA permits to obtain better performances. The number of good predictions is more important (2 times better) wile the RMSE is almost not varying. To sum-up the results for this Scenario 1.A we can order the different algorithms in term of efficiency.

1. OCA
2. CSVM
3. SVM

Scenario 1.B:

Figure 6.1.4.c presents the results for the outdoor Scenario 1.B. It compares the SVM use with CSVM.

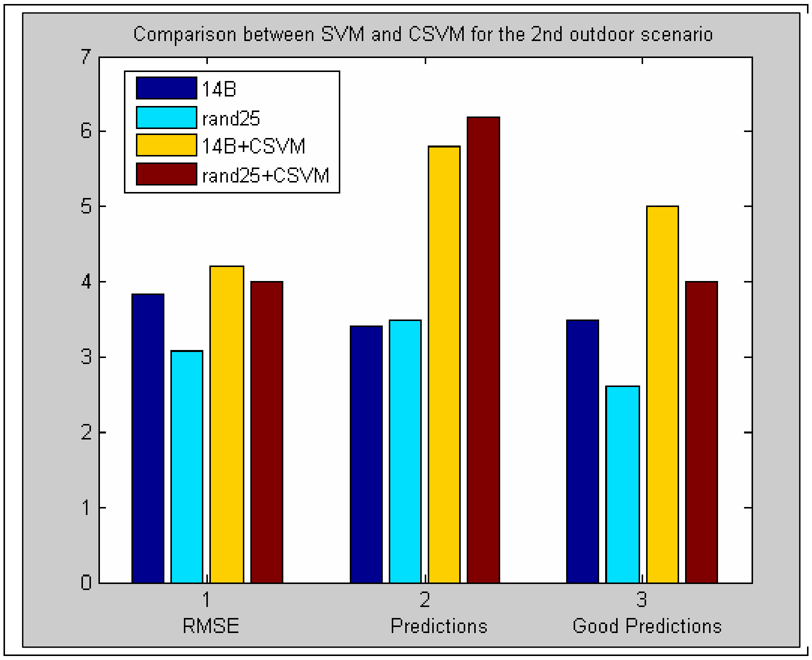

Figure 6.1.3.c: Comparison CSVM vs SVM for Scenario 1.B

Contrarily to Scenario 1.A the RMSE for the fixed AP is higher than the one which characterizes the 25 randomly distributed AP. This gap is due to the difference in figure (14 against 25). Moreover we took a very small number of fixed beacon nodes (14) compared to the size of the area considered. This explains why the RMSE is higher is that case compared to the random distribution. CSVM algorithm increases the RMSE which remains really small (4m at the maximum). Moreover it permits to get a more important number of predictions (6 otherwise than 4 for the simple SVM), indeed we state a strong increase (70%) between the SVM and CSVM. However the increase concerning the predictions does not bring so many results regarding the well predicted MS. Indeed we can notice only one more good prediction, which seems a low improvement. Regarding the comparison between the random and fair distribution the better increase of predictions for the former is inefficient for the good predictions statistics. Namely the fair distribution, despite its lower amount of predictions obtains more MS well predicted than the random distributed case. The logic is respected in that the fixed distribution outperforms the random one.

Conclusion:

The RMSE, issued from CSVM algorithm stays at a very low value, namely 3 to 4m while CSVM permits to gain 66% of good predictions. Moreover the highest amount of good predictions for the random distribution does not have the expected impact on the good prediction statistics. As we did before for the Scenario 1.A we are going now to analyze the OCA results.

The Figure 6.1.4.d presents the results for the outdoor Scenario 1.B. It compares the SVM use with OCA ones.

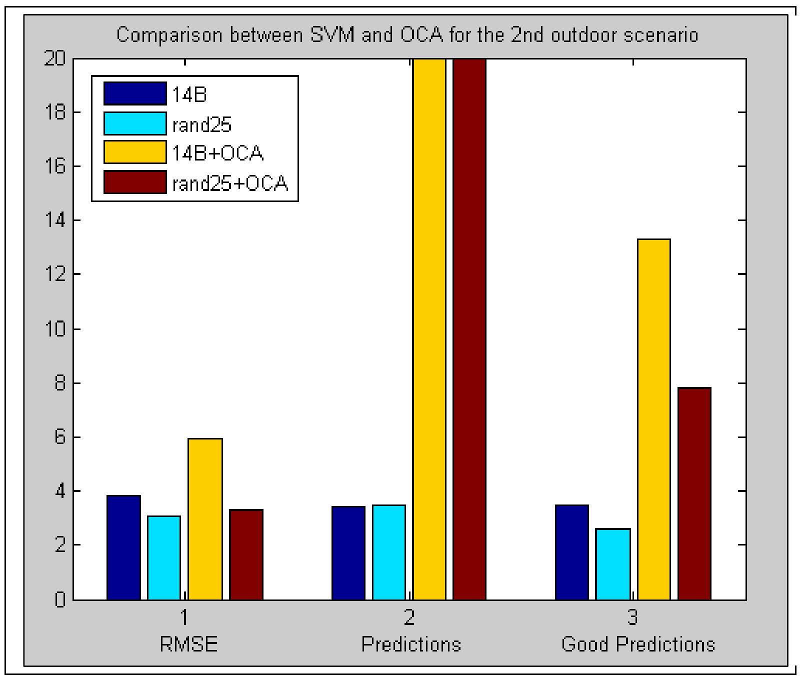

Figure 6.1.3.d: Comparison OCA vs SVM for Scenario 1.B

The RMSE value is changing for the fixed AP case passing from 4m to almost 6m, otherwise for the random distribution it is still the same. The increase which is not present in the Scenario 1.A can be explained by the little amount of beacon nodes in this Scenario. As we observed in the previous Scenario all the MS have now a prediction regarding their respective location. An important result: almost 14 good predictions and 8, respectively for the fixed and random AP generations. The number of well predicted MS soars up with both beacon nodes distributions. We also notice the highest figure for the fixed one. This explanation is the following: even if there are just 14 beacon nodes, the fair distribution permits to at at least a little amount of them to get close to the MS. Being surrounded by AP is important to be well localized by SVM. It means that OCA is efficient in this Scenario.

Conclusion:

OCA is more time better that SVM and the number of good prediction which soars up let us think that the former is a good additional algorithm to SVM. Moreover, even if CSVM is not so performing compared to OCA it remains a nice alternative. First of all to conclude this Scenario 1.B we have noticed that the ranking is not modified in that we obtained better results regarding the number of good predictions with OCA than CSVM. The latter is still interesting compare to the simple use of SVM. The RMSE remains interesting in both of the Scenarios. Second, in both of the outdoor Scenarios we observed an important advantage of using OCA and an appreciable result regarding CSVM algorithm. The next study will be about the indoor environment.

6.2 Indoor Scenarios

6.2.1. Environment considered

In this Scenario we consider a floor with 14 different offices which dimensions are 10 x 10m. The rooms are separated by a 70 x 5m corridor. Figure 6.2.1.a reminds the environment considered.

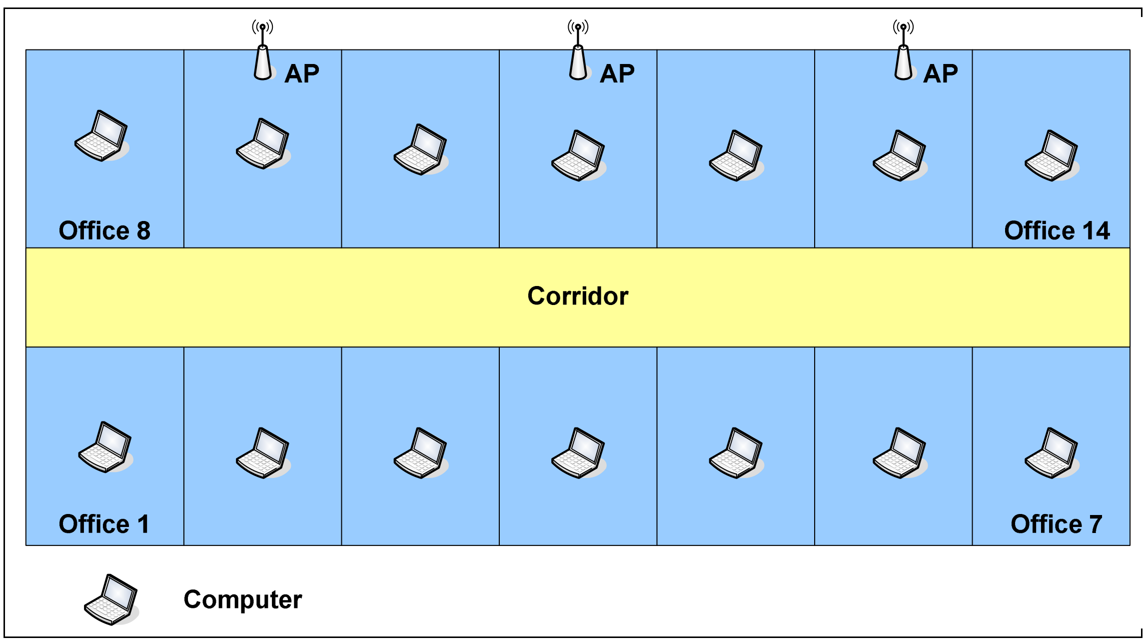

Figure 6.2.1.a: Indoor Scenario with APs and computers

A MS is going to receive 17 different signals (3 from the AP and 14 from the computers).

The aim is to estimate in which room the MS is.

Pathloss model:

We based our Scenario on Scenario A1 in [9]. So, we took into account the same pathloss formulas, which are:

LOS case:

$$\boxed{pathloss = 18.7\log_{10}(\mathrm{d}) + 46.8}$$

NLOS case:

$$\boxed{pathloss = 36.8\log_{10}(\mathrm{d}) + 38.8}$$

Respective shadowing standard deviations for both of the cases:

LOS case:

$$\boxed{\sigma\_(dB) = 3.1}$$

NLOS case:

$$\boxed{\sigma\_(dB) = 3.5}$$

In order to decide if we are in LOS or not we focus on the distance *d* between the 2 AP or the AP and the MS. Namely, if the distance $d \leq 2.5$m then we estimate that we are in LOS. On the contrary, if $d > 2.5$m then we use the following equation:

$$\boxed{P = 1 - 0.9 \times (1 - (1.24 - 0.61 \times \log_{10}(d))^3)^{1/3}}$$

The output of this equation gives the probability to be in LOS. So we generate a uniform distributed value which ranges [0;1] and compare the latter with the value of *P* obtained thanks to the distance *d*. If the former exceeds the latter we are in NLOS, otherwise we use the LOS pathloss equation for our calculations and also a standard deviation equals to 2.3dB.

Pertinence of the Scenario:

Nowadays so many companies' building are equipped with a wireless internet access. We took a map which already exists [9]. Moreover we decide to add one computer (which could be a desktop or a laptop of course) which is going to work as an AP. For instance in Aalborg University there is at least one computer in each room or office. The desktops cannot be easily removed and we can consider that a laptop is put on a table and that this table is not going to be moved everyday. So this Scenario is relevant in that we use the network configuration without adding any new object. We enter into the floor with our mobile and we are detected. In Sections 6.2.2 and 6.2.3 two different environment are considered for the simulations: only the 3 AP as beacon nodes and in a second time the same AP with one computer located in each office.

6.2.2. Scenario 2.A: Access Points and Mobile Stations

For the first scenario, which is called Scenario 2.A, we deal with 7 hyperplans along the x-coordinates which stands for the 7 rooms on both of the sides of the floor. Regarding the y-axis we decide to divide it into 3 different hyperplans which correspond to the 2 set of 7 rooms and the 2^{nd} one to the corridor.

Table 6.2.2.a summarizes the features of this Scenario.

Dimensions of the area (m)	x-axis	70
	y-axis	25
Number of hyperplan	x-axis	7
	y-axis	3
Number of fixed AP generated		3
Number of simulations		20

Table 6.2.2.a: Features of the indoor Scenario 2.A

The environment, the AP and the 20 MS can (only the 20 MS are generated randomly) be distributed such as Figure 6.2.2.b shows:

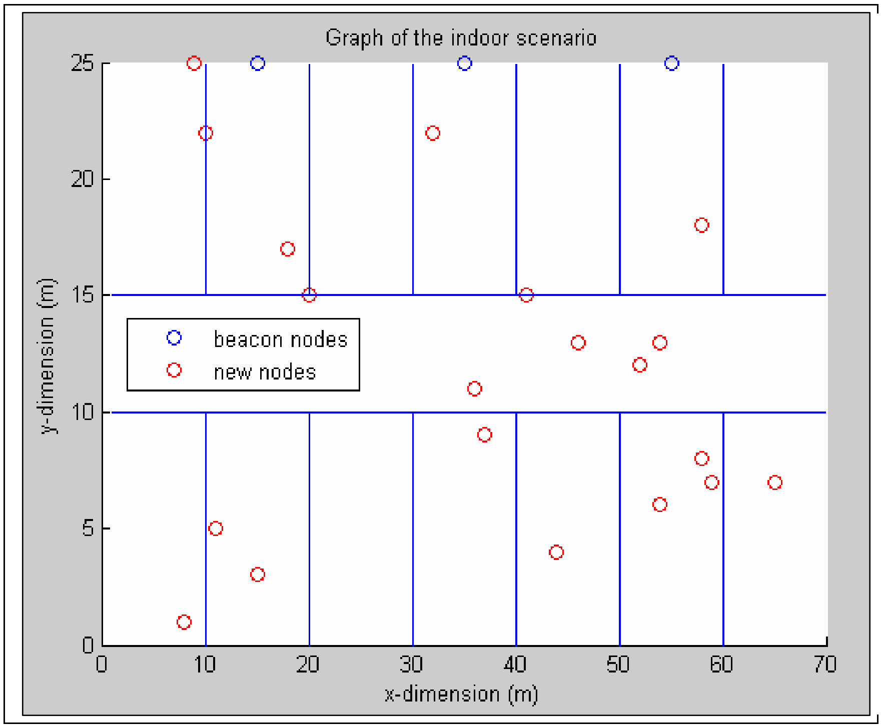

Figure 6.2.2.b: Map of the Scenario 2.A

The results are placed in Section 6.2.4 of the report called "Results and discussion".

6.2.3. Scenario 2.B: Access Points, computers and Mobile Stations

In this Scenario 2.B the number of hyperplans along both of the axes remains the same. The sizes of the rooms and the corridor are unchanged. The only difference results in the add of 14 computers (AP). One in each office.

Table 6.2.3.a sums-up the feature of this Scenario.

Dimensions of the area (m)	x-axis	70
	y-axis	25
Number of hyperplan	x-axis	7
	y-axis	3
Number of AP generated		3+14
Number of simulations		20

Table 6.2.3.a: features of the indoor Scenario 2.B

The diagram following (Figure 6.2.3.b) presents the environment with the 3AP, the 14 computers and the randomly generated 20 MS.

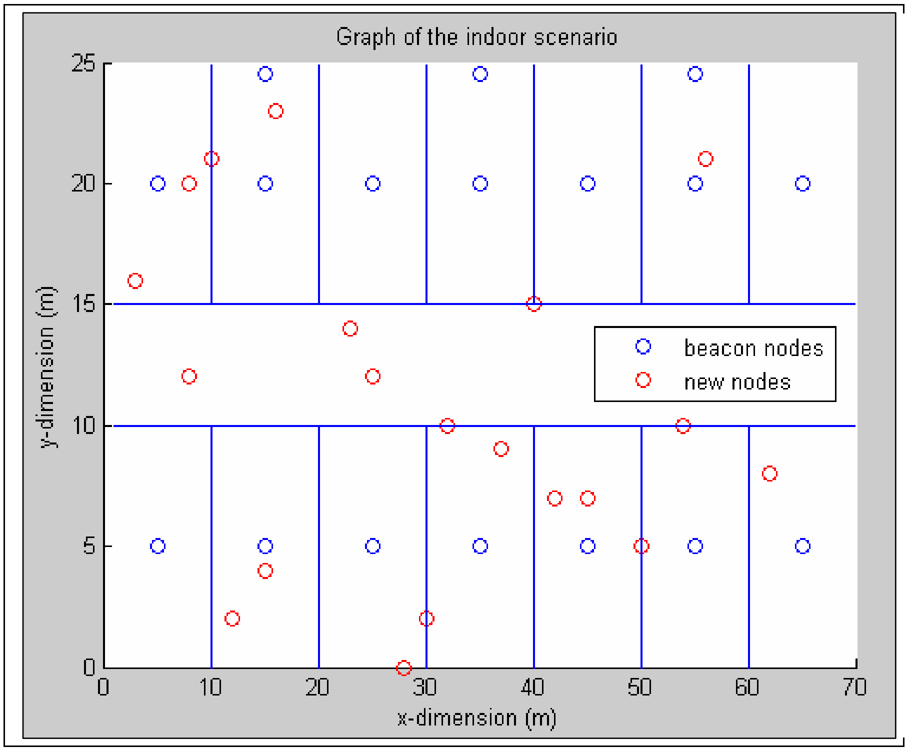

Figure 6.2.3.b: map of the Scenario 2.B

As in the previous section the results are gathered in the following section 6.2.4 called “Results and discussion”.

6.2.4. Results, analysis and discussion

We are going in this section to compare SVM and CSVM algorithms, SVM and ICA, and finally CSVM and ICA for the Scenarios 2.A and 2.B. Figures 6.2.4.a and 6.2.4.b gather all the results. Figure 6.2.4.a compares SVM and CSVM algorithm with Scenario 2.A and 2.B.

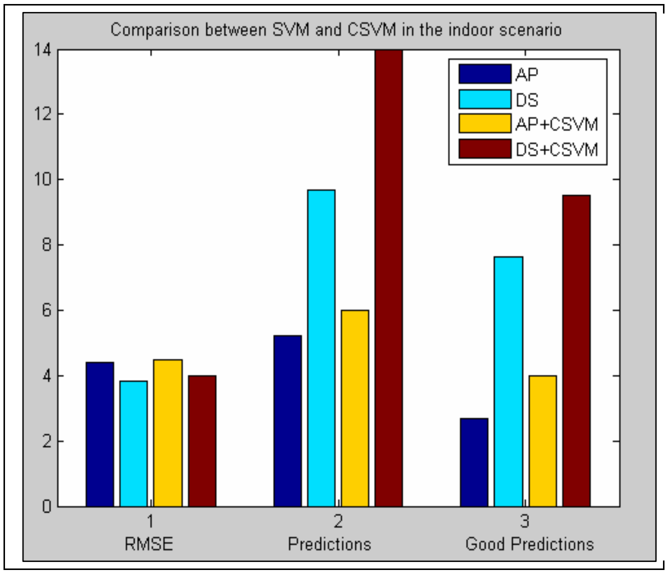

Figure 6.2.4.a: SVM and CSVM results for the indoor Scenarios

Analysis:

The RMSE is not varying so much and stays around 4m (we decided to take the middle of the office as a prediction, this is the reason why there is no change). Regarding the number of predictions for the case with the computers and without them we notice a gap of 4.5. Namely when the computers are present in the rooms SVM outputs more estimated locations which sounds logical. But the fact that the RMSE is not varying let us think that CSVM performs well. Indeed it permits us to pass from 6 to 14 predictions so more than 2 times of increase. It is important to notice that the computers lead to 8 MS localized in the good room compared to the 3 MS well localized thanks to the only 3 AP. The difference is consequent and remains important for the CSVM case. Concerning the comparison between SVM and CSVM an improvement can be observed specially for the AP+computers environment which permits to reach almost 10 good predictions over 20 MS. With or without CSVM we have the same statement which is coherent: the 14 desktops or laptops are a benefit for our localization procedure. Indeed SVM outperforms under that constraints compared to when we only use 3 AP.

Conclusion:

CSVM outperforms SVM while the 3 AP are not sufficient to localize the 20 MS. This is why we need at least one laptop or desktop in each room and then the result becomes interesting. Except an average of 4-5 MS which are positioned inside a wall or a door which can cause problems for the simulations the results are important to show that with CSVM we get better predictions. So cooperation is brings better results in this indoor environment than in the previous scenarios 1.A and 1.B.

Let us focus now on the efficiency of ICA algorithm. Figure 6.2.4.b compares the SVM and CSVM algorithm with Scenario 2.A and 2.B.

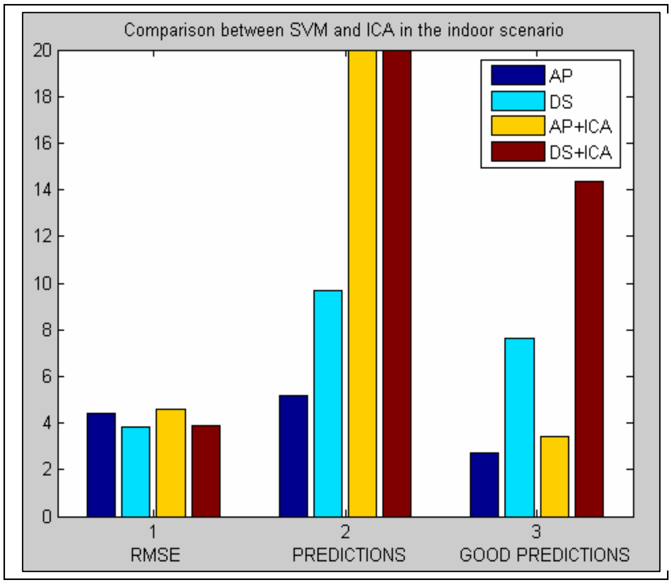

Figure 6.2.4.b: SVM and ICA results for the indoor Scenarios

Analysis:

As in Figure 6.2.4.a the RMSE is not varying for the same reasons as explained previously. Regarding the predictions without ICA, we note the benefit brought by the computers. It results in an enhancement of 2 times the value (10 predictions compared to 5) obtained for the case where there are just the 3 APs. In fact the APs are just useful if the MSs are in their respective office. Otherwise SVM will not be able to predict any positions. ICA carries out a 100% prediction regarding the 20 MS randomly generated in the environment as in Scenario 2.A and as the OCA algorithm did in Scenarios 1.A and 1.B. Adding the computers leads to obtain 8 good predictions (great improvement compared to the 3 good predictions we get without ICA). Even if ICA cannot really increase the localization with the 3AP because only a little number of MS is in the AP's room it outperforms CSVM and SVM in outputting 14 good predictions for the AP + computers case. Namely we reach 70 % of good predictions with ICA.

Conclusion:

ICA outperforms CSVM in both indoor scenarios in that we observed an important increase of good predictions while the RMSE is not varying. Moreover an average of 4-5 MS is generated in unfair location (inside walls or doors) so the results are subject to one more improvement. As for CSVM we notice its efficiency in both of the indoor Scenarios, efficiency less visible in Scenario 1.A and not really demonstrated by the results obtained in Scenario 1.B.

7. Conclusion and future work

7.1 Conclusion regarding this project

We applied SVM in a localization procedure to find MS station positions. We based our calculations on RSSI measurements issued from AP links fairly or randomly distributed in both outdoor and indoor scenarios. The parameters of the SVM were chosen in order to fit with the different environments considered along our simulations. We expect to find out relevant outputs for each scenario but the impact of the models chosen, and severe constraints reduced the amount of predictions. This is why other algorithms have been implemented to deal with this lack of estimations. It has permitted us to see that CSVM, OCA and ICA have improved localization in both of the environment. So cooperation was a good additional process to SVM which carried better estimations regarding the location of MS. ICA and OCA outperformed CSVM. But the latter improved SVM estimations.

Good advantages of SVM such as

- Training phase to run once and only one
- A possible adaptation to every Scenario
- A short calculation time (maximum of 1min)

were really helpful and demonstrate to us the reliability and the efficiency of Support Vector Machines.

7.2 Expectation concerning future work

Our work contribution is limited to a set of Scenarios but we can adapt the required distances thanks to our simulation code which is dedicated to deal with different kind of parameters (frequency of the transmission, length and width of the area considered, number of AP and MS, their generation model).

<u>1st possible improvement (tracking): Kalman filter for the 1st Scenario:</u>

Regarding cooperation for the first Scenario we can improve the simulation taking into account NLOS links between MS and the AP. In fact to solve this transmission problem we predict first the location of the MS and then thanks to a Kalman filter we estimate the position of the 2nd MS in NLOS with the AP putting it a lower weight compared to the AP.

Figure 7.a describes this new scenario of cooperation.

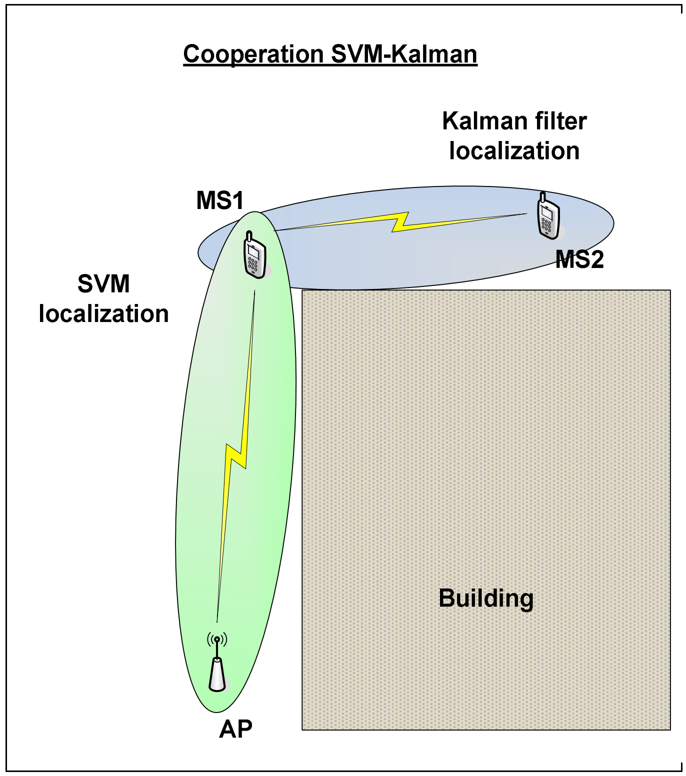

Figure 7.a: SVM and Kalman filter cooperation in localization

2nd possible improvement:practical part

It will be interesting to analyze in a real environment the behavior of SVM, CSVM, OCA and ICA algorithms. For instance the indoor environment stands for a floor in many university or company buildings. After scaling the hyperplans to the environment and create the infrastructure and ad-hoc links to measure the RSSI we can run the calculations.

3rd possible improvement:Compare our result with other algorithms

Compare the performance of SVM with the Non-Linearleast Square Algorithm and the Kalman filter should be done in order to compare their respective efficiency under the same environment.

Bibliography

[1] Vineet Abhishek, Localization in Ad-Hoc Sensor Network: A Machine Learning Based Approach, CS229 Project Report, Fall 2005

[2] Thinh Nguyen, *Localization In Wireless Sensor Networks using SVM*, 2005

[3] Yi Liu and Yuan F. Zheng, *Soft SVM and Its Application to Video Object Extraction,* 2005

[4] Boriana Milenova, Joseph Yarmus, Marcos Campos, *SVM in Oracle Database 10g: Removing the Barriers to Widespread Adoption of Support Vector Machines*, 02/08/2000

[5] Steve R. Gunn, *Support Vector Machines for Classification and Regression, p 21-23, 10th May 1998.*

[6] Vapnik, B. Scholkopf, K. Sung, *Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers, Dec 1996*

[7] Olvi L. Mangasarian, David R. Musicant, *Lagrangian Support Vector Machine,* http://www.cs.wisc.edu/dmi/lsvm/

[8] John C. Platt, Sequential *Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*, p2-4, April 21, 1998

[9] Daniel S. Baum, Hassan El-Sallabi, Tommi Jämsä, Juha Meinilä, Pekka Kyösti, Xiongwen Zhao, Daniela Laselva, Jukka-Pekka Nuutinen, Lassi, Hentilä, Pertti Vainikainen, Jarmo ivinen, Lasse Vuokko, Per, Zetterberg, Mats Bengtsson, Kai Yu, Niklas Jaldén, Terhi Rautiainen, Kimmo Kalliola, Marko Milojevic, Christian Schneider, Jan Hansen, IST-2003-507581 WINNER, D5.4 v. 1.4, *Final Report on Link Level and System Level Channel Models*, p 14-27. Nov. 18th, 2005

[10] Chih-Chung Chang and Chih-Jen Lin, *LIBSVM: a Library for Support Vector Machines,* 3/01/2006

[11] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, *A Practical Guide to Support Vector Classification*, *2002*