A Scalable Localization System for Critical Controlled Wireless Sensor Networks

Thanh-Dien Tran, José Oliveira, Jorge Sá Silva, Vasco Pereira, Nuno Sousa, Duarte Raposo, Francisco Cardoso

Abstract—**Determining the positions of unknown position nodes, especially mobile nodes in a wireless sensor network (WSN), is critical for many applications. It helps to identify the location of the collected data and of the node carrier such as a worker, patient or vehicle. This information is often critical on supporting the right (time) decisions. This paper presents a scalable localization system targeting Controlled WSNs for critical industrial environments. Multiple positioning methods were implemented and evaluated using real testbeds setting up in both laboratory and industrial environments. The measurement used in our localization system is Received Signal Strength Indicator (RSSI). Although it is unstable and with high variance, the experimental results show that pattern matching based methods such as k-nearest neighbors, probability-based (Bayesian Theorem) and Kalman filter over probability-based produce an acceptable accuracy that is sufficient for many applications. In particular, the average distance error of 3.37m can be achieved with 50th and 80th percentile distance errors of 2 and 5.35m respectively. In addition, by carefully designing the positions of beacons it is possible to obtain the average distance error about 2.23m and 50th and 80th percentile distance errors of 0.46 and less than 4.4m respectively.**

Index Terms — **Localization, Wireless Sensor Network, Baysian method, K-Nearest Neighbors**

I. Introduction

With the ability to wirelessly connect a large number of tiny and low cost nodes, Wireless Sensor Networks (WSNs) have an unlimited potential for useful applications. Nodes in a WSN can sense the conditions of the environment (e.g., temperature, humidity, pressure, radiation, PH, etc.), do some simple computation tasks and support wireless communications. Naturally, WSNs can be applied in most of areas including military, industrial, health-care, environment and home monitoring. However, due to the constraints on the form, cost and power consumption, the sensor nodes have small memory, low computing power and limited communication facilities. Consequently, there are many problems related to the application development, deployment and management of this type of networks. Among of them, localization is critical for many kinds of applications because data are meaningless without knowing the location in which it was obtained. The position of the objects involved in the events is essential to many important and useful applications such as searching, rescue, tracking or people health-care (e.g., workers in hazard environments, monitoring Alzheimer).

Localization algorithms estimate the position of unknown objects (i.e., sensor nodes) by using inter-sensor measurements such as the signal strength, the angle, or the propagation time. The localization problem has been studied since 1960s resulting in the most successful location system that is widely in use today, the Global Positioning System(GPS) [1]. GPS based solutions are a good choice for outdoors. However, because of the constraints of sensor nodes, localization in WSNs using GPS is inefficient in most of the cases. Firstly, cost and energy consumption constraints prevent equipping GPS sensors for every node. In addition, it is no longer an option for indoor environments or environments with a lot of obstacles such as industrial plants. Moreover, the accuracy of civil GPS, in some cases, does not satisfy the requirements of the applications.

Although the work in [2] showed that it is possible to embed a GPS receiver to sensor nodes and to reduce energy consumption by offloading GPS processing to the cloud, the accuracy archived was still low (35m). More importantly, in industrial plants, GPS based solution will find limitations, when a person goes into a process area, most of the cases erected in metal and concrete. In such scenarios, GPS is no longer an option, and WSN based solutions get their opportunity to operate.

In this paper we present a scalable Real Time Localization System (RTLS) that uses the RSSI values as the measurement. The contributions of this paper are innovative and include the propositions that follow.

First, we propose a scalable and real time localization system. This is achieved by dividing a large sensor network into multiple subzones to improve the performance of location computation.

Second, the location system consists of multiple localization algorithms. We also combined Kalman filter [3] with a probability-based method in order to improve the accuracy.

Third, the location system was evaluated with the testbeds in different environments, especially a real critical industrial environment, the Soporcel paper plant of Portugal.

Fourth, we introduce a method for organizing the training dataset to deal with the large-scale sensor network, while maintaining the performance of location estimate process.

With a testbed that includes 12 readers (fixed position nodes) distributed over 2 floors, we achieved 50th and 80th percentile distance errors of 2m and 5.35m respectively. In addition, in a real industrial environment with 11 readers distributed over 3 floors in a 18 floor building at the

Manuscript received August 21, 2014.

T.D.Tran, J. Sá Silva and V. Pereira are with the Department of Informatics Engineering, University of Coimbra Polo II - Pinhal de Marrocos, 3030-290 Coimbra, PORTUGAL (corresponding author to provide phone: +351911971788; e-mail: than@dei.uc.pt).

J. Oliveira and F. Cardoso are with the Department of Physics, Universityof Coimbra, RuaLargaP-3004516 CoimbraPortugal. J. Oliveira also with Eneida Wireless Solutions. (e-mail: joliveira@eneida.pt).

C. Teixeira is the CEO of Eneida Wireless Solutions, Ed. IPN, Rua Pedro Nunes, 3030-199 Coimbra, PORTUGAL

Soporcel paper mill, we achieve 50th and 80th percentile of overall distance errors of 1.46 and 8.87m, correspondingly. Moreover, with carefully designing and implementing the positions of readers we can, in the critical industrial environment, attain 50th and 80th percentile distance errors of 0.46m and 4.4m respectively. The results from the experiments presented in this paper show that the proposed and implemented RTLS is appropriate for applying in the real critical environments.

The paper is organized as follows. The next section discusses the related work. Section III presents the general application scenarios. Section IV describes the proposed scalable localization system. Section V presents the testbeds and experimental results. The final section presents the conclusions and future works.

II. RELATED WORK

The localization in WSNs can be divided into two broad classes: one for ad-hoc and the other for controlled networks. The former is based on the assumption that WSNs are intended to be deployed randomly into the field and nodes are rarely moved after deployment excepting nodes die or randomly added. The latter is applied for WSNs that are carefully designed. A controlled WSN comprises two main parts: the infrastructure, which is a set of nodes with fixed known positions called anchors or beacons, and the mobile nodes, which are usually attached to workers, patients, or vehicles. Determining the locations of the mobile nodes in a controlled network is essential for many types of applications. This section discusses the methods for determining location of the mobile sensor nodes.

Perhaps the simplest localization method to estimate the position of a mobile node is proximity which based on "closest beacon" principle. This means that the location of the mobile node is that of the closest beacon based on some measurement such as Received Signal Strength Indication (RSSI) or distance. Active Badge [4] is one of the positioning systems that employed this method. Another simple localization method is the centroid algorithm [5] in which the position of a node is computed as the arithmetic mean of the coordinates of the beacons in range of it. This means that if a unknown node receives signals from n beacons located at positions $\{X_i=(x_i,y_i),\ i=1,..,n\}$, then its position is computed using the formula: $X = \frac{1}{n}\sum X_i$. The advantage of these two methods is that it is very simple to implement. However, they only produce relative locations such as in which room the mobile node is.

Another approach for localization is to express the problem as a system of ***n*** equations (e.g., circles, spheres, or hyperbola) and solving the equation system to find the position of unknown position node. However, because of the errors in measurements, the system usually doesnot have a solution. Consequently, instead of solving the equation system, the linear or non-linear least square method [6] [7] is used. Depending on the type of measurement the methods in this approach are divided into two groups: lateration and angulation. The lateration method uses the distance measurement (e.g., Time of Flight or Time of Arrival - ToA, Round-trip Time of Flight or RSSI) between a device and several anchors to model the problem as a system of circular (2D) or sphere (3D) equations. On the other hand, the angulation uses the Angle of Arrival (AoA) of the transmitted signal at several anchors to form a system of lines. The most well-known public localization service employing lateration algorithm is GPS [1]. The recent work in [2] is a GPS-based solution for WSNs which tries to reduce energy consumption for sensor nodes by only getting the raw signals from satellites and then forwarding them to the location service responsible for the offline computation of the location. However, the accuracy of [2] was still low (35m). Bat [8] and Cricket [9] are the example of the multi-lateration based location system for WSNs. The accuracy of two systems is in the range of centimeters. However, the use of a combination of ultrasound and RF, make these systems more costly. In addition, it is very susceptible to noise, obstacles, and non-LOS (Line-Of-Sight) environments. Ubisense system [10] is a location system which combined AoA and UWB (Ultra-Wideband) to estimate the location in WLAN. Its accuracy was very high (cm). However, its cost is high because it requires directional antennae. In addition, its accuracy was also severely affected by noise, obstacles, and non-LOS environments.

Recently, algorithms in machine learning field such as KNN (K-nearest neighbors), Probability-based and neural networks are employed for localization problem in WSNs. The localization algorithms in this group are also known as pattern matching, fingerprinting, or scene analysis. KNN [11] can be used to estimate the location of an unknown position node by first searching the fingerprint dataset for k records that close match with the new observed data, according to a distance function (e.g., Euclidean distance). By averaging these k positions, an estimated position of the mobile object is obtained. The distances can be used as the weights when averaging k positions; and in this case it is called weighted KNN. RADAR [12], MoteTrack [13] and LANDMARC [14] are examples of localization systems that employ the KNN. For the RADAR system, it is possible to achieve 90thpercentile of location error around 5.97m. MoteTrack [13] achieves an impressive accuracy with the 80th percentile accuracy of 1.6m with data from 16 channels and about 3.5m with data from one channel. However, MoteTrack [13] system requires the sensor nodes to wait a rather long time in order to acquire enough data for computing their location. In addition, because beacons have to continue to broadcast messages on multiple channels and power levels, its energy consumption is very high. Furthermore, because the algorithm is implemented on the sensor nodes, it cannot apply for the large sensor network with hundreds or thousands of anchors. Differing from other methods, LANDMARC [14] employs a set of reference tags for collecting data. With a testbed having 2m x 1m reference tags, it can achieve the 50th percentile of location error of about 1m and 75th percentile of about 1.2m. The problem with this method is that it requires a high density of reference tags, about one tag in each two meters, to obtain good accuracy. Consequently, the cost of this system is high and it is not appropriate for use in many real environments.

Probability-based localization method is similar to KNN but it uses the likelihood to compute the "similarity" between the new observed data and the records in the training dataset. Horus [15] [16] was a probability-based system for WLAN with the 90^{th} percentile of errors of about 1.43 m. However, in order to obtain this accuracy the density of Access Points (APs) is high (average 6 access point per location). In addition, it required multiple samples for each location estimate. [17] is another probability-based system for WLAN. The results from this study showed that using the RSSI for estimating location has an expected bound for performance with a median localization error of 3m and 97^{th} percentile of 9.15m [17].

Neural networks can also be used for solving the localization problem in WSNs as research done in [18], [19] and [20]. The study in [19] used the Cricket system [9] in its experiments and showed that the performance of the neural network based solutions were better than those of the Kalman Filter [21]. However, the testbeds in these experiments were too small to evaluate the performance of the neural networks. On the other hand, the experiments of the study in [20] were only done using simulation, whose results are rarely trusted for evaluating the accuracy localization. A more realistic study in [22] showed that the 90^{th} percentile of accuracy of a neural network for WLAN was 5.4m.

Every solution has its own merits and drawbacks, making it appropriate for different applications' requirements. The proximity and centroid methods are easy to implement but their accuracy is low. Similarly, lateration and angulation have a low memory and computation requirements but the accuracy is dependent on the accuracy and reliability of the measurements, which is very difficult to achieve in WSNs. Consequently, this approach usually gives a very low accuracy in WSNs. On the other hand, the machine learning based methods are complex in implementation and require high memory and computation demand. However, they are more accurate than the others.

III. THE APPLICATION SCENARIOS

This section presents the application scenarios to which the localization system is targeted. In critical environments such as refinery and industrial plants, determining the position of people (e.g., workers) or moving devices (e.g., RFID tags attached to bicycles) is very important in order to have instantaneously appropriate responses. In such environments, the sensor nodes are attached to the humans or vehicles, to monitor people's current health situation (e.g., heart rate or blood pressure) or the environmental conditions (e.g., temperature, humidity or air pressure), as well as their positions. By experiential working critical industrial environments such as oil refinery, paper mill and chemical plant, we found that these environments have a lot of obstacles. Fig. 1, a part of a refinery, is an example of such an environment with tubes and tanks present everywhere. For these reasons, using GPS may not be a good choice due to its low accuracy or infeasibility because of obstacles.

Our study presented in [23] shows that, within the refinery, whereas there are no other wireless networks that interfere with the IEEE 802.15.4, there is still a considerable amount of signal noise caused by several types of machines and pumps that work 24h a day. In addition, the effects of noise are different between channels of the same standard. Moreover, the noise and interference severely affect the performance of the IEEE 802.15.4 based sensor networks in term of delay and packet loss as shown in [18]. Consequently it is necessary to have a suitable localization for such scenarios. This means that the localization solution should work in a noisy environment with a lot of obstacles, but with low power consumption.

Fig. 1. A typical part of a refinery

The position can be calculated at the mobile nodes or at a central server, depending on which place is more appropriate. However, in the case the mobile node estimates the location for itself, the location data has to be sent to the central servers for further processing. In our approach, to reduce the power consumption as well as the computation burden of the sensor nodes, we decided to develop a central localization system in which the positions of mobile nodes are estimated at the central servers. The next section presents the proposed model.

IV. SCALABLE LOCALIZATION SYSTEM

In order to create a positioning system that can response to a large number of concurrently requests, we model our system using multi-layered software architecture. In addition, we also employ the service-oriented principles to our system, i.e., the communication between layers is based on web services. The following sub-sections detail the proposed system.

A. General Model of the System

The main goal of this model is to allow interoperability between the WSNs and external environment such as web 2.0, virtual worlds (e.g., second life [24]) or enterprise applications. In addition, the model should also be flexible to easily add infrastructure services (e.g., localization and mobility) for WSNs. Moreover, it is also scalable to apply for large WSNs. As shown in Fig. 2 the model employs the proxy and gateway as a mediate layer for interoperability between WSNs and the applications that monitor and control them.

The proxy is responsible for obtaining, analyzing and forwarding the data from WSNs to the gateway. It uses the event mechanism to activate the appropriate functionalities (e.g., sending data to the gateway for storing). It also includes components for supporting localization as well as for sending commands to the sensor nodes (e.g., configuration or controlling commands).

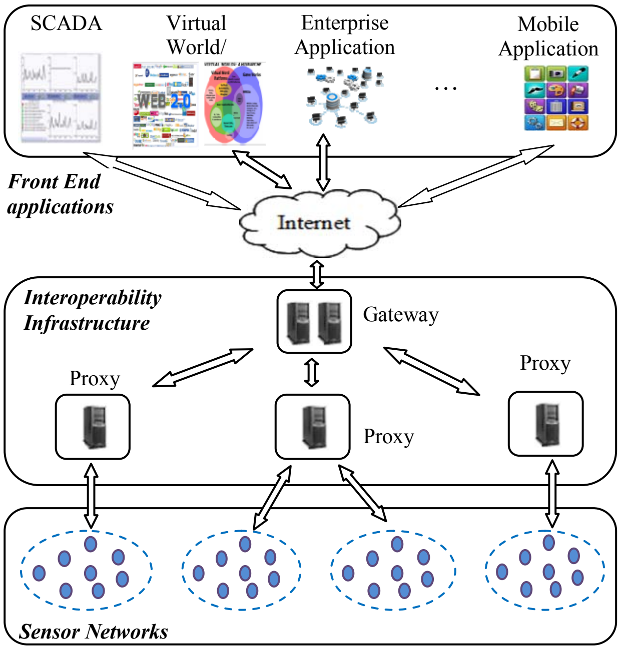

Fig. 2. The General Model for the Location System

The gateway, also called web service gateway, acts as the entry point for the system, i.e., it accepts and responses the requests from the proxies and user applications. It hosts the middleware including localization engine, which implements the localization methods. The localization engine accepts the localization requests from the proxies, computes the location, and returns the results back to the proxies. More importantly, it also stores the location results into the database on the gateway (or a database server) for using by other applications such the visual application or SCADA. The main advantage of the web service gateway is that it isolates the services from their consumers. In addition, any changes to the middleware will not affect the client applications provided that its interfaces are kept intact. Moreover, we can employ the central security control at the gateway.

To reduce the unnecessary overhead, the communication between the sensor network and proxy can be serial port or IP based. To support the scalability of sensor network, there may be more than one proxy associated with one gateway. Moreover, there can be more than one gateway in the system.

B. Localization Middleware

The localization middleware comprises a list of location services (with different algorithms) for estimating the position of the mobile nodes. The general components of the location engine are shown in Fig. 3. Besides a list of implemented localization algorithms, the middleware also includes services for proxies to request the positioning service as well as other applications to access the location results. There are two methods for applications to get location data from the gateway: polling and publisher-subscriber model. The former method requires the application periodically sends request to the gateway to get the latest location data. On the other hand, with the publisher-subscriber model, the application only subscribes the location request to the gateway once for each interest data. Then, the location engine will raise an event and inform the application for every new location result. All of these functionalities are provided by the gateway via web service interfaces.

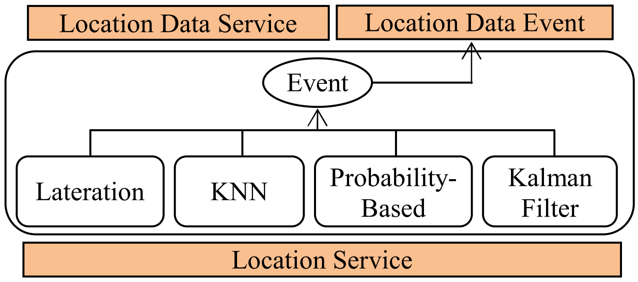

Fig. 3. Localization Middleware

In order to be able to evaluate and select the appropriate localization method for different sensor networks, we implemented the following localization methods in our location engine: Lateration, KNN, Probability-based, and Kalman filter over probability-based. For the final localization method, we combine two algorithms: (1) Probability-based; and (2) Kalman filter. This method first estimates the location of a node using the Probability-based method. Then, the resultant position is used as input for the Kalman filter. Because we donot have a rigorous mathematical model for this case, we assume that the estimation of the current position is linearly related to the previous estimated position, the current measurement and the noise. Consequently, our model for Kalman filter is as follows:

$$\begin{cases} x_k = x_{k-1} + w_{k-1} \\ z_k = x_{k-1} + v_{k-1} \end{cases}$$

In which, z_k is the position given by the probability-based method; x_k is the final estimated position; w_{k-1} and v_{k-1} are the process and measurement noise, respectively.

In our implementation, we assume that the noise does not change over time, i.e., $w_k=w_{k-1}=\ldots=w_1$ and $v_k=v_{k-1}=\ldots=v_1$. Consequently, the process and measurement noise matrixes are computed once and based on the errors of the Probability-based method.

C. Scalable Mechanisms

The above localization methods work well for sensor networks with limited number of anchors. However, they will have some problems when applied for sensor networks that have a large number of anchors. First, the data training significantly grows if it comprises all the anchors of the entire sensor network. In addition, the location estimation process is slower when the training data set becomes larger. Moreover, because the range of each anchor is limited, only a small number of anchors are present in each record of the training dataset. As a result, it wastes a lot of system resources and affects the performance. The problems are more severe for the KNN and Probability-based methods. The reason is that these two methods belong to the lazy machine learning algorithm type, i.e., instead of creating the model from the training dataset these algorithms use it as is during estimation process. Therefore, making the training dataset smaller results in computation is faster. However, if we do this by taking fewer measurement positions, then the accuracy will be reduced. Consequently, we have to keep taking the same number of measurements but somehow reduce the number of records in the estimation process.

Our solution for this problem is to divide the entire network into smaller areas called subzones. With this division, the entire training dataset will be divided into multiple datasets (one for each subzone). The training dataset for each subzone is reduced and only comprises data collected in the area covered by the corresponding subzone. By organizing training dataset in this way, the performance of the location estimate is significantly improved. In fact, the location computation time remains the same regardless how large the sensor network is. The problem with this approach is to select the subzone for each localization request. To do this, for each subzone we select one or more position as the representative records. As a result, we come up with a two steps localization process: (1) subzone determination (2) localization estimation on the selected subzone.

In the first step, the location engine computes the similarity (e.g., distance) between the input data and the representative records of the subzones. Then, the closet subzone is chosen to use for the second step, i.e., for the localization process, which is the same as the one presented in previous section, but with the subzone dataset. To further improve the accuracy as well as the performance, for each location request we use a few readers with strongest RSSI values to limit the possible area of the tag. The result of this step is list of subzones for the location engine to choose in the first step.

V. The Testbeds and Experimental Results

This section presents the testbeds and the experimental results obtained. The objectives are to evaluate our solution and to compare the performance of different localization methods. More importantly, it also presents the experimental results obtained in a real critical industrial environment, the Soporcel paper mill, and the comparison of the performance between the different environments.

A. Testbed Environments

The testbed uses the hardware developed by Eneida [25], an engineering company specialized in Instrumentation, Energy and Communications dedicated to the process industries. The equipment used in the testbed includes: (1) the smart active tags (EWS µ433M and EWS µ433M1Ex), which broadcasts its own ID; (2) the wireless communication devices (also called readers) (EWS G433M and EWS G433M1Ex), which accept the data from the tags and forward them to the gateway devices; and (3) the communications gateway devices (EWS GIP or EWS GIPW), which accept the data from the readers and forward them to the central servers for further processing. These sensor networks operate on the frequency band of 433MHz. One possible topology of this type of sensor network is depicted in Fig. 4. The communication between readers and the gateway can also be wireless.

To evaluate the performance of the location system with different environments, we set up and did the experiments with two testbeds. The first one consists of two sites: the first site is Instituto Pedro Nunes in Portugal (IPN) building in which Eneida Company is situated and the second one is

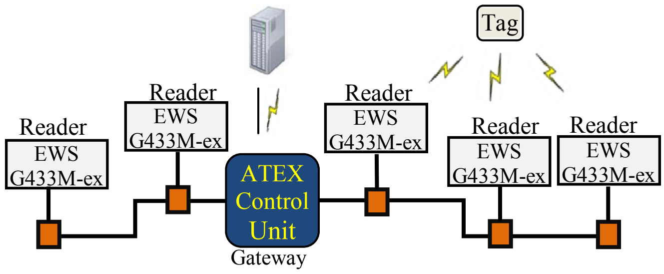

Fig. 4. . EWS Sensor Network Topology

at fire department which is about 300 meters from IPN building. The former is a two-story building where readers were deployed in both floors, creating two different subzones. Consequently, the entire sensor network is divided into 3 subzones: subzone 1 is located at the 1st floor of IPN, subzone 2 at the 2nd floor of IPN building and zone 3 at the fire department. The layout of the testbed is shown in Fig. 5. In this scenario the localization system is implemented on one server and placed at the 2nd floor of the IPN building.

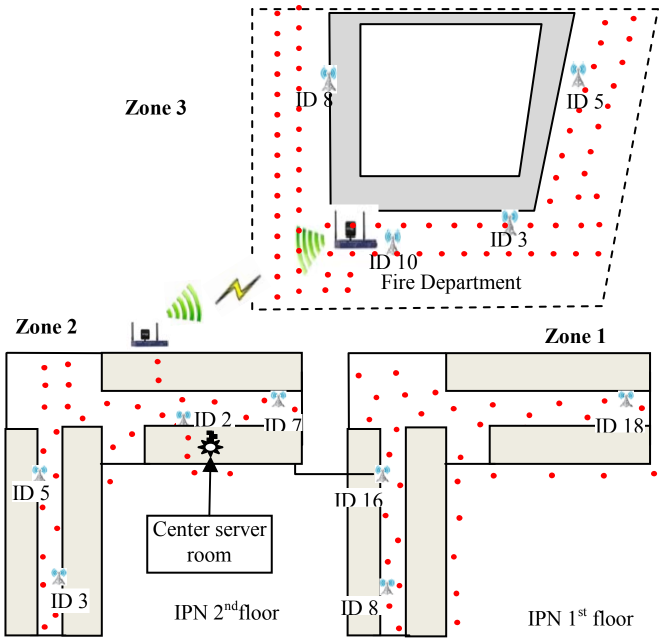

Fig. 5. Test-bed Environment at IPN building

The second testbed was set up in the recovery boiler for power production building of the Soporcel paper mill, Portugal, which is a very hazard and critical environment. There were 11 readers installed on the floor 4, 5, and 6 of the 18 story building. The area of each floor is about 990 square meters (33 m x 30 m).

B. Experimental Results

With the installed test-beds we did numerous experiments with different configurations and objectives. Before presenting the results of the location methods, we did numerous analyses to find ways to improve the accuracy and robustness of the location methods.

1) RSSI Calibration

In order to find a good correlation between RSSI value and distance, we tried to evaluate four possible methods to convert RSSI value to distance: (1) Mean RSSI value; (2)

linear regression; (3) Friis Transmission Equation [26]; and (4) kernel smoother algorithm [27]. To evaluate these methods, we collected RSSI values between a tag and a reader at different distances from 1 to 40m with 1 meter a step. At each distance one hundred RSSI values were collected.

It can be seen from the Table 1 that the kernel smoother is more stable (smallest standard deviation - SD) than the others. In addition, this method has also the best error rate with 90% of errors below 8.26m. For most of the methods (except Friis equation) the average error is around 5m. We can see that the average RSSI and linear regression methods also have a average and median similar to those obtained from the kernel smoother method. However, they have the larger SDs and have more outliers. Although the Kernel Smoother is better than the other methods, its average error is also high (5.16m).

Table 1
ERRORS IN RSSI VALUE TO DISTANCE CONVERSION

	Average of RSSI	*Linear regression*	*Friis Equation*	*Kernel Smoother*
Min (m)	0.01	0	0.17	0.07
Max (m)	27.38	34.67	65.58	19.43
Average (m)	5.39	5.26	11.88	5.16
Q1 (m)	0.98	1.94	6.79	2.93
Median (m)	3.73	3.84	11.00	4.72
Q3 (m)	6.70	7.06	16.53	6.83
SD	5.07	4.92	6.76	3.55
90 percentile (m)	12.04	10.86	18.46	8.26

2) The Spread and Distribution of RSSI Values

Finding the spread of RSSI is important because it provides information about how stable (or unstable) the RSSI is. In addition, it is worth noting that the probability-based localization methods assumed that the distribution of RSSI values is normal. Consequently, it is necessary to have an empirical study to see whether or not this assumption is acceptable. In order to evaluate the spread and distribution of RSSI values, we put a tag and a reader at fixed positions and measured the RSSI values received at the reader. We totally collected 58978 RSSI values. The result of this experiment is depicted in Fig. 6.

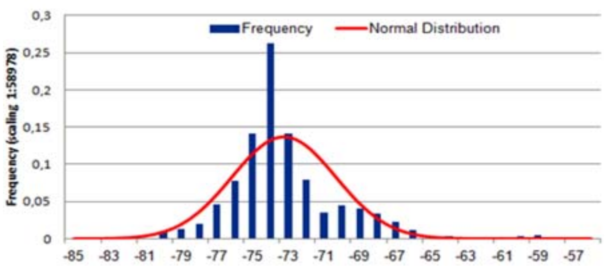

Fig. 6. Distribution of RSSI Value

The histogram in Fig. 6 indicates that the spread, the range between the maximum and minimum, of RSSI value is 29. This implies that we can conclude that the RSSI values are unstable because of a very high range and large SD. The red line is normal distribution, using the computed mean and SD, over the scaled histogram of the real data. Although these two graphs are not totally matched each other, the distribution of the RSSI values is approximate to the normal distribution. With this result it is acceptable to use the normal distribution in the probabilistic-based localization algorithms.

3) Training Dataset

In order to estimate the position of a node using the machine learning based localization methods, a training dataset is needed. A training dataset comprise a list of records which are tuples with the following information: <x,y,z, $rssiAvg_1$, $rssiStd_1$, $rssiAvg_2$, $rssiStd_2$,…, $rssiAvg_n$, $rssiStd_n$>. In which:

- ✓ x, y, z are the coordinates of the position where the data is collected;
- ✓ the $rssiAvg_i$ is the average RSSI value from the anchor i^{th} at position x, z, y;
- ✓ $rssiStd_i$ is the SD of the RSSI values of anchor i^{th} at position x, z, y.

Because collecting data is very time-consuming, we could not collect the data in a very fine-grained position (e.g. 1 or 2 square meters each). Therefore, we applied a method similar to that of used in [17], i.e., we collected RSSI values at the distance about 3-5 square meters and then did interpolation to create a fine-grained dataset. To make the testing of the localization algorithms more reliable, we independently collected two datasets: one for training and the other for testing. With testbed at IPN building, the positions at which the data were collected are shown as red points in Fig. 5.

As discussed previously, the accuracy of lateration algorithm depends on the accuracy of the distance measurement between the tag and the anchors. However, previous experiments showed that the conversion between RSSI values and distance has a very high error (Table 1) and the RSSI value is also unstable, with high variance. In our experiments, the accuracy of this method is a very low with the majority of errors was greater than 10m. Therefore, in the following sections, we will not include it in our evaluation.

4) Evaluating the Localization Methods with the laboratory Testbed

This section presents the experimental results with the testbed installed at IPN building. As shown in Table 2 and Fig. 7, it is possible to obtain an average error lower than 3.8m. In addition, the Probability-based and Kalman filter over Probability-based methods are more accuracy (having a lower mean) and more stable (smaller SD) than those of KNN. Comparing to KNN, it can be seen that the average errors of the probabilistic method are improved about 11.12% (from 3.79m to 3.37m). Similarly, the average accuracy of Kalman filter over Probability-based method is also better that of KNN about 14.89% (3.79m and 3.33m, respectively). Equally important, as shown in Fig. 7, the 80^{th} percentile of errors is reduced from 6m (KNN) to about 5m (other methods).

Table 2
THE ACCURACY OF LOCALIZATION ALGORITHMS IN LABORATORY ENVIRONMET

	KNN	***Probability***	***Kalman Filter***
Min (m)	0.07	0	0.06
Max (m)	18.09	16.02	14.56
Average (m)	3.79	3.37	3.33
Median (m)	2.31	1.99	2.06
75 Percentile (m)	5.11	4.35	4.21
80 Percentile (m)	6.09	5.34	5.09
SD (m)	2.01	1.78	1.58

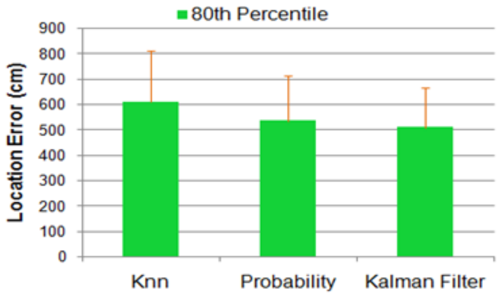

Fig. 7. The Accuracy of Localization Algorithms at IPN

5) Evaluating the Localization Methods with the Testbed in real critical industrial Environment

To evaluate the performance of the implemented location algorithms in a real critical industrial environment, we installed a testbed in the recovery boiler for power production building, located in the Soporcel paper mill. With this testbed, we did numerous experiments with different configurations and objectives. Table 3 and Fig. 8 present the results of experiments with the testbed comprising 11 readers installed on 3 floors (4, 5 and 6) of this building.

Table 3
THE ACCURACY OF LOCALIZATION ALGORITHMS IN SOPORCEL TESTBED

	KNN	*Probability*	*Kalman Filter*
Min (m)	0.02	0	0
Max (m)	20.49	19.68	19.04
Average (m)	4.49	4.32	4.34
Median (m)	1.87	1.22	3.39
75 Percentile (m)	7.53	7.56	7.19
80 Percentile (m)	8.62	8.67	8.07
SD (m)	2.52	2.57	2.25

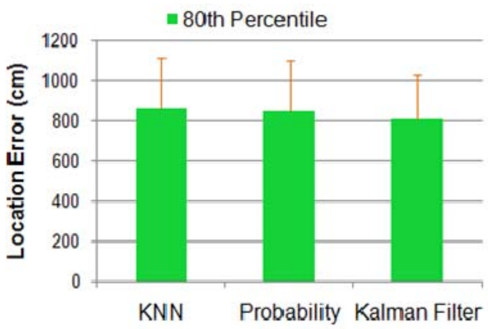

Fig. 8. The Accuracy of the localization Algorithms at Soporcel

Comparing with the experimental results at the IPN building, it can be seen that the average accuracy in this environment is significantly reduced. In particular, the average location error of KNN is increased from 3.79m to 4.49m (15.52%) and those of Probability-based and Kalman Filter over Probability-based methods are increased from 3.37m to 4.32m (21.94%) and from 3.23m to 4.34m (25.64%), respectively. There are several possible explanations for this result. First, the second testbed was installed on a very noisy environment with a lot of motors, pumps, and other machinery. In addition, the heating from the recovery boiler also affected the stability of RSSI and thus the accuracy of the location estimate. Furthermore, the obstacles also contributed to the accuracy of the system. Moreover, the position of the readers is also a factor that influences the performance of the system. It is also important to note that when comparing to KNN, the accuracy of the other two methods is also a little bit better. However, the differences are not as substantial as those obtained in the first testbed.

With the collected data, we also analyzed the accuracy of the location system at different areas in the real environment. The objective of this study was to find out the subzones whose accuracy is better than that of the others. In this study, the floors 4, 5, and 6 were divided into 9, 7, 7 sub zones, respectively. From the results of the experiments, we found out that among these 23 subzones, 9 of them (39.13%) had the average distance errors less than 2.23m with 80% of errors were less than 4.4m. The Table 4 and Fig. 9 describe the location errors of these subzones.

Table 4
THE ACCURACY OF BEST SUBZONES OF SOPORCEL TESTBED

Sub-zone	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Min	0	0	0	0	0	0	0	0	0
Average	2.08	1.16	0.42	1.32	2.02	2.23	1.43	2.18	1.94
Median	0.18	0.19	0	0.17	0.19	0.46	0.06	0.16	0.29
75 Percentile	1.46	0.42	0.14	0.94	3.39	3.19	1.82	3.42	3.46
80 percentile	4.26	0.50	0.74	3.59	4.11	4.24	4.25	4.41	4.33
90 percentile	6.05	5.93	1.47	4.80	5.05	5.21	5.27	5.88	6.01
Max	10.66	8.12	9.58	9.29	10.38	10.52	10.94	10.88	9.23
Std,Dev	1.79	1.13	0.75	1.09	1.35	1.39	1.22	1.79	1.44

Unit: m

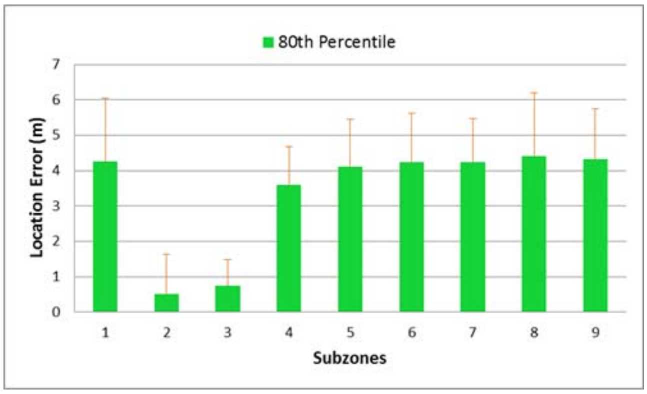

Fig. 9. The Accuracy of the Best Subzones at Soporcel

Observing the locations of these best subzones, we recognized that all of them were located in the boundaries of at least three readers. In addition, the communication between the readers and tags in these subzones were rather clear. This is not necessarily in LOS communication, but the obstacles between the reader and tag were not too big. Moreover, the distance between the readers also contributes to the accuracy, as in these best subzones the distance between them is about 15m. The implication of this study allows us to design and deploy a RTLS with different levels of accuracy for different zones. For instance, we should deploy more readers for the high critical zones and fewer readers for other less important ones.

VI. CONCLUSION

In this paper we presented our proposed model for a localization system that can be applied for large-scale WSNs. The localization engine implements numerous localization methods. We also did several experiments with two testbeds on both laboratory and industrial environments to evaluate the accuracy of the implemented localization methods.

The results from experiments showed that although RSSI values are unstable with high variance, pattern matching based methods such as KNN, probability-based and Kalman Filter over Probability-based produce an acceptable accuracy for application scenarios in industrial environment. With these methods, it is possible to achieve an average error less than 4m and 80 percent of error less than 6m. More importantly, with carefully engineering the positions of readers, it is possible to reduce the average errors to about 2.2m.

From the results of our experiments and from those of other authors, e.g., [17], we can conclude that by only using RSSI values we cannot make more improvement on the accuracy of localization than those obtained in our experiments. The fundamental problem concerns the lack of stability of the measured RSSI values, and not of the algorithms. Therefore, if more accuracy is wanted in a positioning system, other measurement methods must be used.

ACKNOWLEDGMENT

The work presented in this paper is partially financed by Eneida Wireless & Sensors, S. A., by COMPETE, QREN Portugal and European Union, by iCIS project (grant CENTRO-07-ST24-FEDER-002003) and by the Portuguese Foundation for Science and Technology, FCT.

REFERENCES

[1] A. El-Rabbany, *Introduction to GPS: The Global Positioning System*, Second Edition ed.: Artech House Publishers, 2006.

[2] J. Liu et al., "Energy Efficient GPS Sensing with Cloud Offloading," in *10th ACM Conference on Embedded Networked Sensor Systems (SenSys 2012)*, Toronto, 2012.

[3] M. S. & Andrews Grewal and A. P., *Kalman Filtering: Theory and Practice Using MATLAB*, 3rd ed. New Jersey: Wiley-IEEE Press, 2008.

[4] R., Hopper, A., Falcão, V. & Gibbons, J. Want, "The active badge location system," *ACM Transactions on Information Systems (TOIS)*, vol. 10, no. 1, pp. 91-102, January 1992.

[5] Y.C., Chawathe, Y., LaMarca, A. & Krumm, J. Cheng, "Accuracy characterization for metropolitan-scale Wi-Fi localization," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services (MobiSys '05)*, New York, NY, USA, 2005, pp. 233-245.

[6] A. Bjõrck, *Numerical Method for Least Squares Problems.*: SIAM: Society for Industrial and Applied Mathematics, 1996.

[7] W. & Murphy, W. S. Hereman, "Determination of a position in three dimensions using trilaterationand approximate distances.," Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, Technical Report 1995.

[8] A. Harter, A. Hopper, P. Steggle, A. Ward, and P. Webster, "The Anatomy of a Context-Aware Application," in *5th ACM MOBICOM* , Seattle, WA, 1999.

[9] N.B., Chakraborty, A. & Balakrishnan, H. Priyantha, "The Cricket Location-Support System," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, Boston, MA, USA, 2000, pp. 32-43.

[10] A. Teuber and B. Eissfeller, "A two-stage fuzzy logic approach for wireless LAN indoor positioning," in *IEEE/ION Position Location Navigation Symp*, 2006, pp. 730–738.

[11] M. & Tan, P. Steinbach, *kNN:k-NearestNeighbors*, X. & Kumar, V. Wu, Ed.: Chapman and Hall/CRC, 2009.

[12] P. & Padmanabhan, V. N. Bahl, "RADAR: An in-building RF-based userlocation and tracking system," in *IEEE INFOCOM*, vol. 2, 2000, pp. 775–784.

[13] Konrad Lorincz and Matt Welsh, "MoteTrack: a robust, decentralized approach to RF-based location tracking," *Personal Ubiquitous Comput.*, vol. 11, no. 6, pp. 489-503, August 2007.

[14] L.M., Yunhao, L., Yiu C.L. & Patil, A.P. Ni, "LANDMARC: indoor location sensing using active RFID," in *The First IEEE International Conference on Pervasive Computing and Communications (PerCom 2003)*, Fort Worth, Texas, USA., 2003, pp. 407- 415.

[15] M. A., Agrawala, A. & Shankar, A.U. Youssef, "WLAN location determination via clustering and probability distributions," in *The First IEEE International Conference on Pervasive Computing and Communications*, Washington, DC, USA, 2003, pp. 143-151.

[16] Moustafa Youssef and Ashok Agrawala, "The Horus WLAN Location Determination System," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services (*

[17] E., Li, X. & Martin, R.P. Elnahrawy, "The limits of localization using signal strength: a comparative study," in *The First International Conference on Sensor and Ad Hoc Communications and Networks (SECON2004)*, Santa Clara, CA, 2004, pp. 406-414.

[18] J. Oliveira, J. Fonseca, P. Bartolomeu, and L.C. Costa, "Evaluating severe noise interference in IEEE 802.15.4 based location systems ," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008)*, Hamburg , 2008, pp. 893 - 898.

[19] A., Zhu, Y. & Musavi,. Shareef, "Localization using neural networks in wireless sensor networks," in *The 1st international conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (MOBILWARE'08)*, Innsbruck, Austria, 2008, pp. 1-7.

[20] Mohammad S. Rahman, Youngil Park, and Ki-Doo Kim, "Localization of Wireless Sensor Network using artificial neural network," in *9th International Symposium on Communications and Information Technology (ISCIT 2009)*, Icheon , 2009, pp. 639 - 642.

[21] A. & Zhu, Y. Shareef, *Localization Using Extended Kalman Filters in Wireless Sensor Networks*, V. M. & Pigazo, A. Moreno, Ed. Crotia: In-Teh, 2009.

[22] M. & Battiti, R. Brunato, "Statistical learning theory for location fingerprinting in wireless LANs," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 47, no. 6, pp. 825-845, April 2005, DOI:10.1016/j.comnet.2004.09.004.

[23] T.D., Silva, R., Nunes, D. & Sa Silva, J. Tran, "Characteristics of Channels of IEEE 802.15.4 Compliant Sensor Networks," *Wireless Personal Communications*, vol. 67, no. 3, pp. 541-556, 2011.

[24] Inc. Linden Research. (2013, Jan.) Second Life. [Online]. http://secondlife.com/

[25] Eneida. (2012) Eneida Company. [Online]. http://www.eneida.pt/produtos/

[26] X., Bischoff, O., Laur, R. & Paul, S. Wang, "Localization in Wireless Ad-hoc Sensor Networks using Multilateration with RSSI for Logistic Applications," in *The Eurosensors XXIII conference* , 2009, pp. 461-464.

[27] E. Alpaydin, *Introduction to Machine Learning*, 2nd ed., E. Alpaydin, Ed. Cambridge, Massachusetts London, England: The MIT Press, 2010.