Bayesian Filtering for a Bluetooth Positioning System

Javier Rodas, Carlos J. Escudero, Daniel I. Iglesia
Electronic and Systems Department, Universidade da Coruña.
{jrodas, escudero, dani}@udc.es

Abstract—Positioning systems are one of the multiple applications of the wireless sensor networks. These networks are very adequate in environments where other positioning technologies, as satellite systems, do not work. Bluetooth is a promising technology, since it is present in any kind of portable devices. By using the Received Signal Strength Indicator (RSSI) it is possible to make an estimation of the distance between a transmitter and a receiver. By using this information, it is possible to develop an algorithm that estimates positions, even with the Bluetooth constraints when the RSSI is obtained. Our experimental results show that our algorithm, based on a particle filter, can achieve a good performance.

I. INTRODUCTION

Location systems are one of the most interesting applications of the wireless sensors networks. They are being demanded due to the continuous proliferation of the mobile communications systems and portable personal devices. In this kind of applications, the Wireless Sensors Networks (WSN) gather some physical parameters used for positioning the devices in the coverage area. These networks are complementary to the Global Navigation Satellite Systems (GNSS) [2], which are valid only for outdoor environments where there is line of sight with the satellites and they fail in shadow zones and indoors.

The WSN can be based in standards like Bluetooth, WiFi, ZigBee, UWB, Ultrasounds, etc. [1], that obtain different kind of physical parameters. The decision of the standard to be used will depend on the characteristics on the environment (extension, obstacles, . . . ), the kind of devices that are desired to locate, precision, ...

One of the standards with more penetration in the last years is Bluetooth. Most of manufacturers are integrating it in all type of portable devices, like mobile phones, MIDs (Mobile Internet Devices), PDAs (Personal Digital Assistant), UMPCs (Ultra Mobile PCs), laptops, . . . . The evolution of the multimedia microprocessors, the elimination of cables in the communications and the clear tendency to miniaturize the devices, ensure the success of this type of wireless technologies.

In this paper we present a location system based on a Bluetooth sensor network, analyzing the most characteristic problems. To overcome these problems we introduced a new algorithm based on a particle filter [7].

The paper is organized as follows: In the section II we analyze the characteristics, advantages and disadvantages that offer these networks, applied to location systems. In the section III we it is introduced an algorithm based on a particle filter, used for the estimation of the position of a mobile device. The section IV shows the results obtained by using a experimental network. In this section, we will show the advantages of using the introduced algorithm, emphasizing which are the challenges to solve in this type of networks. Also we will compare its performance with respect to classical algorithms of location based on multilateration techniques. Finally, the section V is devoted to the conclusions.

II. LOCATION WITH BLUETOOTH SENSOR NETWORKS

Due to the characteristics of the radio signal, when using a Bluetooth network for positioning tasks, the physical parameter used for distance estimations is the Received Signal Strength Indication (RSSI). With this parameter we can obtain the loss produced due to the propagation. It is well-known that the power of the transmitted signal decays exponentially with the distance, depending on the obstacles that surround or interpose between the transmitter and the receiver [4]. Therefore, from the RSSI it can be obtained a distance estimation.

In Bluetooth, the RSSI parameter can be obtained in a relatively simple way by using the standard feature named Inquiry, which detects devices in the area coverage. Moreover, since Bluetooth release 1.2, there exists an extended function of the Inquiry called Inquiry Result with RSSI [3], that provides the RSSI level form the detected device (it also obtains the MAC, clock offset and the class of the detected device). This way, by using Bluetooth devices, we can estimate the distance between nodes. Therefore, if we consider several Bluetooth nodes (beacons), with known positions, asking for mobile devices (i.e. launching Inquiries), we could estimate the position of these devices.

Nevertheless, there exists several problems to obtain the RSSI with Bluetooth that we can summarized in two:

- The Inquiry scheme does not guarantees a periodic answer. Unlike which it happens typically in other networks, like Wifi or ZigBee, where pilot sequences are transmitted periodically with beacons each 100ms, with Bluetooth the information about nodes is obtained aperiodically.
- Since Bluetooth is a technology considered for low consumption, the percentage of time that each device is listening Inquiry requests (i.e. in Inquiry Scan state) is relatively small. Normally, a Inquiry Scan window time has a duration of 11.25ms and the period of Inquiry (time between two consecutive windows) is 1.28s. Moreover, since Bluetooth uses a pseudo-random frequency hopping.
pattern to transmit [3], it is necessary some random time to synchronize frequencies between transmitter and receiver.

- The number of answers from a detected device is reduced with the number of existing devices in the coverage area and with the number of beacons that are doing of Inquiry operations [5]. This is produced since a device only can answer to a single inquirer and some collisions between responses can be produced.

Therefore, the reduced number of beacons signals by unit time is a special feature of Bluetooth and it represents a big challenge for positioning systems. The time from RSSI measures varies from the milliseconds to tens of seconds [6], and it is completely aperiodic. We cannot guarantee a deterministic time between RSSI measures in the beacons nodes.

On the other hand, in general, the RSSI varies in a random way depending on the environment characteristics. These variations can be interpreted by means of propagation models of small and great scale [4], that represent statistically the changes in the signal level. In this paper, we use a classic model of propagation based on the path loss produced with the distance. It is important to note, that this model is just an approximation, since it does not consider the effects caused by the multipath fading. The model is defined as:

\[
P_L(d)[dB] = P_L(d_0) + 10n\log \left( \frac{d}{d_0} \right) + X_{\sigma_L} \tag{1}
\]

where \(P_L(d)\) is the received signal power in a distance \(d\), \(P_L(d_0)\) is the same power but in a reference distance \(d_0\), \(n\) is the path loss exponent and \(X_{\sigma_L}\) represents the noise using a random variable with normal distribution, zero mean and standard deviation \(\sigma_L\). As we can see, the random variations for the power follow a log-normal model. To estimate \(n\) and \(\sigma_L\) we use linear regression analysis from real data RSSI.

III. PARTICLE FILTER ALGORITHM

The particle filter is a Monte Carlo (MC) method for implementing a recursive Bayesian filter [7]. It is based on a set of random samples, denominated particles, associated to different weights that represent a probability density function (pdf). Basically, the objective is to construct the a posteriori pdf recursively, \(p(s_i(t)|z(t))\), where \(s_i(t)\) is the state of the particle \(i\)-th and \(z(t)\) is the observation that we have at the \(t\) instant.

In our case, the state of the \(i\)-th particle is composed by its position coordinates\(^1\), \(x_i\) and \(y_i\), its speed, \(v_i(t)\), and its direction, \(\alpha_i(t)\). Moreover, each particle has an associated weight \(w_i(t)\) directly related to \(p(s_i(t)|z(t)−1))\) [8].

The algorithm works as follows:

First, \(N_p\) particles are initialized with random states and identical weights \(w_i = 1/N_p\). Periodically, the algorithm makes successive iterations each one with the following steps:

1) Prediction step: the new state of each particle is determined by using a dynamic model that updates the position, speed and direction. This model plays a very important role as we will see in the section of results.

In this article we consider two dynamic models: one simplified (that not considers the speed and the direction of the particles) and other based on the dynamics of realistic human walking [9].

- The simplified model considers static positions, with zero speed, that are updated by using random noise as follows:

\[
x_i(t) = x_i(t−1) + n_x Δt
\]

\[
y_i(t) = y_i(t−1) + n_y Δt
\]

\[
n_x ∼ \mathcal{N}(0, \sigma_{pos})
\]

\[
n_y ∼ \mathcal{N}(0, \sigma_{pos}) \tag{2}
\]

where \(\mathcal{N}(\mu, \sigma)\) represents a Gaussian distribution with \(\mu\) mean and \(\sigma\) typical deviation, and \(Δt\) is the time interval between iterations.

- The model of human walking considers the speed and typical trajectories of a pedestrian. In this case, it is considered that the direction of a person has \(360^o\) of freedom when it begins to walk, but when the maximum speed is reached, \(v_{max} \approx 1.3 m/s\), its trajectory is a straight line [9]. Considering this constraint, the state of a particle will be updated in the following way:

\[
x_i(t) = x_i(t−1) + v_i(t−1)cos(\alpha_i(t)) Δt
\]

\[
y_i(t) = y_i(t−1) + v_i(t−1)sen(\alpha_i(t)) Δt
\]

\[
v_i(t) = v_i(t−1) + n_v Δt, \quad v_i(t) \in [0, v_{max}]
\]

\[
α_i(t) ∼ α_i(t−1) + π n_α(t)
\]

\[
n_v(t) ∼ \mathcal{N}(0, \sigma_{vel})
\]

\[
n_α(t) ∼ \mathcal{U} \left( \frac{v_i(t)}{v_{max}} - 1, 1 - \frac{v_i(t)}{v_{max}} \right) \tag{3}
\]

where \(\mathcal{U}(a, b)\) represents an Uniform distribution between \(a\) and \(b\) values.

2) Update step: The particles weights are updated as follows [8]:

\[
w_i(t) = \pi_i(t−1)p(z(t)|x_i(t))
\]

\[
\pi_i(t) = \frac{w_i(t)}{\sum_{j=1}^{N_p} w_j(t)} \tag{4}
\]

where \(\pi_i(t)\) represents the normalized \(i\)-th weight. The likelihood function, \(p(z_i(t)|x_i(t))\), is defined by the propagation model (1) and the known statistics of \(X_{\sigma_L}\).

In our case, this random variable is Gaussian and, therefore, we can express this function for the \(k\)-th beacon as follows:

\[
p(z_k(t)|x_i(t)) = \frac{1}{2\pi\sigma_L} \exp \left( -\frac{1}{2\sigma_L^2} (z_k(t) - P_L(d))^2 \right) \tag{5}
\]

where \(P_L(d)\) is determined by (1).

\(^1\)Without loss of generality, we are supposing that everything moves in the same plane, avoiding the azimuth.
Considering the likelihood of the $K$ different beacons, we obtain the following expression:

$$p(z(t)|x_i(t)) = \prod_{k=1}^{K} p(z_k(t)|x_i(t))$$

3) Resampling step: After few interactions many particles degenerates by obtaining negligible weights. In order to avoid the problem of particle degeneration, it is necessary to generate new particles. The main idea is to generate $N_p$ new particles by using a sampling scheme that eliminates particles with small weights and replicates particles with large weights. The resampling process should be made when the particle degeneration is detected. Therefore, we use the effective sample size:

$$N_{eff} = \frac{N_p}{\sum_{i=1}^{N_n} w_i^2(t)}$$

Resampling will be made if $N_{eff} < N_{threshold}$, where $N_{threshold}$ is a threshold that indicates a severe particle degeneration.

4) Estimation step: Finally, the position estimation and the speed of the object is made by means of the weighted sum of the information provided by all particles of the following way:

$$x(t) = \sum_{i=1}^{N_n} \pi_i(t)x_i(t)$$

$$y(t) = \sum_{i=1}^{N_n} \pi_i(t)y_i(t)$$

$$v(t) = \sum_{i=1}^{N_n} \pi_i(t)v_i(t)$$

$$\alpha(t) = \sum_{i=1}^{N_n} \pi_i(t)\alpha_i(t)$$

IV. RESULTS

This section presents the results of some experiments in order to analyze the behavior of the algorithm introduced in the previous section. These experiments are based on empirical measurements obtained from four Bluetooth devices v2.0 (model AIRcable Host XR, with omnidirectional antenna of 2 dBi) connected to GNU/Linux host, that are the beacon system (Inquirers) and a mobile phone with Bluetooth support, as the mobile device to be located. The measurements were done in a laboratory of the Faculty of Computer Science of A Coruña, with dimensions 6 x 10 meters, where the beacons were placed at the corners.

Figure 1 shows how the model in (1) fits the real path loss. These measurements were obtain at distances between 1 and 9 meters, from one beacon to the mobile device. Observe that we have obtain, by using linear regression, a path loss exponent, $n = 1.801$ and a standard deviation $\sigma_L = 5.0466$ dB. It is important to mention that the log-normal model is a good approximation when the Line Of Sight (LOS) predominates over the multipath effect. However, it is well known [10] that Rice, Rayleigh and/or Nakagami distributions could be more suitable for situations where the multipath has strong influence. All the measurements made for this section consider LOS.

As we have mentioned before, the number of Bluetooth responses to an inquiry process depends on many factors, for example: presence of multiple Bluetooth devices [5] and random frequency hopping pattern [3]. If we consider short periods of time between RSSI measurements, $T_s$, the probability of not inquiry response from a mobile device, in a single beacon, $P_{ndetect}$, increases. Table I shows the experimental results that we obtain in our sensor network. Note that if we use a $T_s = 0.1$ seconds, typically in WIFI and Zigbee, the $P_{ndetect}$ is extremely high.

<table>
<thead>
<tr>
<th>$T_s$ (s)</th>
<th>0.1</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{ndetect}$ (%)</td>
<td>53.9</td>
<td>40.6</td>
<td>28.6</td>
<td>20.3</td>
<td>12.3</td>
<td>8.6</td>
<td>2.2</td>
<td>0.1</td>
</tr>
</tbody>
</table>

By using the real RSSI measures from the previous experiment, we had analyzed the performance of the introduced algorithm considering: $\sigma_{pos} = 0.3m$, $\sigma_{nu} = 0.6v_{max}$m/s and $N_{threshold} = 0.1$. The figure 2 shows the Cumulative Distribution Function (CDF) of the estimation position error when the mobile device is in a static position (coordinates: (3.5) meters). We have considered four different values of the RSSI sampling period ($T_s = 0.1s$, 0.5s, 1.5s, 3s) for two algorithms: the introduced particle filter with dynamic human walking model (3) and a simple multilateration algorithm [11]. Particle filter runs an iteration each $T_s$ seconds. Note
that multilateration shows a discontinuity at 3 meters, due to the physical limits of the laboratory. It can be seen as the particle filter outperforms the multilateration algorithm. When we reduce $T_s$ it is observed some performance degradation, since the number of beacons available in each RSSI sampling period is smaller. Considering a trade-off between probability of response and speed of sampling, we are going to consider a RSSI sample period, $T_s = 0.5$ seconds (i.e. each beacon will have a probability of not inquiry response from the mobile device equal to 40.6%).

In order to prove the main advantage of using the dynamic model based on human walking (3) it is necessary to try the algorithm with a mobile device in movement. Therefore, we are going to consider a computer simulation with the movements of the mobile device indicated in the table II.

In the figure 3 the position estimation error time evolution is shown. It can be seen that the highest errors take place when the object changes its dynamics, due to the inertia of the weights particles. At the moment of a speed increment, the particles with higher weight are the ones with smaller speed. Therefore, after some iterations, they will lose weight with respect to the ones with higher speed. However, when the simplified model (2) is used, the speed of the particles is not considered, so changes in the mobile device speed cause bigger errors. On the other hand, the multilateration algorithm presents a speed of adaptation similar to the particle filter with dynamic human walking model, but the error, when the mobile device is not moving, are considerable higher, due to their simplicity.

Finally, it is important to point out that recently real time experimental tests have been carried out. These results are very promising, since they corroborate the presented simulations when the effects of the multipath are negligible and, therefore, the model in (1) is a good approximation of the real signal loss. However, when there are some objects in the environment shadowing the LOS, new models [10] should be incorporated to (5).

V. CONCLUSIONS

This paper has introduced a Bluetooth sensor network location system based on a Bayesian filter algorithm. This algorithm uses the RSSI measurements obtained from a standard Bluetooth feature: the inquiry process. This process is suitable for obtaining the RSSI in a Bluetooth network, but it has some problems as, for example, the low period between measurements (inquiry responses).

We have introduced a particle filter algorithm that uses a model based on a dynamic human walking. The results of this algorithm, in real environment conditions, show a good performance that overcomes the typical problems of this kind of sensor networks when they are used as a location system.

ACKNOWLEDGMENT

This work has been supported by: 07TIC019105PR (Xunta de Galicia, Spain) y TEC2007-68020-C04-01 (Ministerio de Educación y Ciencia, Spain).
REFERENCES