LAURA - LocAlization and Ubiquitous monitoRing of pAtients for health care support

Alessandro Redondi*, Marco Tagliasacchi*, Matteo Cesana*, Luca Borsani*, Paula Tarrío†, Fabio Salice*
*Dipartimento di Elettronica e Informazione, Politecnico di Milano, Milano, Italy
†Data Processing and Simulation Group, Universidad Politécnica de Madrid, Madrid, Spain
*{redondi, tagliasa, cesana, borsani, salice}@elet.polimi.it, †paula@grpss.ssr.upm.es

***Abstract*—This works illustrates the *LAURA* system, which performs localization, tracking and monitoring of patients hosted at nursing institutes by exploiting a wireless sensor network based on the IEEE 801.15.4 (Zigbee) standard. We focus on the indoor personal localization module, which leverages a method based on received signal strength measurements, together with a particle filter to perform tracking of moving patients. We discuss the implementation and dimensioning of the localization and tracking system using commercial hardware, and we test the *LAURA* system in real environment, both with static and moving patients, achieving an average localization error lower than 2 m in 80% of the cases. The data sets containing the real measurements of received signal strengths collected during the experiments are made publicly available to enable reproducible research.**

I. Introduction

The provision of enhanced mechanisms/solutions to support health care is increasingly becoming one of the most challenging tasks faced by the modern society. Indeed, the combined effects of population ageing and nursing staff shortage may eventually lead current health care systems to collapse [1]. This is due to the fact that most of the current tasks in the hospital workflows, related to patient monitoring, care, management and supervision, are manually executed, thus constituting *de facto* an efficiency bottleneck.

It is commonly recognized that the recent achievements in the field of Information and Communications Technology may be leveraged to improve efficiency and quality of health care processes. Several initiatives within this field have been launched/stimulated in the last decade by private organizations (hospitals, pharmaceutical companies, etc.) [2], public institutions and governments, and researchers worldwide [3].

The work presented in this paper has been carried out within project *LAURA* (*LocAlization and Ubiquitous monitoRing of pAtients for health care support*), arising from the partnership between the research laboratories of the Electronics and Information Department of Politecnico di Milano and a small-size nursing institute, Fondazione Eleonora e Lidia. The latter is specialized in the assistance of people with a broad range of pathologies, including cognitive and/or perceptual disorders, Down's syndrome, epilepsy, etc.

The final goal of the project is the design and the implementation of a lightweight system based on Wireless Sensor Networks (WSNs) for the automatic supervision of patients within the nursing institute. Patient's supervision include two major services:

- **Patient Localization and Tracking:** The exact knowledge of the location of patients is a valuable asset, since it allows reaching them in a short amount of time, in case they need urgent assistance from the staff.
- **Patient's Status Monitoring:** The up to date status of critical patients must be continuously available to the medical/nursing staff, when patients can roam around the premises of the hosting institution. Depending on the specific pathology, different pieces of information on the patient's status may need to be collected (movement characteristics, heart beat, breath, proximity to other patients, etc.), eventually implementing automatic detection of anomalous changes in such parameters.

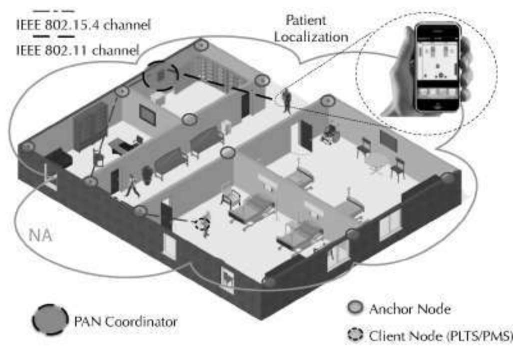

Fig. 1. Reference Functional Architecture of the System.

The *LAURA* system implements the aforementioned services through the deployment of a distributed WSN to collect the required data from the patients and deliver the very same data remotely to a central controller. Referring to Figure 1, the main architectural building blocks of the *LAURA* system are a Personal Monitoring System (PMS), a Personal Localization and Tracking System (PLTS) and a lightweight and flexible Network Architecture (NA), delivering the information to an automatic central controller. The PMS is composed of small wearable devices geared with sensing and wireless transmission capabilities to detect patient's status and transmit the sensed information to a central controller through the NA.

In this work, we describe the PLTS functional module, which leverages wearable devices mounted on the patients and the network architecture to implement the localization/tracking service. Namely, we adopt a localization algorithm based on the Received Signal Strength (RSS), which requires a minimal setup cost, and is able to track the position of moving patients with an average accuracy below 2 meters. We illustrate the implementation of the PLTS module within the *LAURA* network infrastructure, featuring a ZigBee-based wireless sensor network. The accuracy of the patient localization and tracking system is then evaluated through experimental tests, to further assess the interplay between the localization engine and the specific configuration of the wireless sensor network.

The remainder of the paper is organized as follows: Section II overviews the related literature on RF-based indoor localization algorithms. In Section III, the localization algorithms used within the LAURA system are presented, whereas Section IV describes the network architecture used to support the localization system.

Experimental numerical results on the dimensioning and performance evaluation are reported in Section V. Section VI concludes the paper and comments on future research directions.

II. Related work

Indoor localization has been widely addressed in the past literature. A first taxonomy of the proposed technological solutions can be defined based on signal types, e.g. infrared, ultrasound and radio frequency (RF). RF-based systems are particularly attractive, due to the widespread availability of existing wireless network infrastructures (e.g. WiFi, GSM) and to the ease of deployment of lightweight ad hoc WSN.

Most of the RF-based systems aim at leveraging existing WiFi (IEEE 802.11) access points (anchor nodes), in order to localize and track mobile WiFi-enabled devices (client nodes) [4], [5], [6], [7]. A similar setup arises in the case of WSN. While some works focus on proprietary networking protocols [8], [9], others exploit the Zigbee (IEEE 802.15.4) standard, specifically designed for low-power devices. The localization accuracy depends on the placement of the anchor nodes and on their density. Typically, an average localization error of just a few meters is observed in real experiments.

Regardless of the underlying network infrastructure, localization is performed adopting similar processing methods. The location of the anchor nodes is assumed to be fixed and known. Then, RSS measurements are collected between pairs of nodes, since they are related to the inter-node distances. Indeed, off-the-shelf devices provide a RSS indicator (RSSI) without the need of dedicated hardware.

Based on that, the systems described in [5], [6], [10], [8], [9] adopt a simple non-parametric method known as fingerprinting, which avoids the explicit modeling of the RSSI-distance relationship. During a training phase, a database of fingerprints is collected. Each fingerprint consists of a vector, whose elements are the RSSI values measured between a client node at a specific location and the anchor nodes. A client node forms a vector of RSSI measurements between itself and each anchor node and it compares this vector with the pre-computed fingerprints in the database. Then its position is estimated based on the location of the most similar fingerprints. Fingerprinting methods suffer from a non-negligible setup cost, related to the construction of the fingerprint database, which requires a human operator. In addition, they are not robust in face of changes of the environment. Indeed, people, doors and furniture affect RF propagation and pre-computed fingerprints might be no longer valid.

Parametric methods explicitly assume the knowledge of a propagation model that relates the RSSI metrics $\mathbf{S}$ (expressed in dBm units) and the inter-node distance $\mathbf{d}$. A simple dependency is described by the following model [11]

$$\mathbf{S} = \mathbf{S}_0 - 10\alpha \log_{10} \mathbf{d}/\mathbf{d}_0 + \nu, \tag{1}$$

where $\mathbf{S}_0$ is the RSSI metrics measured between two nodes $\mathbf{d}_0$ meters apart. The parameter α represents the power decay index (also known as path loss exponent) and is in the range $[2, 4]$ for indoor environments. The noise term ν is typically modeled as a Gaussian random variable $N(0, \sigma_\nu^2)$ representing shadow-fading effects in complex multipath environments, whereas the value of standard deviation σ_ν depends on the characteristics of the specific environment. The parametric model (1) is leveraged, for example, in [12], [13], [14], [15]. The localization problem is solved by computing the position that maximizes the likelihood with respect to the model in (1). In [15] the model in (1) is extended to take into account the signal attenuation due to walls. Model parameters are estimated solely based on RSSI measurements between anchor nodes, in order to limit the system setup cost. A similar approach is pursued in [7]. There, the propagation model in (1) is implicitly assumed, but the distance between a client and an anchor node is modeled as a linear combination of the RSSI measurements between the client and all the anchors. The RSSI-to-distance linear mapping is estimated solely based on measurements collected by the anchors, thus enabling a zero-configuration setup.

Due to the noise that affects RSSI measurements, temporal averaging is applied over a time window. The length of the window enables to define a trade-off between the accuracy of the RSSI measurement and the system delay. As a rule of thumb, 3-5 RSSI measurements between pairs of nodes are used, each one collected every 200ms.

In case of moving targets, tracking the position of the client node over time improves the quality of localization. To this end, particle filtering [5], [6] enables to merge the RSSI measurements together with a dynamic model that takes into account the typical movement patterns of humans in indoor environments. If the map of the environment is known, particle filtering (PF) can incorporate this a-priori information, thus avoiding wall crossing [5].

III. Localization algorithm

The PLTS module of *LAURA* leverages the localization algorithm presented in [7], which is briefly illustrated in Section III-A. Then, Section III-B describes the particle filter used to enhance the localization accuracy.

A. Zero-configuration indoor localization

Let $\mathbf{x}_i \in \mathbb{R}^2$, $i = 1, \dots, N$ denote the position of an anchor node. Each anchor node obtains a vector $\mathbf{s}_i = [s_{i1}, \cdots, s_{iN}]^T$ of RSSI measurements exchanging data with the other nodes, where s_{ij} is the RSSI relative to the signal emitted by the jth anchor node. The overall RSSI measurements among all anchor node pairs can be represented by an $N \times N$ matrix $\mathbf{S}$

$$\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N]. \tag{2}$$

Similarly, we define the distance vector $\mathbf{d}_i = [d_{i1}, \cdots, d_{iN}]^T$, and the corresponding matrix $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_N]$, where d_{ij} is the Euclidean distance between anchors i and j, i.e. $\|x_i - x_j\|_2$.

Note that $\mathbf{D}$ is symmetric and has zero diagonal entries. Conversely, $\mathbf{S}$ is generally not symmetric, due to asymmetries in radio links. Diagonal entries of $\mathbf{S}$ contain the self-RSSI values. These are the only parameters that need to be manually obtained during the system calibration phase, as they depend on the specific hardware.

The method in [7] postulates a model that describes a linear relationship between the RSSI measurements and the logarithm of the inter-node distances, i.e.

$$\log(\mathbf{D}) = \mathbf{T}\mathbf{S}, \tag{3}$$

where $\mathbf{T}$ is a $N \times N$ matrix defining the signal-to-distance mapping. As such, each row vector $\log(\mathbf{d}_i^T)$ is represented as a linear combination of the columns of $\mathbf{S}$, weighted by the elements of the i-th row $\mathbf{t}_i^T$. Given the measurements between pairs of anchor nodes, i.e. $\mathbf{D}$ and $\mathbf{S}$, the matrix $\mathbf{T}$ is estimated by means of least squares as

$$\mathbf{T} = \log(\mathbf{D})\mathbf{S}^T(\mathbf{S}\mathbf{S}^T)^{-1}. \tag{4}$$

In order to improve robustness to measurement noise, the pseudo-inverse in (4) can be computed using the truncated SVD of $\mathbf{S}$.

Once the signal-to-distance mapping is determined, the localization of a client node proceeds collecting RSSI measurements between itself and its neighboring anchor nodes in the vector $\hat{\mathbf{s}}$. Then, the corresponding distance vector $\hat{\mathbf{d}}$ can be computed as $\hat{\mathbf{d}} = \exp(\mathbf{T}\hat{\mathbf{s}})$.

In [7], a gradient descent algorithm is employed to estimate the location based on the obtained distance vector $\hat{\mathbf{d}}$, and it is employed in our system to initialize tracking.

B. Improving localization with particle filtering

In static conditions and with fixed client nodes, the localization algorithm described above performs fairly well, as robust RSSI measurements can be obtained by means of time averaging. Conversely, in case of moving client nodes, severe short-time RSSI fluctuations are observed, especially when the node is attached to the patient's body. As such, the accuracy of the estimated location decreases, and artificial motion discontinuities are detected.

Incorporating a-priori knowledge about the moving node and the geometry of the environment might help improving the localization accuracy. In our system, we employ a PF to track the position of the client node over time. At each time instant t, the state of the node is represented by the vector $\mathbf{z}(t) = [\mathbf{x}(t)^T, \mathbf{v}(t)^T]^T$, where $\mathbf{x}(t) \in \mathbb{R}^2$ indicates the position of the node and $\mathbf{v}(t) \in \mathbb{R}^2$ its velocity. PF estimates the a-posteriori probability density function of the state $\mathbf{z}$ at time t, given the sequence of previous states and all available observations. Such density function is represented in non-parametric form by means of a set of particles $p = 1, \dots, M$, each associated to a state vector $\mathbf{z}_p(t) = [\mathbf{x}_p(t)^T, \mathbf{v}_p(t)^T]^T$ and a weight $\omega_p(t)$. Thus, we compute a point-wise estimate of the position and velocity of the client node as follows:

$$\begin{cases} \hat{\mathbf{x}}(t) &= (1-\alpha_x)(\hat{\mathbf{x}}(t-1) + \hat{\mathbf{v}}(t-1)\Delta_T) + \alpha_x \sum \omega_p(t)\hat{\mathbf{x}}_p(t) \\ \hat{\mathbf{v}}(t) &= (1-\alpha_v)\hat{\mathbf{v}}(t-1) + \alpha_v(\hat{\mathbf{x}}(t) - \hat{\mathbf{x}}(t-1)) \end{cases} \tag{5}$$

where Δ_T is the sampling period and α_x, α_v are the adaptation rates, which are tuned experimentally based on the node dynamics.

For the problem at hand, PF alternates the execution of two steps: prediction and update. Later we describe how the system is initialized.

1) Prediction: At each step, we compute the new state vector for the p-th particle. To this end, we employ the following dynamic model constructed upon kinematic equations

$$\begin{cases} \mathbf{x}_p(t) &= \mathbf{x}_p(t-1) + \mathbf{v}_p(t-1)\Delta_T + \xi_x \\ \mathbf{v}_p(t) &= \mathbf{v}_p(t-1) + \xi_v \end{cases}, \tag{6}$$

where ξ_x and ξ_v are samples drawn from zero-mean Gaussian random variables with variance, respectively, $\sigma^2_{\xi_x}$ and $\sigma^2_{\xi_v}$, which provide the system with a diversity of hypotheses. If the geometry of the environment is known, it can be incorporated in the dynamic model, e.g. by assigning $\omega_p = 0$ to those particles that crossed a wall.

2) Update: After the prediction step, the weight of the p-th particle is updated based on the RSSI measurements collected at time t. First, we obtain the vector $\hat{\mathbf{d}}$ as described in Section III. Then, we assign a weight computed as

$$\omega_p(t) = \omega_p(t-1)\exp\left(-\frac{1}{2}\sum_{i=1}^{N}\omega_i(\|\mathbf{x}_p(t) - \mathbf{x}_i\|_2 - \hat{d}_i)^2\right), \tag{7}$$

considering the difference between the estimated distance $\hat{d}_i$, and the current distance of the particle from the i-th anchor, $\|\mathbf{x}_p(t) - \mathbf{x}_i\|_2$.

Then, we check the weight distribution of the particles in order to avoid critical situations, which arise, for example, when all particles are trapped within a room while the client node had moved outside. This is done by verifying:

$$\sum_{p=1}^{M}\omega_p(t) > \tau_1 \tag{8}$$

where τ_1 is a threshold value that we set equal to 10^{-5} in our experiments. If the test fails, particles are not correctly tracking the client node. Thus, the PF is re-initialized as described below. Otherwise, the particle weights are normalized such that $\sum_{p=1}^{M}\omega_p(t) = 1$ and the position of the client node is determined based on (5).

As customary when working with PFs, a resampling step is performed if the weight distribution is severely skewed, such that there are just a few particles with non-negligible weight. Here, we adopted the SIR (sequential importance resampling) algorithm. Resampling is performed if the following condition is verified:

$$\frac{1}{\sum_{p=1}^{M}\omega_p(t)^2} < \tau_2 \tag{9}$$

where τ_2 is equal to $M/2$ in our experiments.

3) Initialization: At system startup, when the first set of RSSI measurements are collected for a client node, its location $\hat{\mathbf{x}}(0)$ is computed based on the estimated vector $\hat{\mathbf{d}}$. To this end, we use a gradient descent method that minimizes the following cost function:

$$\hat{\mathbf{x}}(0) = \arg\min_{\mathbf{x}} \frac{1}{2}\sum_{i=1}^{N}\theta_i\left(\|\mathbf{x} - \mathbf{x}_i\|_2 - \hat{d}_i\right)^2 \tag{10}$$

where $\hat{d}_i$ is the estimated distance between the client node and the i-th anchor and θ_i is a weight factor computed as:

$$\theta_i = \frac{\hat{d}_i^{-2}}{\sum_{i=1}^{N}\hat{d}_i^{-2}} \tag{11}$$

By differentiating equation (10) with respect to $\mathbf{x}$ we obtain an iterative equation used to update the estimate of $\hat{\mathbf{x}}$:

$$\hat{\mathbf{x}}^{(k+1)} = \hat{\mathbf{x}}^{(k)} + \alpha\sum_{i=1}^{N}\theta_i\left(1 - \frac{\hat{d}_i}{\|\hat{\mathbf{x}}^{(k)} - \mathbf{x}_i\|_2}\right)(\hat{\mathbf{x}}^{(k)} - \mathbf{x}_i) \tag{12}$$

where the parameter α is set to 0.1 and the initial estimate $\hat{\mathbf{x}}^{(0)}$ is equal to the location of the nearest anchor node. Then we initialize the state of each particle $\mathbf{z}_p(0) = [\mathbf{x}_p(0)^T, 0, 0]^T$, $\forall p$.

Since the initialization of the particle filter might potentially locate the target outside the building, we leverage the knowledge of the geometry of the environment to constrain the initialization point to lie inside the building.

IV. Network Architecture and Implementation

The *LAURA* system is composed of: i) anchor nodes, which are part of the infrastructure and are statically deployed in the areas to be monitored; ii) client nodes, which are attached to patients in order to support localization, tracking, and patient supervision services. The sensed information is collected at the control point through a NA.

The network is operated through the *Hierarchical Addressing Tree (HAT)* routing protocol, which creates a tree-like routing/forwarding topology among the network nodes (anchor and client nodes). The routing tree is rooted at a Personal Area Network (PAN) coordinator which collects the traffic of the entire tree. The hierarchical routing tree is created and maintained through dynamic association (de-association) policies, which allow sensors to retrieve (release) network addresses and join (leave) the routing tree. Figure 2 shows an example of address format and management.

Upon activation, a sensor node starts in the *Initialization* state. After having initialized all the internal components and variables, the node switches to the *Scanning* state and starts collecting beacons sent by the surrounding nodes. Then, the node, hereafter denoted associating node, chooses the parent node to be associated to among the elements of a set $\mathcal{P}$, which includes the nodes that sent a received beacon message. Association proceeds by maximizing a utility function that depends on three factors: i) Received Signal

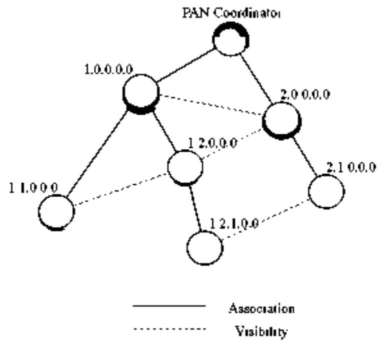

Fig. 2. Example of Address Format and Tree Topology.

Strength Indicator ($RSSI_i$); ii) current number of children of the i-th candidate parent (N_i) and iii) distance of the candidate parent from the PAN coordinator (Hop Count, HC_i). Namely, the associating node chooses the parent node i^* according to the following equation:

$$i^* = \arg.\max_{i \in \mathcal{P}} (\alpha RSSI_i + \beta N_i + \gamma HC_i)$$

The RSSI gives an estimate of the quality of the link and it is used to prevent nodes from associating with a parent through unstable links. The last two factors impact on the target network topology. Intuitively, the larger is the weight β (γ), the deeper (wider) is the resulting tree. The weights of the three factors have been empirically set to $\alpha = 1$ $\beta = -3$, and $\gamma = -6$.

Upon selection, the associating node sends an explicit association request to the selected parent node, which, in turn, responds with a message containing the proposed network address. Finally, the associating node sends back a confirmation message to the parent and switches to the *Associated With Network* (AWN) state.

In the AWN state, the association is maintained by means of periodic beacon messages which carry all the information needed to manage and maintain the association. Namely, each beacon message contains a sequence number (that can be used for synchronization purposes), the current route cost (used by requesting nodes to compute cost function), and a CHILD_MASK used to inform associating nodes about the current number of children of the parent.

The NA (routing tree, association/re-association policies, beaconing) is used to implement and support *LAURA* services. The localization algorithm described in Section III requires the dynamic construction and maintenance of the RSSI matrix among anchor nodes ($\mathbf{S}$) as well as the collection of the RSSI samples measured at the client nodes ($\hat{\mathbf{s}}$). The entries of $\hat{\mathbf{s}}$ change over time due to sensor node mobility (patients roaming around the premises of the host institution). Although anchor nodes are fixed, the entries of $\mathbf{S}$ need to be periodically updated, in order to capture changes affecting RF propagation (e.g. multi-path fading, temperature, humidity, people, doors and furniture). To this end, all the elements of $\mathbf{S}$ are initialized to -91 dBm. This value is replaced with the incoming RSSI value s_{ij}^{new} as soon as it becomes available. Then, every element of the matrix is updated according to the following autoregressive model:

$$s_{ij}(t) = (1 - \alpha_s) s_{ij}(t-1) + \alpha_s s_{ij}^{new}, \qquad (13)$$

where α_s is the adaptation rate, which is set to 0.9 in our system.

Thus, the implementation of the patient localization and tracking services requires two functionalities: i) *dynamic collection* of RSSI samples at the sensor nodes (mobile sensors and anchors); ii) *effective delivery* of such collected data to the PAN coordinator, where the localization algorithm is executed.

Dynamic RSSI Collection RSSI measurements among anchor nodes are collected leveraging the periodic beaconing mechanism already employed to maintain the routing tree. Namely, upon reception of a beacon from an anchor, each sensor node (either client or anchor node) extracts the corresponding RSSI sample and stores it locally. Client and anchor nodes maintain first-in/first-out buffers to store consecutive RSSI samples. Client (anchor) nodes store the last 3 (5) RSSI samples received from each anchor.

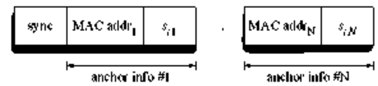

Fig. 3. Payload Format of the localization information packets. A sequence number is attached to the packet for synchronization purposes.

The beaconing period is a central parameter that determines the effectiveness of the *LAURA* system. Intuitively, the shorter the beaconing period the lower is the system delay in providing the estimated location of the client node. On the other hand, a short beaconing period may lead to traffic congestion. Thus, a trade-off needs to be sought. In the following, we have considered a beaconing period equal to 200 ms.

Localization Information Delivery Periodically, sensor nodes calculate the median of the last RSSI samples available in the buffer corresponding to each anchor. Then, client (anchor) nodes sort the median values and send to the PAN coordinator the top 5 (12) values, together with the corresponding anchor identifier. To accomplish such transfer, sensor nodes leverage multi-hop routing-specific packets whose generic payload is reported in Figure 3. We decided to include in the packet only RSSI median values greater than -87 dBm according to [16]. Finally, to cope with patients' mobility, a new packet is sent from each client node every 1 s. Conversely, a longer period is used for anchor nodes, which is set equal to 20 s.

V. Performance evaluation

In order to evaluate the performance of the proposed personal localization system in real indoor environments we carried out experiments using a sensor network based on the IEEE 802.15.4 standard. The network architecture described in Section IV consists of MicaZ and IRIS motes operated by the TinyOS. For these devices, the measured self-RSSI value (see Section III-A) was -25 dBm.

In the first experiment, we evaluate the localization error of the algorithm described in Section III-A (i.e. when particle filtering is not enabled), for the case of static nodes. To this end, we deployed several anchor nodes at different positions in a 100 m^2 indoor area, characterized by non line-of-sight propagation due to the presence of walls and furniture. A client node was used to take RSSI measurements at different test points on a 90 cm x 90 cm grid. The node was placed at each test point for 30 seconds, so that 30 RSSI packets (following the format depicted in Figure 3) were sent to the PAN coordinator. For every RSSI packet, the position of the node is estimated using the algorithm described in Section III-A. The test was repeated for different anchor node densities. For each target node density, we repeated the estimation by selecting a random subset of anchor nodes, in order to eliminate the bias related to a particular anchor node deployment. Figure 4 shows the cumulative density function (CDF) of the localization error for different values of the anchor node density. We observe that, for an anchor density of 0.15 nodes/m^2, in 80% of cases, the localization error is below 2.5 m. The localization accuracy is still acceptable for the considered application even when the density is below 0.1 nodes/m^2

In the second experiment we evaluated the performance of the complete localization system described in Section III, which inte-

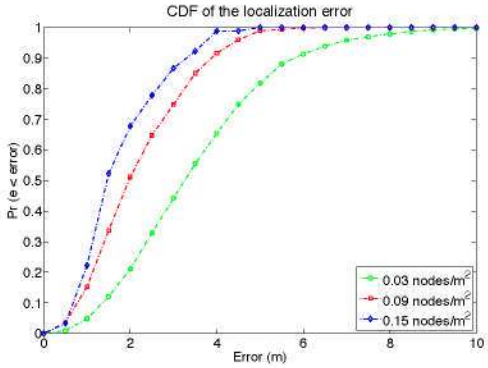

Fig. 4. Cumulative density function of the localization error in a test area of approximately 100 m^2.

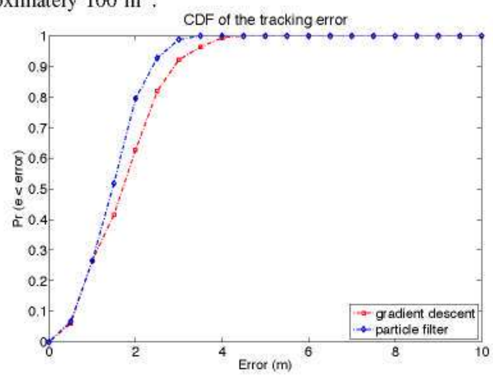

Fig. 5. Cumulative density function of the tracking error in a test area of about 250 m^2.

grates particle filtering. In this case, we tracked a client node carried by a person that followed a known path in an indoor area of around 250 m^2. The path was sampled each 90 cm so that it was possible to synchronize the RSSI message reception with the actual position of the mobile node. For every RSSI packet, the location of the node is estimated using the algorithm described in Section III. Figure 5 shows the cumulative density function of the localization error for an anchor node density of 0.08 nodes/m^2. We note that when the particle filter is disabled, an error below 2.5 m is achieved in 80% of the cases. Enabling the particle filter improves the localization accuracy of about 0.5 m.

Similarly to the case of static sources, we studied the impact of the anchor node density on the tracking accuracy. As before, for each target density value, several combinations of anchor nodes were selected at random and results averaged to reduce the bias of individual deployments. Figure 6 shows the averaged CDFs for four different values of the anchor node density. Note that, even at density value as low as 0.04 nodes/m^2 (i.e. one anchor node each 25 m^2), the localization error is acceptable (less than 3.5 m in 80% of the cases). The latter result is encouraging for practical applications, as the number of anchor nodes has a significant impact on the traffic load, the energy consumption and the total cost of the system.

VI. Conclusions

We described the personal localization and tracking system of *LAURA* project. The system can be rapidly deployed in any indoor environment, due to adopted self-calibration method. The experimental evaluation carried out in real environments has shown that a localization accuracy of about 2-3m can be achieved also with a relatively sparse deployment of the anchor nodes (approx 0.15 nodes/m^2). To enable reproducible research, the measurements collected in our tests are publicly available for download at http://laura.como.polimi.it,

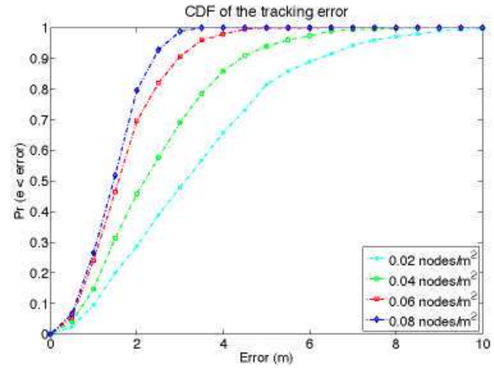

Fig. 6. Cumulative density function of the tracking error for different values of the anchor node density in a test area of about 250 m^2.

together with the Matlab scripts implementing the localization algorithms algorithms presented in this paper. Future work will address the integration of an inertial navigation system and the study of energy-aware localization algorithms.

References

[1] W. Mann, "The aging population and its needs," *IEEE Pervasive Computing*, vol. 3, no. 2, pp. 12–14, 2004.

[2] J. Ko, T. Gao, R. Rothman, and A. Terzis, "Wireless sensing systems in clinical environments: Improving the efficiency of the patient monitoring process," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 29, no. 2, pp. 103 –109, march-april 2010.

[3] Y. Ren, R. Pazzi, and A. Boukerche, "Monitoring patients via a secure and mobile healthcare system," *Wireless Communications, IEEE*, vol. 17, no. 1, pp. 59 –65, february 2010.

[4] *Bayesian indoor positioning systems*, vol. 2, 2005. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1498348

[5] F. Evennou and F. Marx, "Advanced integration of wifi and inertial navigation systems for indoor mobile positioning," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 164–164, uary.

[6] H. Wang, H. Lenz, A. Szabo, J. Bamberger, and U. Hanebeck, "Wlan-based pedestrian tracking using particle filters and low-cost mems sensors," march 2007, pp. 1 –7.

[7] H. Lim, L.-C. Kung, J. C. Hou, and H. Luo, "Zero-configuration indoor localization over ieee 802.11 wireless infrastructure," *Wireless Networks*, vol. 16, no. 2, pp. 405–420, 2010.

[8] K. Lorincz and M. Welsh, "Motetrack: A robust, decentralized approach to rf-based location tracking." in *LoCA*, ser. Lecture Notes in Computer Science, vol. 3479. Springer, 2005, pp. 63–82.

[9] M. D'Souza, T. Wark, and M. Ros, "Wireless localisation network for patient tracking," dec. 2008, pp. 79 –84.

[10] S.-Y. Lau, T.-H. Lin, T.-Y. Huang, I.-H. Ng, and P. Huang, "A measurement study of zigbee-based indoor localization systems under rf interference," in *WINTECH '09: Proceedings of the 4th ACM international workshop on Experimental evaluation and characterization*. New York, NY, USA: ACM, 2009, pp. 35–42.

[11] N. Patwari, J. Ash, S. Kyperountas, A. Hero III, R. Moses, and N. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, 2005.

[12] X. Li, "Collaborative localization with received-signal strength in wireless sensor networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3807 –3817, nov. 2007.

[13] M. Sugano, T. Kawazoe, Y. Ohta, and M. Murata, "Indoor localization system using rssi measurement of wireless sensor network based on zigbee standard," in *Wireless and Optical Communications*. IASTED/ACTA Press, 2006, pp. 1–6.

[14] R. Bader, M. Pinto, F. Spenrath, P. Wollmann, and F. Kargl, "BigNurse: A Wireless Ad Hoc Network for Patient Monitoring," in *Pervasive Health, First Workshop on Location Based Services for Health Care (Locare'06)*, IEEE. Innsbruck, Austria: IEEE, 11 2006.

[15] P. Barsocchi, S. Lenzi, S. Chessa, and G. Giunta, "Virtual calibration for rssi-based indoor localization with ieee 802.15.4," jun. 2009, pp. 1–5.

[16] K. Srinivasan and P. Levis, "Rssi is under appreciated," in *In Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets*, 2006.