Ad Hoc Networks 11 (2013) 39–53

Contents lists available at SciVerse ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

An integrated system based on wireless sensor networks for patient monitoring, localization and tracking

Alessandro Redondi, Marco Chirico, Luca Borsani, Matteo Cesana *, Marco Tagliasacchi

Dipartimento di Elettronica e Informazione, Politecnico di Milano, P.zza Leonardo da Vinci, 32, Milano, Italy

ARTICLE INFO

Article history:
Received 28 November 2011
Received in revised form 23 March 2012
Accepted 25 April 2012
Available online 4 May 2012

Keywords:
Indoor localization
E-health
Wireless sensor network

ABSTRACT

This works describes the system *LAURA* which provides patient localization, tracking and monitoring services within nursing institutes through a wireless sensor network. The system is composed of three functional blocks: a localization and tracking engine which performs localization out of samples of the received signal strength and tracking through a particle filter; a personal monitoring module based on bi-axial accelerometers which classifies the movements of the patients eventually detecting hazardous situations, and a wireless communication infrastructure to deliver the information remotely. The paper comments on the design and dimensioning of the building blocks. Two approaches are proposed to the implementation of the localization and tracking engine: a centralized implementation where localization is executed centrally out of information collected locally, and a distributed solution where the localization is performed at the mobile nodes and the outcome is delivered to the central controller. Strengths and weaknesses of the two solutions are highlighted from a system's perspective in terms of localization accuracy, energy efficiency and traffic loads. *LAURA* modules are finally tested in a real environment using commercial hardware. The main outcomes are an average localization error lower than 2 m in 80% of the cases and a movements classification accuracy as high as 90%.

1. Introduction

The improvement of health care systems and infrastructures is one of the most challenging and compelling goals of today's society. In fact, it is commonly recognized that the combined effects of population aging and nursing staff shortage may eventually lead current health care systems to collapse [1].

The most relevant drawback of current models for patient monitoring, care, management and supervision, stems from the fact that the required operations are often manually executed by the nursing staff, which constitutes *de facto* an efficiency bottleneck.

Recent achievements in the broad fields of the Information and Communication Technologies (ICT) and, in particular, in the field of device miniaturization and wireless communication technologies have opened up enormous opportunities to design ubiquitous and pervasive systems for the support the improvement of health-care-related processes. The specific applications scenarios where ICT is being adopted include the automatic and remote monitoring/tracking of patients in hospitals, and nursing institutes, the implementation of tele-medicine services, the supervision of fragile, elderly people in their own domestic environment through automatic systems to handle remote drug delivery.

Several initiatives within the field of E-Health have been launched/stimulated in the last decade by private organizations (hospitals, pharmaceutical companies, etc.) [2], public institutions and governments, and researchers worldwide [3]. As an example, the European Commission has recently launched a thematic portal to act as focal point for the research activities in the field, and, further, a specific challenge (ICT Challenge 5) has been promoted

* Corresponding author. Tel.: +39 02 2399 3695; fax: +39 02 2399 3413.
E-mail address: cesana@elet.polimi.it (M. Cesana).

1570-8705/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.adhoc.2012.04.006

within the Framework Programme 7 for the development of sustainable and personalized health-care through integrated systems.

This work describes the design, implementation and optimization of a lightweight system based on Wireless Sensor Networks (WSNs) for the automatic supervision of patients within nursing institutes. In particular, the work has been stimulated by a 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, which is specialized in the assistance of fragile people with a broad range of pathologies, including cognitive and/or perceptual disorders, Down's syndrome, epilepsy, etc. The implemented system named LAURA (Localization and Ubiquitous Monitoring of Patients for Health Care Support) provides a general service of patient's supervision which includes two major functionalities:

- *Patient localization and tracking:* The exact knowledge of the location of patients is a valuable asset, since it enables prompt reaction in case urgent assistance is needed.
- *Patient's status monitoring:* The current 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.), possibly implementing automatic detection of anomalous changes in such parameters.

The *LAURA* system implements the aforementioned services through a Wireless Sensor Network (WSN) deployed throughout the nursing institute premises to collect the required data from the patients and deliver the very same data remotely to a central controller. Referring to Fig. 1, the main architectural building blocks of *LAURA* system are a Personal Localization and Tracking System (PLTS), a Personal Monitoring System (PMS), and a lightweight and flexible Network Architecture (NA) composed of fixed relaying nodes (anchor nodes) and portable devices (client nodes) mounted of patients to deliver the information related to localization and monitoring to an automatic central controller.

The PLTS and the PMS functional modules leverage wearable devices mounted on the patients and the network architecture to implement the localization/tracking and monitoring services. Namely, the PLTS is built around 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. The PMS implements a lightweight movement classifier based on a decision-tree to properly classify the current movement pattern of the patients out of the acceleration samples captured by biaxial accelerometers. The contributions of the present work are twofold: first, we illustrate the design, implementation and optimization of each building block of *LAURA* system; second, we show how these building blocks can be effectively set together and optimized to provide the required services. The performance of each building block and of the system as a whole are then evaluated through simulations and testbed implementation in terms of localization/classification accuracy (for the PLTS and PMS, respectively), overall energy efficiency and required traffic load.

The remainder of the paper is organized as follows: Section 2 overviews the related literature on RF-based indoor localization algorithms and human movement classifiers. In Section 3, the three main building blocks of LAURA system are individually presented, whereas Section 4 describes the LAURA system as a whole further introducing centralized and fully alternatives to implement the PLTS. Section 5 reports the performance evaluation of LAURA systems through simulations and practical testbed implementation. Concluding remarks are given in Section 6.

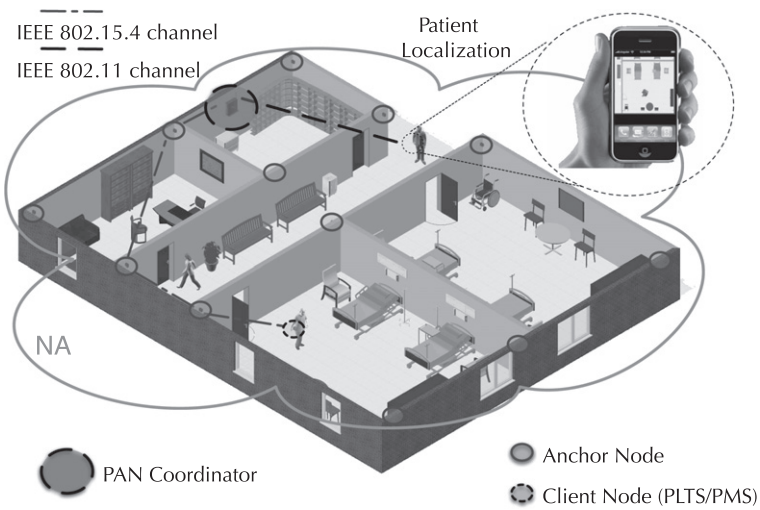


Fig. 1. Reference functional architecture of the system.

2. Related work

This section briefly overviews the related work in the fields of localization algorithms/systems and movement classification through WSNs.

2.1. Localization systems

Indoor localization systems can be broadly classified based on the physical transmission technology used for localization purposes; infrared, ultrasound and radio frequency (RF) are commonly used in commercial localization systems. 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 access points (anchor nodes), in order to localize and track mobile WiFi-enabled devices (client nodes). Regardless of the underlying network infrastructure (IEEE 802.11, IEEE 802.15.4, etc.) 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.

RF-based localization systems use both non-parametric and parametric algorithms. Non-parametric methods such as fingerprinting [4–8] avoid the modeling of the explicit RSSI-distance relationship, but they suffer from a non-negligible setup cost, related to the construction of the fingerprint database, and they are not robust in face of changes of the environment. Conversely, 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 [9]

$$\mathbf{S} = \mathbf{S}_0 - 10\alpha \log_{10} \mathbf{d}/\mathbf{d}_0 + v, \quad (1)$$

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 v is typically modeled as a Gaussian random variable $N(0, \sigma_v^2)$ representing shadow-fading effects in complex multipath environments, whereas the value of standard deviation σ_v depends on the characteristics of the specific environment. The parametric model (1) is leveraged, for example, in [10–13]. The localization problem is solved by computing the position that maximizes the likelihood with respect to the model in (1). A similar approach is pursued in [14], where, the propagation model 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. In case of moving targets, tracking the position of the client node over time improves the quality of localization. To this end, particle filtering [4,5] enables to merge the RSSI measurements together with a dynamic model that takes into account the typical movement patterns of humans in indoor environments.

Focusing on the specific topic of patients monitoring and tracking, various examples can be found in the literature, but the number of implemented solutions is extremely low. In [3] the diversity of factors that take part in the design of reliable, intelligent and secure patient monitoring systems is presented; in [15] WSNs are depicted as a particularly well-suited technology for the task, together with an analysis regarding both their potentials and the multitude of technical challenges arising from their deployment. Indeed, healthcare applications impose a set of stringent requirements in terms of system reliability, quality of service, privacy and security.

A similar analysis is carried out in [2], together with a prototype of MEDiSN, a WSN-based system capable of monitoring vital signs of unattended patients. However, patients localization is only presented as broadly useful, according to the indications of many staff members, depicting this topic as a future feature that can be added as soon as researchers will be able to provide efficient and effective ways to perform it.

A wireless localization network capable of tracking patients in an indoor environment is presented in [8]: a set of fixed nodes is arranged in a grid fashion all around the building, in order to guarantee that every mobile node can be detected by al least three fixed nodes at any time. The frequency of beacons sent by a mobile node and delivered to a central computer is used to estimate the region in which the node is located and its position is approximated through the triangle's centroid. As a consequence, the localization accuracy is proportional to the number of fixed nodes arranged all around the environment.

In [16] the design a location aware WSN for patients localization is discussed and an algorithm called Ranging using Environment and Mobility Adaptive RSSI (REMA) Filter is proposed. The experimentations show that the position estimation error is reduced by 6% for the REMA range error model than that of the CC2431 range error model, however the system is not implemented and consequently not tested at work.

2.2. Human movement classifiers

As for indoor localization, also movement classification has been widely addressed in the past. Most of the existing systems are based on low cost and small size multi-axial accelerometers that respond to both acceleration due to gravity and acceleration due to body movement. This make them suitable for monitoring postural orientation as well as body movements. Several solutions are available to indirectly assessing metabolic energy expenditure [17], to measure various parameters of movement (step rate, postural sway) [18] and in smart personal alarm systems to detect falls [19]. The classification is done either leveraging a single accelerometer strapped on the waist, or using multiple body-worn accelerometers. Different approaches to classification were proposed by researchers, including fixed - threshold classification [20,21], pattern-recognition [22] and fuzzy-logic [23]. These previous studies involved

the processing and analysis of raw acceleration data after transmission to a local computer, where algorithm performed classification off-line. Recent advances in MEMS technologies have made possible to perform such signal processing onboard the wearable unit using embedded intelligence, and to transmit only the final result, thus saving energy for raw data transmission. However, since available low-cost and low-power micro controllers are limited in memory and processing resources, such real-time movement classification algorithms are strictly constrained and using computationally intensive tools such as the Discrete Fourier Transform to discriminate movements would be infeasible [24].

3. LAURA building blocks

The LAURA system is composed of three main building blocks: a localization and tracking engine, a movement classification algorithms and a multi-hop network architecture. In this section, we describe individually each one of these building blocks. Section 4 is then dedicated to the system-wise integration/optimization of these atomic components.

3.1. The patient localization and tracking engine

The PLTS module of *LAURA* builds on the localization approach of [14], briefly illustrated in Section 3.2, which is then enhanced by applying a particle filter to increase the localization accuracy (Section 3.3).

3.2. Zero-configuration indoor localization

The localization algorithm has to estimate a relationships between RSSI samples and the distance between nodes. In order to achieve this, each anchor i (whose given coordinates are $\mathbf{x}_i \in \mathbb{R}^2$) listens to packets sent from other anchors and measures the corresponding signal strength, obtaining a vector $\mathbf{s}_i = [s_{i1}, s_{i2}, \ldots, s_{iN}]^T$, where s_{ij} is the RSSI value relative to the signal emitted by anchor j. Collecting each $\mathbf{s}$ vector, it is possible to represent the RSSI measurements of the whole system in a $N \times N$ matrix $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N]$. Similarly, being d_{ij} the Euclidean distance between anchors i and j, the distance vector $\mathbf{d}_i = [d_{i1}, d_{i2}, \ldots, d_{iN}]^T$ and the corresponding matrix $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_N]$ can be defined.

While $\mathbf{D}$ is symmetric and has zero diagonal entries, $\mathbf{S}$ is generally asymmetric, as radio links are. 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.

As in [14], we postulate a model that describes a linear relationship between the RSSI measurements and the logarithm of the inter-node distances, i.e.

$$\log(\mathbf{D}) = \mathbf{TS}, \tag{2}$$

where $\mathbf{T}$ is a $N \times N$ matrix defining the signal-to-distance mapping. As such, each row vector $\log\left(\mathbf{d}_i^T\right)$ is represented as a linear combination of the columns of $\mathbf{S}$, weighted by the elements of the ith 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

$$\mathbf{T} = \log(\mathbf{D})\mathbf{S}^T(\mathbf{S}\mathbf{S}^T)^{-1}. \tag{3}$$

In order to improve robustness to measurement noise, the pseudo-inverse in (3) 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 $\hat{\mathbf{d}} = \exp(\mathbf{T}\hat{\mathbf{s}})$. A gradient descent algorithm is employed to estimate the location based on the obtained distance vector $\hat{\mathbf{d}}$, which then provides the starting point to initialize the tracking through the particle filter.

3.3. Particle filtering

LAURA PLTS further enhances the localization and tracking accuracy through the use of a particle filtering to "smooth down" the fluctuations of the RSSI sample. This method represents the state of a node at instant t as a vector $\mathbf{z}(t) = [\mathbf{x}(t)^T, \mathbf{v}(t)^T]^T$, being $\mathbf{x}(t) \in \mathbb{R}^2$ its position and $\mathbf{v}(t) \in \mathbb{R}^2$ its velocity. Given the sequence of previous states and all available observations, PF estimates the *a posteriori* probability density function of the state $\mathbf{z}$ at time t, represented in non-parametric form by means of a set of particles $p = 1, \ldots, M$ each associated to a state vector $\mathbf{z}_p(t) = [\mathbf{x}_p(t)^T, \mathbf{v}_p(t)^T]^T$ and a weight $\omega_p(t)$. The state of the target node is then point-wise estimated as:

$$\begin{cases} \hat{\mathbf{x}}(t) = (1-\alpha_x)(\hat{\mathbf{x}}(t-1) + \hat{\mathbf{v}}(t-1)\Delta_T) + \alpha_x \sum \omega_p(t)\hat{\mathbf{x}}(t), \\ \hat{\mathbf{v}}(t) = (1-\alpha_v)\hat{\mathbf{v}}(t-1) + \alpha_v(\hat{\mathbf{x}}(t) - \hat{\mathbf{x}1}(t-1)), \end{cases} \tag{4}$$

where Δ_T is the sampling period and α_x, α_v are the adaption rates, experimentally tuned based on the node dynamics. The execution of PF consists of two alternating steps, taking place after the system initialization: prediction and update.

3.3.1. Prediction

The prediction phase computes the new state vector for the pth particle, following the dynamic model built upon the kinematic equations

$$\begin{cases} \mathbf{x}_p(t) = \mathbf{x}_p(t-1) + \mathbf{v}_p(t-1)\Delta_T + \xi_x, \\ \mathbf{v}_p(t) = \mathbf{v}_p(t-1) + \xi_v, \end{cases} \tag{5}$$

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.

3.3.2. Update

The update phase, performed after the prediction one, updates the weight of the pth particle according to the RSSI measurements collected at time t by computing:

$$\omega_p(t) = \omega_p(t-1)\exp\left(-\frac{1}{2}\sum_{i=1}^{N}\omega_i(\|\mathbf{x}_p(t)-\mathbf{x}_i\|_2-\hat{d}_i)^2\right), \tag{6}$$

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

The next step checks the weight distribution of the particles in order to avoid critical situations, which can arise, for example, when all particles are trapped within a room while the target node had moved outside. This is done by verifying:

$$\sum_{p=1}^{M}\omega_p(t) > \tau_1, \tag{7}$$

being τ_1 a threshold value set as 10^{-5} in the experiments. A failed test implies that the particles are not correctly tracking the target, thus the PF is re-initialized. Otherwise, a normalization is performed, such that

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

and the target node's position is determined based on (4). 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.

3.3.3. Initialization

The initialization of the system, performed at startup, computes the location of a target node $\hat{\mathbf{x}}(0)$ using the estimated vector $\hat{\mathbf{d}}$ via a gradient descent method which minimizes the following cost function:

$$\hat{\mathbf{x}}(0) = \underset{\mathbf{x}}{\operatorname{argmin}}\frac{1}{2}\sum_{i=1}^{N}\theta_i(\|\mathbf{x}-\mathbf{x}_i\|_2-\hat{d}_i)^2, \tag{10}$$

where $\hat{d}_i$ is the estimated distance between the target node and the ith 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 (10) with respect to $\mathbf{x}$ it is possible to obtain an iterative equation used to update the estimated $\hat{\mathbf{x}}$:

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

where 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, $\mathbf{z}_p(0) = [\mathbf{x}_p(0)^T, 0, 0]^T$ is set $\forall p$. 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.

3.4. The personal monitoring algorithm

The PMS module of *LAURA* is based on a hierarchical tree-based decision algorithm, where the classification output is refined at each level of the tree: broad classifications are made in the top levels with low-effort threshold decisions, while more detailed classifications are made in the lower levels. Decision-tree-based classifiers have been proven to be flexible yet accurate enough to be implemented on low cost and resource-constrained hardware [25].

With reference to Fig. 2a the algorithm uses the signals coming from a single biaxial accelerometer strapped on the waist of the patient (see Fig. 2b) to perform movement classification. To cope with the limited amount of memory available on the target nodes while ensuring good classification results, the following design parameters have been selected:

- Sampling frequency: previous studies on human body movements (e.g., [26]) show that the 99% of the energy of such movements are contained within frequency component below 15 Hz. Each axis of the accelerometer is hence sampled at 30 Hz.
- Classification window length: we adopted a window-by-window classification scheme, where movement is classified analyzing the data collected in a 1-s window, which is also presented as optimal in [25].

For each classification window, the following steps are performed:

3.4.1. Signal normalization and filtering

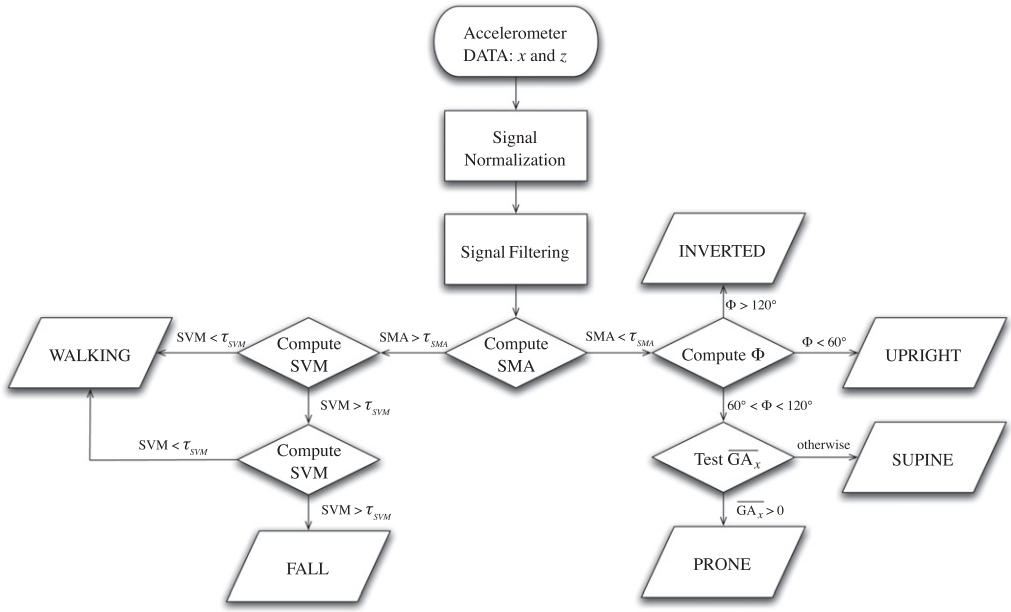
Data coming from the accelerometer is first normalized such as 0 g corresponds to 0 in output and 1 g corresponds to 1. The normalized data is then filtered with a second-order elliptical infinite impulse response (IIR) filter with low-cut frequency at 0.25 Hz, in order to separate the acceleration component due to gravity (GA) and the one due to body movement (BA). These two components are fused in the original signal, and since they overlap both in time and frequency, their separation is not trivial. However, as shown in [24], low pass filtering allows to approximate the two components. The GA component is approximated as the output of the filtering operation, while the BA component is the difference between the original signal and the GA component. The BA component is used to discriminate activity from rest while the GA to infer the postural orientation of the subject.

3.4.2. Activity/rest test

Once the two acceleration components have been separated, the normalized Signal Magnitude Area *SMA* is computed for the BA signal. Such measure can be interpreted as an approximation of the energy expenditure due to physical activity and it is defined as:

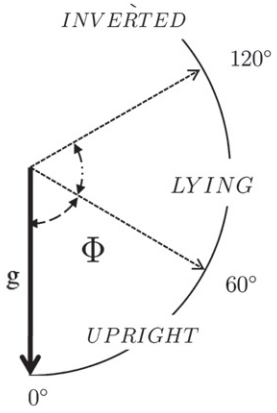
$$SMA = \frac{1}{N}\left(\sum_{i=1}^{N}|BA_x(i)| + \sum_{i=1}^{N}|BA_z(i)|\right), \tag{13}$$

where BA_x and BA_z are the x and z components of the acceleration due to body movement and N is the length



(a)

(b)



(c)

Fig. 2. (a) The block diagram of the classification algorithm. Simple decisions are made at the top of the tree, while more detailed processing is made in the lower levels. (b) The classification algorithm leverages a single bi-axial accelerometer strapped on the waist of a patient. (c) Φ is the angle between the z-component of the gravitational part of the acceleration and the gravitational vector g. It is used to estimate the tilt of the user.

in samples of the observation window. Since the energy expenditure during movement is much greater than during rest the *SMA* value is checked against a fixed threshold τ_{SMA} (equal to 0.2 in our tests) to determine between situations of movement/rest.

3.4.3. Postural orientation

If the *SMA* test results correspond to a static situation, the postural orientation of the patient is estimated leveraging the acceleration due to gravity (GA). First, the z-component of the GA is averaged over the observation window, then the user's tilt angle Φ is computed as:

$$\Phi = \arccos(\overline{GA_z}). \tag{14}$$

With reference to Fig. 2c, the postural orientation is then assigned as following:

$$OUT = \begin{cases} UPRIGHT & \text{if } 0^\circ < \Phi < 60^\circ, \\ LYING & \text{if } 60^\circ < \Phi < 120^\circ, \\ INVERTED & \text{if } \Phi > 120^\circ, \end{cases} \tag{15}$$

where *OUT* is the classification output, and the *INVERTED* status occurs if the accelerometer is strapped in a wrong way (e.g with the z axes pointing upwards). If the

classification output is *LYING* we can further process the x component of the gravitational acceleration:

$$OUT = \begin{cases} PRONE & \text{if } \overline{GA_x} > 0, \\ SUPINE & \text{otherwise}, \end{cases} \quad (16)$$

where $\overline{GA_x}$ is the average of the x-component of the GA over the observation window.

3.4.4. Fall detection

If the *SMA* test determines activity, the BA signal is further analyzed. At this stage of implementation, we are interested only in the detection of possible falls; it is however possible to extract other information about the user's movement (walking, running or other activities). In our implementation we just assume that if the patient is upright and active, then she/he is walking. During normal daily activities, the body movement acceleration have a magnitude lower than 1g [27], except for running, which is not the case of a patient in a hospital. That means that magnitude values far greater than normal accelerations are suspect and may indicate a critical event such as a fall. Thus, for the fall detection algorithm, we check for two consecutive samples in the observation window such that:

$$SVM(i) = \sqrt{BA_x(i)^2 + BA_z(i)^2} > \tau_{SVM}, \quad (17)$$

where *SVM* is the Signal Magnitude Vector for sample i and τ_{SVM} is equal to 1.2g in our implementation (see Section 5.4 for further details). To improve the performance of the fall detection algorithm a fall alarm is issued only if the patient's status remains inactive and lying down for 10s after condition (17) is verified. Otherwise the fall event is considered as recovered.

3.5. Network architecture

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 network architecture.

In previous work [28,29], we propose a 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, further enabling uplink/downlink connectivity to mobile sensor nodes.

However, such tree-based routing mechanism is over dimensioned for the reference application scenarios, which mainly feature convergecast (uplink only) traffic with data flowing from the mobile nodes to the PAN coordinator. A new mechanism is hence needed, in order to (i) lighten the routing procedures, providing a simpler management of both target and anchor nodes, (ii) reduce packets overhead to manage the routing functionalities and (iii) simplify the debugging of the network layer related code.

Building on this observation, LAURA system adopts a classical distance vector routing protocol that provides reliable paths, keeping the set up and maintenance of the routing topology as lightweight as possible. The protocol leverages periodic beaconing among anchor nodes which is also used by the PLTS to enable inter-node RSSi measurements. Additionally every anchor node maintains a single-line routing table featuring: (i) the distance in hops from the root node and (ii) the ID of the next hop on the path towards the root node. When receiving a beacon from the PAN, each anchor node sets its own distance to 1 and the PAN as next hop. This distance is then advertised to the neighbors through the beacons. As the beacon transmission goes on, each anchor node chooses as next-hop to the PAN coordinator the node corresponding to the shortest path. In case two beacons are received with the same distance, the one with the highest RSSi is selected. Fig. 3 shows an example of a 6-nodes WSN in which nodes 5 choose the best next hop between all the nodes from which it receives beacons from. Obsolete entries pointing to no more active next hops are flushed from the routing tables if a flush timer T_{flush} expires before receiving beacons from the reference next hop. Mobile nodes do not participate in the beaconing transmission process, but only listen to beacons to choose the best next hop to reach the root node. The periodic beaconing and the flush timer are also used to enforce connectivity check onto the network topology. Alarm messages are sent out to the central controller whenever an anchor node does not receive any beacons from one of the known neighbors for a given amount of time, and whenever an entry of the routing table is flushed.

4. LAURA system design: distributed vs. centralized approaches

This section comments on the design and optimization of the LAURA system as a whole. In detail, we introduce two alternatives on how the building blocks described in Section 3 can be integrated to provide the localization and tracking service. A centralized and a distributed solution are presented which differ on "where" the localization engine is placed into the LAURA system; namely, in a distributed approach (D-LAURA) localization is performed at the mobile nodes, whereas in a centralized approach the localization engine (and algorithm) runs at a central controller (C-LAURA). We present hereafter the implementation details of both solutions, further commenting on the advantages/drawbacks in terms of energy efficiency, induced traffic load.

4.1. Distributed LAURA

D-LAURA is designed to offload the central controller from the processing burden coming from the localization, further reducing the traffic load in the network. The idea here is to move the processing burden to the sensor nodes which implement on-board both the localization and

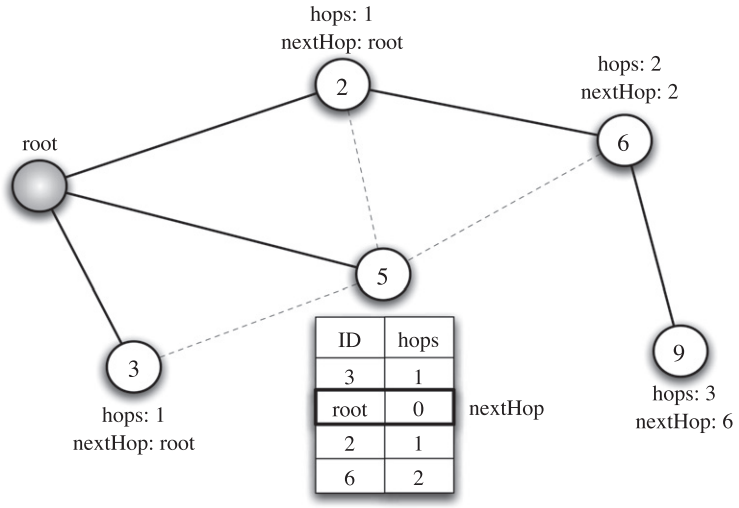


Fig. 3. Distance vector routing algorithm.

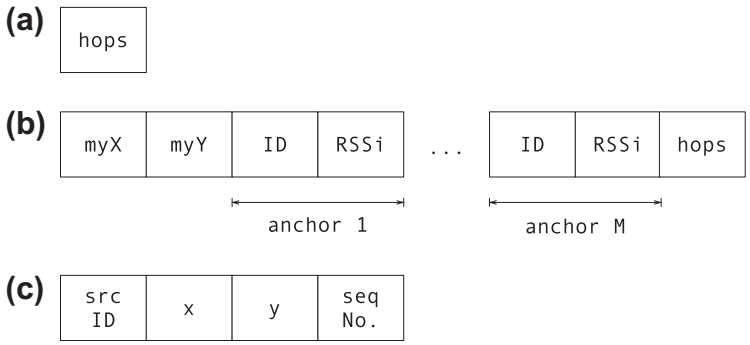


Fig. 4. Messages exchanged in D-LAURA. (a) Beacon-like HELLO message contains the current number of hops in the distance vector routing algorithm and are used to obtain RSSI measurements. (b) INFO messages contain informations to build **S** and **D**. (c) After the execution of the localization algorithm, a client node sends an UPDATE message to the PAN coordinator.

classification engines presented in Section 3.2 and Section 3.4, respectively.

The localization algorithm requires the dynamic construction and maintenance of the RSSI matrix among anchor nodes (**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 **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 achieve these goals, anchor nodes periodically send two types of messages: HELLO and INFO messages. With reference to Fig. 4, HELLO messages are used to implement the routing beacons described in Section 3.5 further allowing client nodes to sense the corresponding anchor's signal strength (so that $\hat{\mathbf{s}}$ can be estimated). Inter-anchor measurements are contained in the INFO message: anchor i stores the RSSI from other anchors in a column vector $\mathbf{s}_i$ and periodically broadcast it so that client nodes can construct the RSSI matrix. In addition, the anchor current coordinates are contained in the INFO message, so that also the distance matrix **D** can be set up. After having collected enough beacons and constructed the inter-anchor measurements matrix, the mobile node can estimate its position using the algorithm presented in Section 3.2. The estimated position can then be delivered to the PAN coordinator using a specific UPDATE message. Since the localization engine is implemented on resource-constrained devices, the "heavy" particle filter is substituted by a more lightweight IIR low-pass filter to support the tracking service [30]. The estimated position at time k is thus given by:

$$\hat{\mathbf{x}}(k) = (1 - \alpha_x)\hat{\mathbf{x}} + \alpha_x \hat{\mathbf{x}}(k-1), \tag{18}$$

where α_x is set to 0.8 in our implementation and $\hat{\mathbf{x}}$ is estimated with the gradient descent algorithm presented in Section 3.3. Results and comments on the distributed approach are presented in Section 5.

4.2. Centralized LAURA

In C-LAURA, the localization algorithm is executed at a central controller which leverages the information on the RSSI samples collected by the anchor nodes. The rationale is that each mobile node periodically broadcast beacon messages which are captured by the anchor nodes. Each anchor node processes the beacons to retrieve the corresponding RSSI samples which are then sent to the central controller together with the inter-anchors RSSI samples.

As for the required signaling for supporting the localization service, the C-LAURA requires periodic beacons from mobile nodes and anchors. The former messages are used to let the anchors retrieve RSS samples from mobile nodes, whereas the latter messages are used to collect at the anchors the RSSI samples out of the other anchors. The beaconing frequencies of the two types of messages is, in general, different. The frequency of mobile node beacons is driven by the required frequency of the localization service. On the other hand, anchor node beacons may be sent out with lower frequencies since the propagation pattern among anchors is mostly stable (anchor nodes are fixed). Finally, signaling messages are required to deliver the information on the RSSI collected at the anchor nodes (from mobile and anchor nodes) to the central controller. To this extent, messages to deliver Anchor-to-Anchor (A2A) RSSI samples and Mobile-To-Anchor (M2A) RSSI samples are to be used (see Fig. 5).

Unlike the distribute case, where the RSSI measurements from the anchor nodes to the target are made by the target itself, here the paradigm is inverted. The vector $\mathbf{s}$ that is collected for each target at the central nodes contains in fact a measure on how the anchor node receive beacons from that particular target. Thus the centralized algorithm will use $\mathbf{S}^T$ instead of $\mathbf{S}$ in the computation of the signal-to-distance map as explained in Section 3.2.

4.3. Using the PMS to enhance the localization energy efficiency

The output of the PMS on the current movement status of the patient can be leveraged to further increase the energy efficiency of the localization system. The rational is to "switch off" the localization engine and signaling if the target patient is not moving. In C-LAURA, the mobile nodes may quit sending beacons, whereas in D-LAURA the mobile nodes may turn off the localization engine and early discard beacons received form the anchors. The use of the accelerometer does not incur in any additional "energy" cost since the accelerometer is already used in the PMS. In practical implementations, the Signal Magnitude Area (SMA) is computed in a short observation windows of 200 ms and compared against a threshold to decide if the target is moving or not.

5. Performance evaluation

The distributed and centralized versions of LAURA have been designed and implemented in order to accomplish the same task following different paradigms. In the first, the burden of computation is spread across the motes; in the second it is entirely committed to a central controller, trying to save as many resources as possible on the sensor nodes. In this section we evaluate and compare the performance of the two proposed systems in terms of localization accuracy, energy consumption and traffic load. The performance evaluation proposed in this section does characterize strengths and weaknesses of the distributed and centralized approaches thus providing general guidelines for the design of the complete system.

This performance evaluation is carried out on a testbed wireless sensor network which runs both D-LAURA and C-LAURA composed of MEMSIC MICAz motes operated by

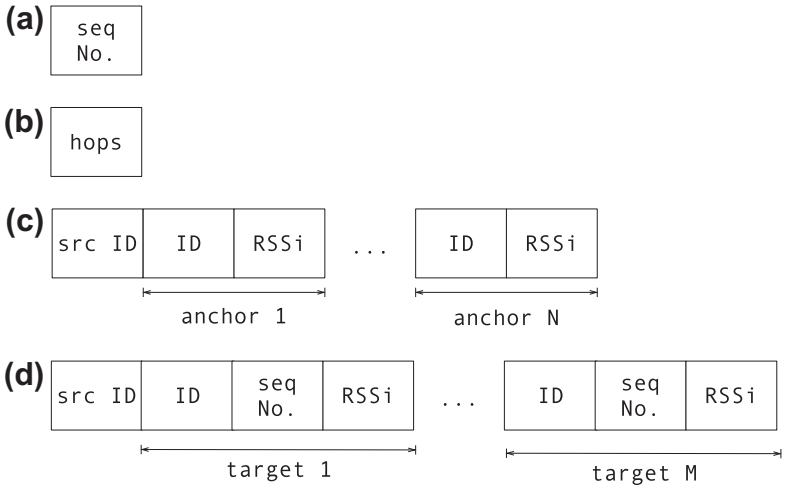


Fig. 5. Messages exchanged in C-LAURA. (a) Mobile node beacons contain a sequence number (time stamp), (b) anchor node beacons contain the current number of hops in the distance vector routing algorithm and are used to exchange RSSI measurements among anchors; (c) messages used to deliver the information on the RSSI samples among the anchor nodes (A2A messages) and (d) messages used to deliver the information on the RSSI samples collected at the anchor nodes out of mobile nodes beacons (M2A samples).

Table 1
LAURA memory footprints.

	RAM (bytes)	ROM (bytes)
D-LAURA		
Anchors	1301	17496
PLTS	1146	25826
PMS	324	10314
C-LAURA		
Anchors	881	17022
PLTS	414	15072
PMS	324	10314

TinyOS. The mobile nodes are additionally equipped with a MEMSIC MTS310 sensor board that enables acceleration measurements. Table 1 shows the memory footprints of the TinyOS executables for the different modules in both versions of LAURA.

5.1. Localization accuracy

The target application scenario requires accurate localization to ease up intervention of the medical staff in case of medical emergency of critical situations. To this extent, we start off by evaluating the localization error of the algorithm proposed in Section 3.2 for the case of static nodes (i.e. when the particle filter is deactivated). Several anchor nodes are deployed at different positions in a 100 m^2 indoor environment characterized by non-line-of-sight propagation due to the presence of walls and furniture. The test environment is divided into a 90 cm $\times$ 90 cm grid of points, resulting in more than 100 test points. A target node running C-LAURA is then used to estimate the localization error at each of the test points: the node is placed at a specific test point for 30 s, and the results are averaged over this period. The same test is also repeated for different anchor node densities. We attached 9, 12 and 15 anchor nodes on the walls of the test environment, resulting in densities of 0.09, 0.12 and 0.15 anchors/m^2. Fig. 6a shows the cumulative density function (CDF) of the localization error for different values of the anchor node density. We observe that, at the 80th percentile, the localization error varies between 2.5 and 4 m when using 0.15 and 0.09 anchors/m^2 respectively.

In the second experiment, we evaluate the performance of the complete localization system that integrates particle filtering. In this case, a mobile node is tracked while moving along a pre-fixed path in the very same indoor environment of the first experiment. The path is sampled each 90 cm to allow synchronization of the RSSI message reception with the ground-truth position of the mobile node. For every RSSI packet, the location of the target is estimated using the algorithm described in Section 3.2. Fig. 6b shows the CDF of the localization error for an anchor node density of 0.15 nodes/m^2. We note that when the particle filter is disabled, an error below 3 m is achieved in 80% of the cases. Enabling the particle filter improves the localization accuracy of about 1 m.

In the distributed case, the localization algorithm is entirely executed on resource-constrained mobile nodes, thus, besides localization accuracy, also localization processing time may become an issue. In other words, the system has to achieve acceptable localization accuracy with a reasonably low processing time. As explained in Section 3.2, the most complex (and processor-eager) task to be performed by the localization algorithm is an SVD of the matrix S containing the inter-anchors RSSI measurements. The required processing time scales up with the dimension of matrix S which depends on the number of anchor nodes N to be considered for localization.

Roughly speaking, N should be set as small as possible to obtain an acceptable localization error while keeping the computation feasible. To evaluate the impact of N onto the localization processing time, we have implemented the localization algorithm of Section 3.2 on the MICAz sensor boards featuring an ATMega 128L processor and we have measured the required processing time for different values of N. Fig. 7a reports the outcome of this experiment, which shows that the processing time increases more than linearly with the number of anchors considered in the localization process.

For what concern localization accuracy, one may think that the higher is N, the lower the localization error is.

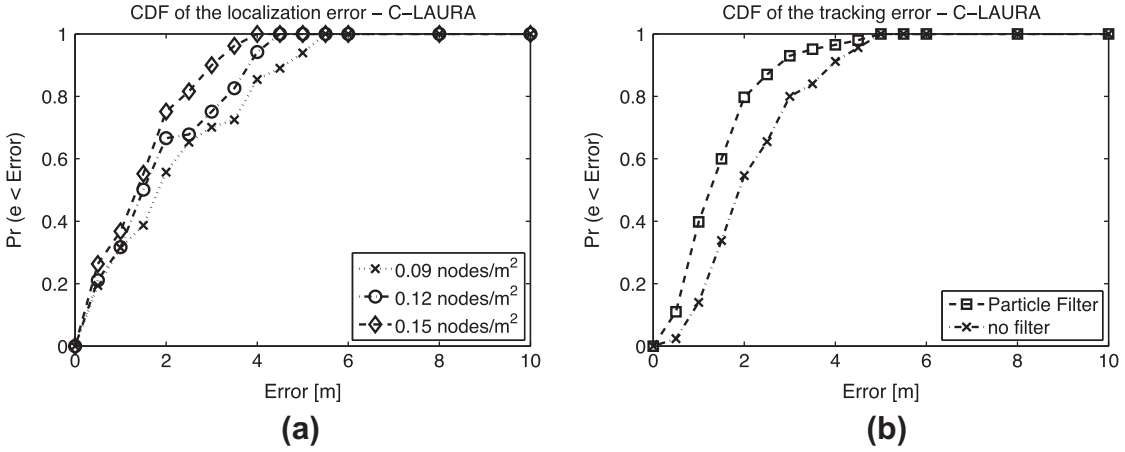


Fig. 6. C-LAURA Cumulative density Function of (a) the static localization error and (b) the tracking error in a test area of 100 m^2.

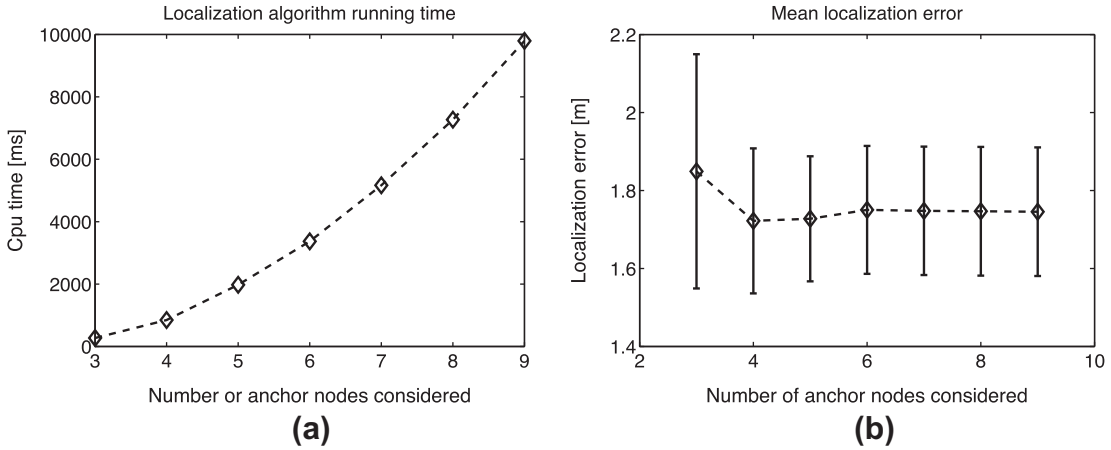


Fig. 7. (a) CPU time spent in the computation of the zero-configuration indoor localization algorithm on a MICAz node (ATMega128L MCU) and (b) the mean error vs the number of anchor nodes considered by the localization engine.

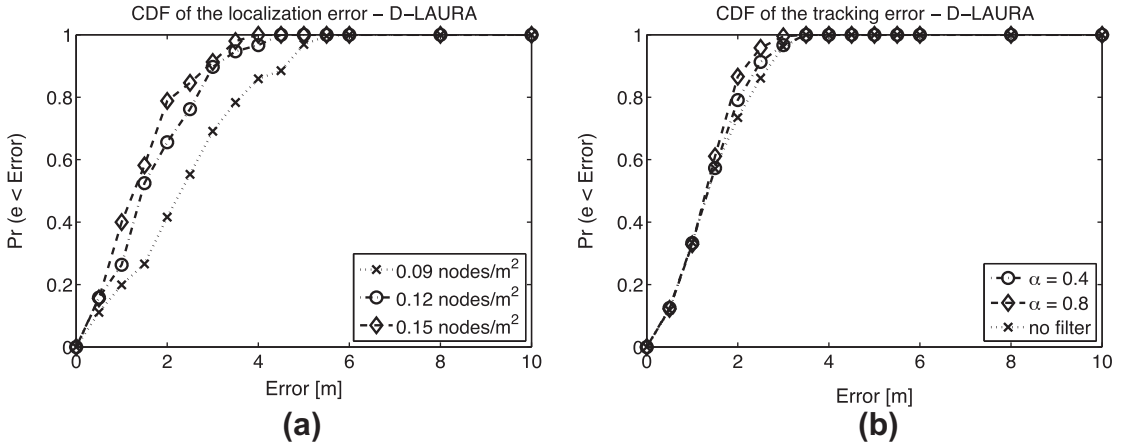


Fig. 8. D-LAURA Cumulative density Function of (a) the static localization error and (b) the tracking error in a test area of 100 m^2.

While this assumption is generally true for time-of-arrival based localization systems (i.e. GPS), this is not the case of RSSI-based localization systems. When anchor nodes are scattered in the monitored area with low density, it is very likely that most of the entries in $\hat{\mathbf{s}}$ will be very. These very low RSSI values are highly impacted by random noise over time and may impair the the improve localization accuracy. In Fig. 8 we summarize this behavior: we plot the mean localization error for the same setting of Fig. 6a, with a fixed anchor node density of 0.15 nodes/m^2. Similar results have been obtained under different anchor node densities. As one can see, using N greater than 4 does not improve accuracy: on the contrary the mean localization error slightly increases as N increases. From the results of Section 3.2 and Fig. 8, a reasonable trade-off between localization accuracy and localization processing time appears to be achieved when using $N = 4$.

Next, the same experiments carried out for the centralized case are repeated for D-LAURA: in the first experiment we used a static D-LAURA mote to estimate the localization error in different position and for different anchor nodes density. We recall that in this case only the best 4 anchors are considered in order to produce a result. Fig. 8a shows the CDF of the static localization error in the distributed case: the results are in line with the ones in Fig. 6b, showing that we could successfully distribute the localization algorithm without losing any precision.

For the tracking experiment, the same procedure as the centralized case is used. However, as explained in Section 4.1, in D-LAURA the particle filter is substituted with a more lightweight low-pass filter. Figure shows the CDF of the tracking error under different values of the forgetting factor α of the low-pass filter. As clear form the figure, a value $\alpha = 0.8$ provides similar localization as the particle filter scenario, with an error below 2 m at the 80th percentile.

The measure of error for localization used in these tests (the CDF of the euclidian distance between the real and the estimated position) is widely accepted in the research area of indoor localization system. However, in the reference application it is reasonable considering more "loose" error measures: as an example, the nursing staff may be more interested in the room where a specific patient is,

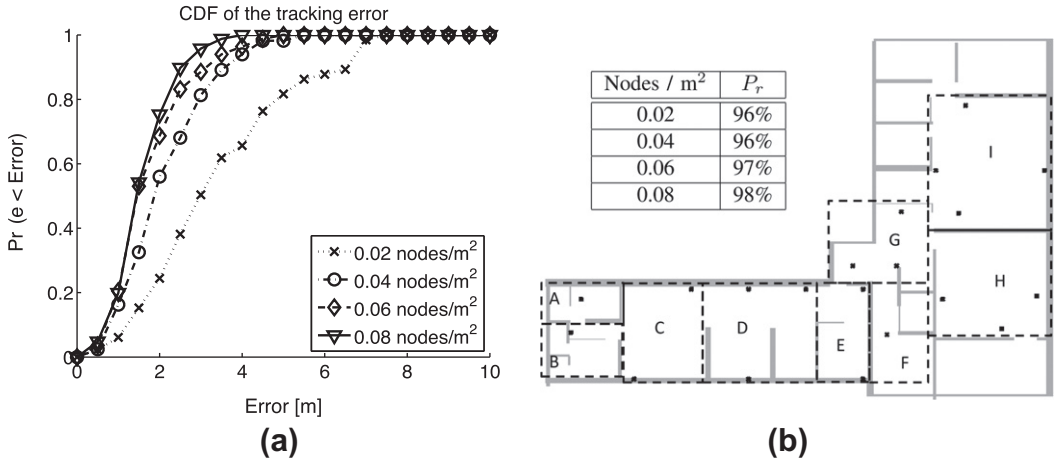


Fig. 9. (a) Standard localization error for the test environment shown in (b). The black markers correspond to the position of the anchor nodes (only the maximum density is shown).

rather then her exact position inside the building. To this extent, we performed a different experiment in an area of 250 m^2. We divided the area in nine areas (i.e. rooms) and we computed the probability P_r that the tracked position lied inside the correct area. Again, we repeated the experiment for different anchor densities, this time investigating sparser distributions, from 0.08 down to 0.02 anchors/m^2. For all the tested configuration we obtained a value for P_r greater than 96%, slightly increasing accordingly to the increasing anchor node density. Fig. 9a shows the standard localization error, while Fig. 9b shows the test environment, where we have emphasized the rooms in which we have divided the test area, labeled by upper-case letters. Fig. 9b also reports the values of P_r for different anchor densities.

5.2. Energy consumption

Energy consumption of the mobile motes is a major issues in every application and should be carefully managed and minimized where possible. In this section we derive the energy consumption models for both the distributed and centralized systems, including radio transmission, acceleration sensing and cpu processing. We base our model on the values of the current draw for each of the MEMSIC MICAz mote components, provided by the MICAz datasheet, and listed in Table 2 (I_{rx} and I_{tx} are the currents draw for radio reception and transmission, I_{cpu} is the current draw when using the CPU and I_{acc} is the current drawn by the accelerometer). The energy consumption analysis is carried out over a reference time span of T = 1 s, N as the number of anchors considered by the mobile mote when running the localization algorithm and V_{cc} = 3.3 V as the voltage supply.

The generic expression of the energy consumed by a mobile target node is:

$$E = E_{cpu} + E_{tx} + E_{rx} + E_{acc} \tag{19}$$

being E_{cpu} the energy consumed by the CPU, E_{tx} the energy consumed by the radio when transmitting data, E_{rx} the energy consumed by the radio when receiving data, E_{acc} the energy consumed by the accelerometer. Expanding (19) we obtain:

Table 2
MEMSIC MICAz energy consumption.

I_{tx}	17.4 mA
I_{rx}	19.7 mA
I_{cpu}	8 mA
I_{acc}	0.6 mA

$$E = V_{cc}(I_{cpu}T_{cpu} + I_{tx}T_{tx} + I_{rx}(T - T_{tx}) + I_{acc}T). \tag{20}$$

In the last equation we have considered that: (i) T_{cpu} is a function of the number N of anchor nodes considered by the algorithm as stated in Section 5.1, plus an offset due to the movement classification algorithm (ii) the MTS310 sensor board is always active and samples acceleration continuously; (iii) the radio is either in the transmission or reception state and does not enter in idle or sleep states; (iv) the transmission/reception times are computed as L/R, where L is the message length in bits and R is equal to 250 Kbps. Moreover, if the system can be "turned off" when the target is not moving in the reference time period T, the average energy consumption is to be scaled down by the mobility factor of the target node. Ideally, if the target is not moving, the sensor node is not consuming any energy, movement estimation apart.

We further point out that in the distributed case (D-LAURA) all the effort is made by the mobile nodes, hence every term in (19) contributes to the final energy consumption. Conversely, in C-LAURA only the transmission and accelerometer energy contributes to the final consumption, as the target node does not receive any message (E_{rx} = 0) or execute any algorithm (E_{cpu} = 0).

Fig. 10a reports the energy consumption for the two cases, D-LAURA and C-LAURA, for different mobility patterns of the target node. Each mobility patterns is characterized by the probability that the mobile node is actually moving in the reference period T going from 0% (static node) to 100% (node always moving). The main result coming form the figure is that C-LAURA outperforms D-LAURA in terms of energy efficiency. In fact, the energy

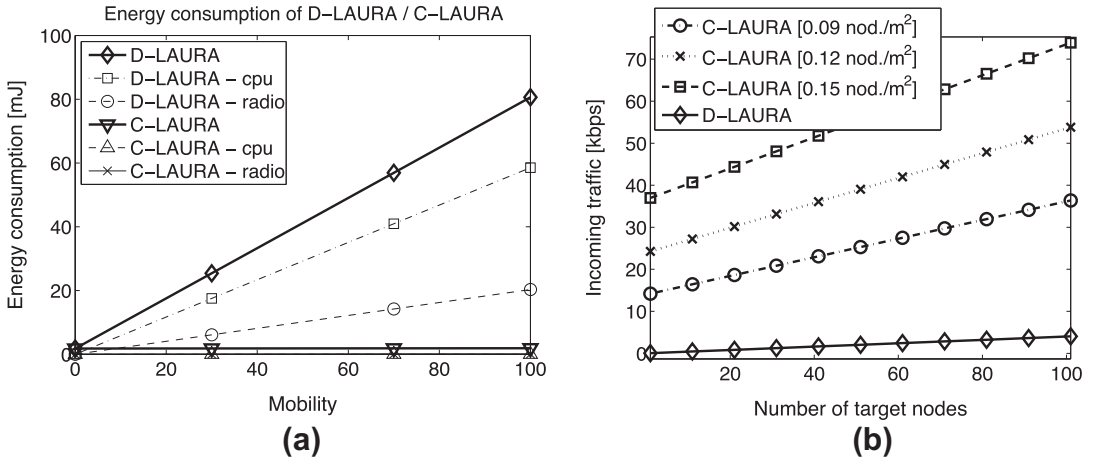


Fig. 10. (a) Energy consumption of the distributed and centralized system for different percentage of mobility. The solid lines represent the total energy consumptions. (b) Signaling traffic reaching the central controller in D-LAURA and C-LAURA, when different anchor node densities and number of mobile targets are used.

consumption of C-LAURA is almost 40 times lower than D-LAURA for medium mobility patterns.

5.3. Signaling overhead

After having analyzed the energy consumption of the two versions of LAURA, it is worth focusing on the induced signaling overhead required by the two solutions. To this extent, we derive a qualitative estimate of the signaling traffic generated by D-LAURA and C-LAURA which flows through the central controller. The calculation is performed assuming N_{tot}deployed anchors and M mobile target to be localized.

D-LAURA performs all the computation needed on the motes, without relying on any single device to execute the localization algorithm. Each of the M active targets listens to INFO and HELLO packets and computes its own position. The produced estimation is then encapsulated in UPDATE messages which includes the node ID, and the estimated positions (x,y). Assuming a node ID field of 2 bytes and x, y fields of 4 bytes each, the average signaling traffic per mobile target induced by D-LAURA is $80 \times \frac{1}{T_{NU}}$[bit/s], being $\frac{1}{T_{NU}}$ the localization frequency. The overall signaling traffic at the central controller is then:

$$S_{D\text{-}LAURA} = 80 \times M \times \frac{1}{T_{NU}} [\text{bit/s}]. \quad (21)$$

In C-LAURA, the signaling traffic reaching the central controller includes the messages sent by the anchor nodes carrying the anchor-to-anchor and node-to-anchor RSSI samples. Referring back to Section 4.1, the localization information carried by the A2A messages includes the ID of the sending anchor (2 bytes) and the ID (2 bytes) and the corresponding RSSI value (1 byte) for each one of the surrounding anchors. M2A messages carry the ID of the sending anchor (2 bytes) and, for each one of the "received" mobile node, the corresponding ID (2 bytes), the RSSI sample (1 byte) and the sequence number (1 byte). To estimate the overall signaling load, we further observe that the A2A messages are sent out with a frequency $1/T_{A2A}$ which is generally lower than the M2A messages T_{M2A}. Under these assumptions, the signaling load can be calculated as:

$$S_{C-LAURA} = N_{tot} \times \left[(16 + 24K) \times \frac{1}{T_{A2A}} + (16 + 32Q) \times \frac{1}{T_{M2A}} \right], \quad (22)$$

where K and Q are respectively is the number of anchor nodes and the number of mobile nodes which are in visibility of a reference anchor. Parameters K and Q obviously depend on the specific network topology and propagation environment. In any case, each signaling message must fit the maximum frame size allowed by the IEEE 802.15.4 link layer[1] which is 128 bytes. To this extent, an hard limit exists on K and Q; namely, $K \leqslant 42$ and $Q \leqslant 31$. Fig. 10b shows the overall signaling traffic of the two versions of LAURA: we assume anchors and targets uniformly distributed in space and we compute the total traffic reaching the central controller when both anchor nodes density and the number of target nodes vary. It can be seen that D-LAURA outperforms C-LAURA in terms of traffic loads, even when hundreds of target nodes are simultaneously localized.

5.4. PMS performance evaluation

In order to evaluate the accuracy of the classification algorithm we have conducted several experiments in a controlled environment. We have successfully ported a TinyOS implementation of the PMS onto a MICAz mote, leveraging a MTS310 sensor board mounting an ADXL202 bi-axial accelerometer (see Table 1 for memory footprints). We emphasize that the PMS is integrated in the existing localization system, providing simultaneous localization and monitoring. We tested the classification algorithm in a gym on eight persons with ages between 21 and 56 years. The accelerometer is carefully attached to the waist of each person, in a configuration similar to the one showed in Fig. 2b. Each person is then asked to perform ten different

[1] To avoid fragmentation.

Table 3
Confusion matrix for prone, supine, standing and walking movements. Rows represent the ground truth, columns represent the classified movement.

	PRONE	SUPINE	STANDING	WALKING
PRONE	100%	0	0	0
SUPINE	0	100%	0	0
STANDING	0	0	97%	3%
WALKING	0	0	0	100%

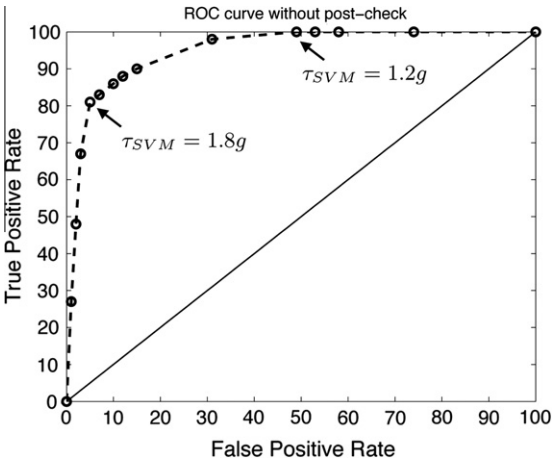


Fig. 11. Receiver Operating Characteristic curve for the fall detection algorithm. The curve is obtained by varying the threshold τ_{SVM} and observing the classifier output over the measurements dataset.

trails that included different actions (lying down, standing still, walking, falling down, etc.).

For each movement we observed the resulting output and compute the accuracy of the algorithm averaging over the ten trails. Table 3 shows the confusion matrix for all the movements except for falling. As clear from the table, almost perfect classification is achieved for all the movement categories. For what concerns the fall detection algorithm, it is clear that setting properly τ_{SVM} plays a critical role in the final deception accuracy. High values of τ_{SVM} minimize the false alarms (false positives)[2] at the risk of ignoring actual falls. Conversely, using a low τ_{SVM} results in accurate fall detection (true positives), at the cost of several false alarms. The fine tuning of τ_{SVM} is thus central in the reference application scenario. To this extent, Fig. 11 shows the Receiver Operating Characteristic (ROC) curve of the fall detection algorithm, which depicts the true positive probability (detection accuracy) against the false positive probability (false alarms) for different values of τ_{SVM}. Since the target application requires high accuracy in fall detection, we select $\tau_{SVM} = 1.2g$ which leads to a true positive probability of 1. The drawback of this choice is the high false positive rate (around 40%). However, enabling the a posteriori check on the patient position explained in Section 3.4 dramatically decrease the false positive rate to zero, without affecting the final accuracy.

[2] Fall is detected but the patient has not fallen.

6. Conclusions

We described the LAURA project, an Integrated system based on Wireless Sensor Networks for patient monitoring, localization and tracking. The system can be rapidly deployed in any indoor environment, due to adopted self-calibration method. We developed both a distributed solution entirely running on a low-cost mobile sensor node and a centralized solution that minimize the target node energy consumption. In both cases, the experimental evaluation carried out in real environments has shown that a localization accuracy of about 2–3 m can be achieved also with a relatively sparse deployment of the anchor nodes (approx. 0.15 nodes/m^2). The personal monitoring system leverages a single biaxial accelerometer and it is used to monitor different movements with high accuracy. The described system has also been deployed and tested at the premises of a partner nursing institute.

Acknowledgement

This work has been carried out within project LAURA sponsored by the "5 per mille giovani" grant of Politecnico di Milano.

References

[1] W. Mann, The aging population and its needs, IEEE Pervasive Computing 3 (2) (2004) 12–14. http://dx.doi.org/10.1109/MPRV.2004.1316812.
[2] J. Ko, T. Gao, R. Rothman, A. Terzis, Wireless sensing systems in clinical environments: Improving the efficiency of the patient monitoring process, Engineering in Medicine and Biology Magazine, IEEE 29 (2) (2010) 103–109, http://dx.doi.org/10.1109/MEMB.2009.935713.
[3] Y. Ren, R. Pazzi, A. Boukerche, Monitoring patients via a secure and mobile healthcare system, Wireless Communications, IEEE 17 (1) (2010) 59–65, http://dx.doi.org/10.1109/MWC.2010.5416351.
[4] F. Evennou, F. Marx, Advanced integration of wifi and inertial navigation systems for indoor mobile positioning, EURASIP Journal on Applied Signal Processing (2006). pp. 164–164. doi:http://dx.doi.org/10.1155/ASP/2006/86706.
[5] H. Wang, H. Lenz, A. Szabo, J. Bamberger, U. Hanebeck, Wlan-based pedestrian tracking using particle filters and low-cost mems sensors, 2007, 1–7. doi:10.1109/WPNC.2007.353604.
[6] S.-Y. Lau, T.-H. Lin, T.-Y. Huang, I.-H. Ng, 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, ACM, New York, NY, USA, 2009, pp. 35–42. doi:http://doi.acm.org/10.1145/1614293.1614300.
[7] K. Lorincz, M. Welsh, Motetrack: a robust, decentralized approach to rf-based location tracking, in: LoCA, Lecture Notes in Computer Science, vol. 3479, Springer, 2005, pp. 63–82.
[8] M. D'Souza, T. Wark, M. Ros, Wireless localisation network for patient tracking, 2008, pp. 79–84. doi:10.1109/ISSNIP.2008.4761966.
[9] N. Patwari, J. Ash, S. Kyperountas, A. Hero III, R. Moses, N. Correal, Locating the nodes: cooperative localization in wireless sensor networks, IEEE Signal Processing Magazine 22 (4) (2005) 54–69.
[10] X. Li, Collaborative localization with received-signal strength in wireless sensor networks, IEEE Transactions on Vehicular Technology 56 (6) (2007) 3807–3817, http://dx.doi.org/10.1109/TVT.2007.904535.
[11] M. Sugano, T. Kawazoe, Y. Ohta, 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.
[12] R. Bader, M. Pinto, F. Spenrath, P. Wollmann, F. Kargl, BigNurse: a wireless ad hoc network for patient monitoring, in: IEEE Pervasive Health, First Workshop on Location Based Services for Health Care (Locare'06), IEEE, Innsbruck, Austria, 2006.

[13] P. Barsocchi, S. Lenzi, S. Chessa, G. Giunta, Virtual calibration for rssi-based indoor localization with ieee 802.15.4, 2009, pp. 1–5. doi:10.1109/ICC.2009.5199566.
[14] H. Lim, L.-C. Kung, J.C. Hou, H. Luo, Zero-configuration indoor localization over ieee 802.11 wireless infrastructure, Wireless Networks 16 (2) (2010) 405–420.
[15] J. Ko, C. Lu, M. Srivastava, J. Stankovic, A. Terzis, M. Welsh, Wireless sensor networks for healthcare, Proceedings of the IEEE 98 (11) (2010) 1947–1960, http://dx.doi.org/10.1109/JPROC.2010.2065210.
[16] A.-K. Chandra-Sekaran, P. Dheenathayalan, P. Weisser, C. Kunze, W. Stork, Empirical analysis and ranging using environment and mobility adaptive rssi filter for patient localization during disaster management, in: Fifth International Conference on Networking and Services, 2009 (ICNS '09), 2009, pp. 276–281. doi:10.1109/ICNS.2009.63.
[17] B. Steele, L. Holt, B. Belza, S. Ferris, S. Lakshminaryan, D. Buchner, Quantitating physical activity in COPD using a triaxial accelerometer, Chest 117 (2000) 13591367.
[18] G. Currie, D. Rafferty, G. Duncan, E. Bell, A. Evans, Measurement of gait by accelerometer and walkway: a comparison study, Medical & Biological Engineering & Computing 30 (1992) 669670.
[19] K. Kiani, C. Snijders, E. Gelsema, Computerized analysis of daily lifemotor activity for ambulatory monitoring, Technology and Health Care 5 (1997) 3073187.
[20] F. Foerster, J. Fahrenberg, Motion pattern and posture: correctly assessed by calibrated accelerometers, Behavior Research Methods, Instrumentation & Computers 32 (2000) 450457.
[21] K. Aminian, P. Robert, E. Buchser, B. Rutschmann, D. Hayoz, M. Depairon, Physical activity monitoring based on accelerometry: validation and comparison with video observation, Medical & Biological Engineering & Computing 37 (1999) 304–308.
[22] P.H. Veltink, H.B. Bussmann, W. de Vries, W. Martens, R. van Lummel, Detection of static and dynamic activities using uniaxial accelerometers, IEEE Transactions on Rehabilitation Engineering 4 (4) (1996) 375385.
[23] J. Winters, Y. Wang, Wearable sensors and telerehabilitation, IEEE Engineering in Medicine and Biology Magazine 22 (3) (2003) 56–65.
[24] D. Karantonis, M. Narayanan, M. Mathie, N. Lovell, B. Celler, Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring, IEEE Transactions on Information Technology in Biomedicine 10 (1) (2006) 156167.
[25] M. Mathie, A. Coster, B. Celler, N. Lovell, Classification of basic daily movements using a triaxial accelerometer, Medical and Biological Engineering and Computing 42 (2004) 670687.
[26] E.K. Antonsson, R.W. Mann, The frequency content of gait, Journal of Biomechanics 18 (1) (1985) 39–47, http://dx.doi.org/10.1016/0021-929 (85)90043-0.
[27] A. Bhattacharya, E.P. McCutcheon, E. Shvartz, J.E. Greenleaf, Body acceleration distribution and o2 uptake in humans during running and jumping, Journal of Applied Physiology 49 (5) (1980) 881–887.
[28] A. Redondi, M. Tagliasacchi, M. Cesana, L. Borsani, P. Tarrio, F. Salice, LAURA – LocAlization and Ubiquitous monitoRing of pAtients for health care support, in: IEEE International Workshop on Advances in Positioning and Location-Enabled Communications, 2010 (APLEC '10), 2010.
[29] L. Borsani, A. Redondi, S. Guglielmi, M. Cesana, Tree-based routing protocol for mobile wireless sensor networks, in: Proceedings of the 8th IEEE International Conference on Wireless On-Demand Network Systems and Services (WONS2011), 2011, pp. 158–163.
[30] D. Anzai, S. Hara, Does particle filter really outperform low pass filter in indoor target tracking? in: 21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '10), 2010, pp. 882–886.

Alessandro E.C. Redondi received his Master degree in Computer Engineering in July 2009. Currently he is a Ph.D. student at the Department of Electronics and Information of the Politecnico di Milano. His research activities are focused on algorithms and protocols for Real Time Localization Systems and Wireless Multimedia Sensor Networks.

Marco Chirico has got his Master degree in Computer Engineering form Politecnico di Milano in September 2011 with a thesis on the development of localization protocols over Wireless Sensor Networks. He is currently employed by IBM as IT Specialist.

Luca P. Borsani received his Master degree in Telecommunication Engineering in December 2008. Since March 2009 he has been working as research associate at the Advanced Network Technologies Laboratory (ANTLab) of the Department of Electronics and Information of the Politecnico di Milano. His research activities include several aspects of wireless sensor networks design, with focus on middleware, routing and MAC protocols.

Matteo Cesana received his M.S. degree in Telecommunications Engineering and his Ph.D. degree in Information Engineering from the Politecnico di Milano in July 2000 and in September 2004, respectively. From September 2002 to March 2003 he has been working as a visiting researcher at the Computer Science Department of the University of California in Los Angeles (UCLA). He is now an Assistant Professor of the Electronics and Information Department of the Politecnico di Milano. His research interest are in the field of protocol design, optimization and evaluation for wireless networks. He is an Associate Editor of Ad Hoc Networks Journal (Elsevier).

Marco Tagliasacchi is an Assistant Professor in the Department of Electronics and Information at the Politecnico di Milano. His research interests include multimedia signal processing and information retrieval. He has a Ph.D. in information engineering from the Politecnico di Milano.