

Article

## Locally-Referenced Ultrasonic – LPS for Localization and Navigation

David Gualda \*, Jesús Ureña, Juan C. García and Alejandro Lindo

Electronics Department, University of Alcalá de Henares, Escuela Politécnica. Ctra. Madrid-Barcelona, Km. 33,600, 28871 Alcalá de Henares, Spain; E-Mails: urena@depeca.uah.es (J.U.); jcarlos@depeca.uah.es (J.C.G.); alejandro.lindo@depeca.uah.es (A.L.)

\* Author to whom correspondence should be addressed; E-Mail: david.gualda@depeca.uah.es; Tel.: +34-918-856-592; Fax: +34-918-856-591.

External Editor: Sisi Zlatanova

*Received: 17 September 2014; in revised form: 15 October 2014 / Accepted: 6 November 2014 / Published: 18 November 2014*

---

**Abstract:** This paper presents a flexible deployment of ultrasonic position sensors and a novel positioning algorithm suitable for the navigation of mobile robots (MRs) in extensive indoor environments. Our proposal uses several independently-referenced local positioning systems (LPS), which means that each one of them operates within its own local reference system. In a typical layout, an indoor extensive area can be covered using just a reduced set of globally-referenced LPS (GRLPS), whose beacon positions are known to the global reference system, while the rest of the space can be covered using locally-referenced LPSs (LRLPS) that can be distributed arbitrarily. The number of LRLPS and their position can be also changed at any moment. The algorithm is composed of several Bayesian filters running in parallel, so that when an MR is under the GRLPS coverage area, its position is updated by a global filter, whereas when the MR is inside the LRLPS area, its position is updated using position increments within a local filter. The navigation algorithm has been tested by simulation and with actual data obtained using a real set of ultrasonic LPSs.

**Keywords:** ultrasonic local positioning systems; globally-referenced LPS; locally-referenced LPS; indoor navigation; mobile robot; data fusion

---

## 1. Introduction

The estimation of the position and navigation tasks of a mobile robot (MR) in indoor environments has been an important research topic during the last few years. Some systems try to replicate indoors the GPS behavior and performances outdoors. The most popular technologies used in order to implement an indoor positioning system are Ultrasound (US) [1–4] and Radiofrequency (RF) [5–7].

Usually, these technologies are used to build local position systems (LPS) [8–10] composed of a set of beacons, placed on known fixed positions that emit a signal (RF, US), which is captured by a mobile receptor in order to estimate its position. Using some kind of distance-dependent information included within the beacons' signals (*i.e.*, time of flight, signal intensity), the receptor can estimate its position. If the recovered information is an absolute distance to each beacon, the estimation of the position can be obtained by a spherical trilateration [11]. The drawback of the spherical trilateration is that it is necessary to synchronize in time beacons and receivers. Alternatively, using the difference of distances to a reference beacon, the estimation of the position can be computed using a hyperbolic trilateration algorithm [12]. However, in such case of hyperbolic trilateration, the minimum set of beacons needed to estimate the receiver's position increases.

In the case of MR navigation, the most used positioning technique is odometry, based on the information provided by the encoders linked with their movement axes. Nevertheless, odometry is a relative positioning system whose estimations have an error that gets higher with time and that can also reflect impulsive error behavior related with skids or some other traction defects. Using process filtering, the odometry errors can be lowered to some grade. The most commonly used algorithms are based on Bayesian theory, like the unscented and extended Kalman filter (EKF, UKF) [13,14], the H-infinity [15] and the particle filter [16].

The best way to prevent heavy odometry errors is to fuse this source of information with complementary absolute positioning systems, like GPS outdoors. However, indoors, GPS cannot be used because of degradations in the reception from satellites, and that is why LPS are needed in those environments.

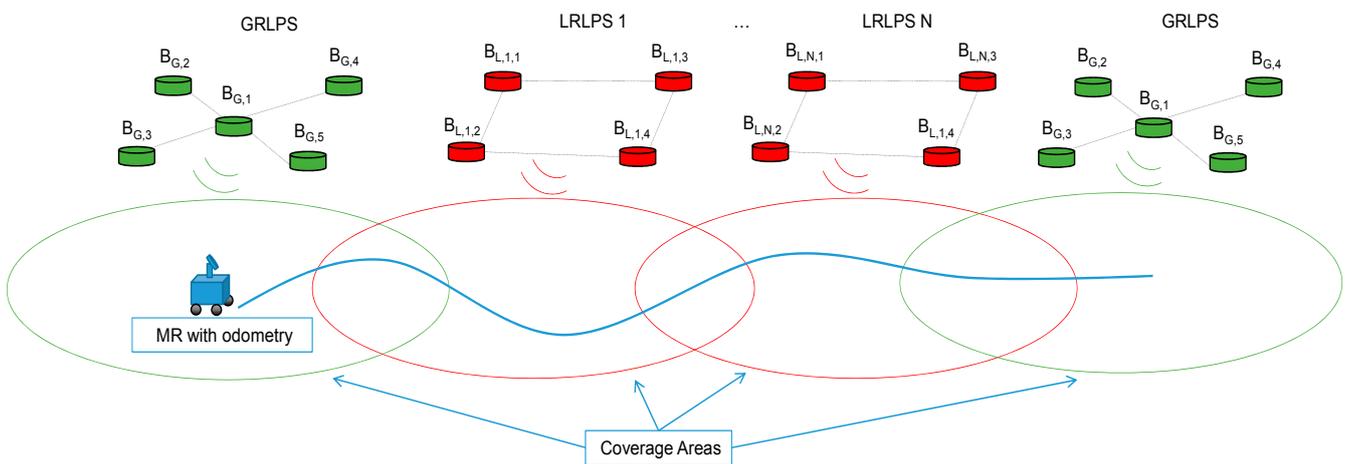
In extensive areas (for instance, inside large buildings), the deployed LPSs should cover most of the area, using as many additional LPS around as required by the degree of precision needed in the positioning task [10]. Ultrasonic beacons, for instance, give very good results when measuring distances by time of flight techniques (TOF), but the reduced range forces one to install several of these systems to cover large areas. Moreover, each LPS needs to be calibrated at their position after being deployed and before beginning the localization tasks. Therefore, the deployment of such systems is complicated when there are dynamic changes or a fast installation is required, for instance in emergencies or similar situations. If some or all of the LPSs in an extensive area could be installed without a previous calibration stage, many of those drawbacks would be avoided.

This paper presents a novel algorithm that allows one to navigate an MR in extensive areas using a set of LPSs, most of them without calibration. Section 2 describes the scenario and the statement of the problem. Section 3 introduces the proposed algorithm in both the spherical and the hyperbolic cases. Section 4 shows some simulated results. Section 5 presents real results. Section 6 describes the possibility to adapt the system for human tracking, and finally, Section 7 discusses some conclusions derived from the proposal.

## 2. Statement of the Problem and Definitions

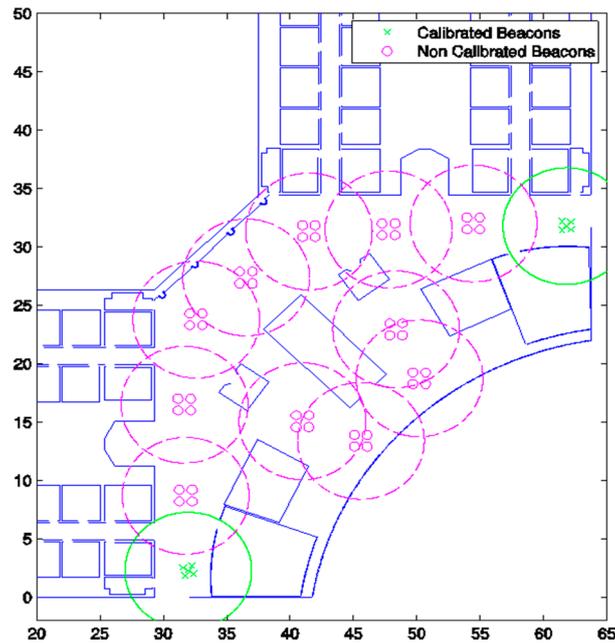
The test scenario is an extended indoor area where it is necessary to locate an MR. This large area is covered using a set of calibrated and uncalibrated LPSs (Figure 1). A calibrated LPS is the one whose beacons have been referenced to a global system, for instance after a calibration procedure; we will denote it as globally-referenced LPS (GRLPS). Normally, a GRLPS will be placed in strategic waypoints, like, for instance, the main entrances to a section in a building. The uncalibrated LPS will be identified as locally-referenced LPS (LRLPS), because only the relative positions of the beacons inside each LRLPS are known, and they have been determined prior to the deployment; they only depend on the geometry of the LPS and not on the place in which it is installed. LRLPSs can be placed all around the movement area, and their number and positions can be changed dynamically (at any time).

**Figure 1.** Particular case of globally-referenced LPS (GRLPS) and locally-referenced LPS (LRLPS) structures, with five and four beacons, respectively. MR, mobile robot.



Each LPS has to be composed of at least three beacons in the spherical case or four beacons in the hyperbolic one, in order to have enough distance measurements to apply the algorithms. Figure 2 shows a simulated scenario, where all beacons are placed on the ceiling, including GRLPSs, LRLPSs and their coverage areas on the ground. In this particular case, the GRPLS is composed of five beacons and the LRLPS of four beacons (so the GRLPS has more redundancy). Because of the performances achieved with the actual LPS used in real tests [9], we will suppose that for all of the LPSs, the coverage area is a circle on the ground with a radius of five meters.

The structure of each LPS must be known. In this case, we have used four beacons for each non-calibrated LPS structure (the LRLPS) composing a square with a side of 1 m; and five beacons, one more in the center of the square, for each calibrated LPS structure (the GRLPS). Some kind of overlapping between the LRLPSs' coverage area is recommended in order to reduce the positioning error of a mobile robot while navigating in the environment. If the MR moves outside the LPS coverage area (either a GR or LR LPS), it will only estimate its position using the odometry, so the error will increase accumulatively.

**Figure 2.** Simulated workspace based on the real plan of a building.

In order to make the paper more self-contained and to understand the proposed algorithm, some notations are included in this section:

- $\hat{\mathbf{X}}_{G,k}$ : Global state vector at instant  $k$ . This vector is composed of the global coordinates of the MR and its orientation. G indicates that the reference is global.
- $\hat{\mathbf{X}}_{L,n,k}$ : Local state vector at instant  $k$ . This vector is composed of the local coordinates of the MR and its orientation. L indicates that the reference is local, and  $n$  is the specific LRLPS.
- $\tilde{\mathbf{X}}_{G,k}$ : *A priori* global state vector at instant  $k$ .
- $\tilde{\mathbf{X}}_{L,n,k}$ : *A priori*  $n$ -local state vector at instant  $k$ .
- $d_{p_{G,k},b_{G,m}}$ : Real distance measured between the global position of the MR at instant  $k$  and the  $m$ -th global beacon of GRLPS.
- $\hat{d}_{p_{G,k},b_{G,m}}$ : Estimated distance between the global position of the MR at instant  $k$  and the  $m$ -th global beacon of GRLPS.
- $d_{p_{L,k},b_{L,m}}$ : Real distance measured between the local position of the MR at instant  $k$  and the  $m$ -th local beacon of the  $n$ -LRLPS.
- $\hat{d}_{p_{L,k},b_{L,m}}$ : Estimated distance between the local position of the MR at instant  $k$  and the  $m$ -th local beacon of the  $n$ -LRLPS.
- $\Delta d_{p_{G,k},b_{G,m}}$ : Real difference of distances measured between beacon  $m$  and beacon  $l$  of GRLPS to the global position of the MR at instant  $k$ .
- $\Delta \hat{d}_{p_{G,k},b_{G,m}}$ : Estimated difference of distances between beacon  $m$  and beacon  $l$  of GRLPS to the global position of the MR at instant  $k$ .
- $\Delta d_{p_{L,k},b_{L,m}}$ : Real difference of distances measured between beacon  $m$  and beacon  $l$  of the  $n$ -LRLPS to the global position of the MR at instant  $k$ .
- $\Delta \hat{d}_{p_{L,k},b_{L,m}}$ : Estimated difference of distanced between beacon  $m$  and beacon  $l$  of the  $n$ -LRLPS to the global position of the MR at instant  $k$ .

$\mathbf{A}_{G,k}$ :	Global transition matrix of the dynamic model.
$\mathbf{A}_{L,n,k}$ :	$n$ -local transition matrix of the dynamic model.
$h(\hat{\mathbf{X}}_{G,k})$ :	Global matrix observations obtained by the <i>a priori</i> information.
$h(\hat{\mathbf{X}}_{L,n,m})$ :	$n$ -local matrix observations obtained by the $n$ -local <i>a priori</i> information.
$\mathbf{H}_{G,k}$ :	Global measurement model matrix.
$\mathbf{H}_{L,n,k}$ :	$n$ -local measurement model matrix.
$\mathbf{P}_{G,k}$ :	Global covariance matrix
$\mathbf{P}_{L,n,k}$ :	$n$ -local covariance matrix.
$\mathbf{Q}_{G,k}$ :	Global process noise covariance matrix.
$\mathbf{Q}_{L,n,k}$ :	$n$ -local process noise covariance matrix.
$\mathbf{R}_G$ :	Global measure noise covariance matrix.
$\mathbf{R}_{L,n}$ :	$n$ -local measure noise covariance matrix.
$\sigma_{wG}^2$ :	Global measure variance (variance of distance in spherical case or difference of distances in hyperbolic one).
$\sigma_{wL,n}^2$ :	$n$ -local measure variance.

### 3. Proposed Algorithm

The pseudocode of the proposed algorithm is presented in Algorithm 1, and it can be applied for both the spherical and the hyperbolic cases. Basically, this pseudocode points out which state vector (local or global) is updated depending on the measurements performed by the MR according to its localization and the coverage area of the different LRLPSs. The process for the main lines of this pseudocode is explained with more detail in the following paragraphs.

The algorithm estimates the 2D location and the orientation of a mobile node inside the global coordinates system. The estimated position values for the actual instant  $k$  are described by the global state vector (1):

$$\hat{\mathbf{X}}_{G,k} = [\hat{x}_{G,k} \quad \hat{y}_{G,k} \quad \hat{\theta}_{G,k}]^T \quad (1)$$

When the mobile node is located inside the coverage area of a LRLPS, another local state vector (2) is defined additionally to the global vector:

$$\hat{\mathbf{X}}_{L,n,k} = [\hat{x}_{L,n,k} \quad \hat{y}_{L,n,k} \quad \hat{\theta}_{L,n,k}]^T \quad (2)$$

where  $n = 1, 2, \dots, N$  is the specific LRLPS.

In order to have always a globally referenced position and orientation of the MR, let us suppose that the first iteration of this algorithm will be computed when the MR is inside the coverage area of a GRLPS. This is a convenient condition and is also quite easy to accomplish, because GRLPSs have been specified to be installed on the entry points of the main building sections' areas of interest.

**Algorithm 1.** Pseudocode of the proposed algorithm.

---

```

1: while the MR is navigating ... do
2:     if (GRLPS coverage exists) then
3:         if (first iteration of GRLPS) then
4:             Global vector initialization
5:             Global covariance matrix initialization
6:         else
7:             Global state vector update using EKF
8:             Global covariance matrix update
9:         end if
10:    else if (LRLPS coverage exists) then
11:        for all LRLPS with coverage do
12:            if (first iteration of LRLPS(n)) then
13:                Local vector initialization
14:                Local covariance matrix update
15:            else
16:                Local state vector update using EKF
17:                Local covariance matrix update
18:            end if
19:        end for
20:        Global state vector update using the increment of distance and angle
        between  $k$  and  $k - 1$  iteration of the local state vector that has the least
        value of its covariance matrix
21:    else
22:        Global state vector update using only odometry
23:        Global covariance matrix update
24:    end if
25: end while

```

---

*3.1. Global Initialization for the GRLPS Area (Lines 4 and 5 in Algorithm 1)*

When the MR goes into a GRLPS area for the first time, the global state vector (3) is obtained using a Gauss–Newton minimization algorithm in order to minimize the Equations (4) or (5) depending on the case (spherical or hyperbolic):

$$\hat{\mathbf{X}}_{G,0} = [\hat{x}_{G,0} \quad \hat{y}_{G,0} \quad \hat{\theta}_{G,0}]^T \quad (3)$$

$$[\hat{x}_{G,0} \quad \hat{y}_{G,0}]^T = \arg \min \sum_{m=1}^M (|d_{p_{G,0}, b_{G,m}} - \hat{d}_{p_{G,0}, b_{G,m}}|) \quad (4)$$

$$[\hat{x}_{G,0} \quad \hat{y}_{G,0}]^T = \arg \min \sum_{m=2}^M (|\Delta d_{p_{G,0}, b_{G,m}} - \Delta \hat{d}_{p_{G,0}, b_{G,m}}|) \quad (5)$$

where  $m = 1, 2, \dots, M$  represents each specific beacon of this GRLPS. Note that in the hyperbolic case,  $m = 2, 3, \dots, M$ , because the first beacon is the reference one.

In the case of spherical trilateration, the estimated distance between the MR to each beacon for that specific GRLPS is used as defined in Equation (6):

$$\hat{d}_{p_{G,0},b_{G,m}} = \sqrt{(\hat{x}_{G,0} - xb_{G,m})^2 + (\hat{y}_{G,0} - yb_{G,m})^2 + (z_{MR} - zb_{G,m})^2} \quad (6)$$

where  $z_{MR}$  is the height of the MR.

For the hyperbolic trilateration, the differences of distances are used, as defined in Equation (7):

$$\Delta \hat{d}_{p_{G,0},b_{G,m}} = \sqrt{(\hat{x}_{G,0} - xb_{G,m})^2 + (\hat{y}_{G,0} - yb_{G,m})^2 + (z_{MR} - zb_{G,m})^2} - \sqrt{(\hat{x}_{G,0} - xb_{G,1})^2 + (\hat{y}_{G,0} - yb_{G,1})^2 + (z_{MR} - zb_{G,1})^2} \quad (7)$$

### 3.2. Global Navigation inside a GRLPS Area (Lines 7 and 8 in Algorithm 1)

For the rest of iterations, when the robot is navigating inside a GRLPS area, the global state vector and its covariance matrix are updated following the EKF structure:

Prediction stage:

$$\tilde{\mathbf{X}}_{G,k} = f(\tilde{\mathbf{X}}_{G,k-1}, \mathbf{u}_k) \quad (8)$$

$$\mathbf{P}_{G,k}^- = \mathbf{A}_{G,k} \cdot \mathbf{P}_{G,k-1} \cdot \mathbf{A}_{G,k}^T + \mathbf{Q}_G \quad (9)$$

Update stage

$$\mathbf{K} = \mathbf{P}_{G,k}^- \cdot \mathbf{H}_{G,k}^T \cdot [\mathbf{H}_{G,k} \cdot \mathbf{P}_{G,k}^- \cdot \mathbf{H}_{G,k}^T + \mathbf{H}_{G,k}^T + \mathbf{R}_G]^{-1} \quad (10)$$

$$\tilde{\mathbf{X}}_{G,k} = \tilde{\mathbf{X}}_{G,k} + \mathbf{K} \cdot (\mathbf{Z}_{G,k} - h(\tilde{\mathbf{X}}_{G,k})) \quad (11)$$

$$\mathbf{P}_{G,k} = (\mathbf{I} - \mathbf{K} \cdot \mathbf{H}_{G,k}) \cdot \mathbf{P}_{G,k}^- \quad (12)$$

Let us suppose that  $\Delta d_{odo}$  is the increment of distance and  $\Delta \theta_{odo}$  is the angular increment extracted from the MR odometry. The *a priori* estimation of the global state vector is defined by Equation (13), where we have used such odometry information to estimate this state.

$$\tilde{\mathbf{X}}_{G,k} = \begin{bmatrix} \hat{x}_{G,k-1} + \Delta d_{odo} \cdot \cos(\hat{\theta}_{G,k-1} + \Delta \theta_{odo}) \\ \hat{y}_{G,k-1} + \Delta d_{odo} \cdot \sin(\hat{\theta}_{G,k-1} + \Delta \theta_{odo}) \\ \hat{\theta}_{G,k-1} + \Delta \theta_{odo} \end{bmatrix} \quad (13)$$

The global transition matrix of the dynamic model is defined by Equation (14):

$$\mathbf{A}_{G,k} = \begin{bmatrix} \frac{\partial f(\tilde{\mathbf{X}}_{G,k}, \mathbf{u}_k)}{\partial \hat{x}_{G,k}} \\ \frac{\partial f(\tilde{\mathbf{X}}_{G,k}, \mathbf{u}_k)}{\partial \hat{y}_{G,k}} \\ \frac{\partial f(\tilde{\mathbf{X}}_{G,k}, \mathbf{u}_k)}{\partial \hat{\theta}_{G,k}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta d_{odo} \cdot \sin(\hat{\theta}_{G,k-1} + \Delta \theta_{odo}) \\ 0 & 1 & \Delta d_{odo} \cdot \cos(\hat{\theta}_{G,k-1} + \Delta \theta_{odo}) \\ 0 & 0 & 1 \end{bmatrix} \quad (14)$$

The global covariance matrix is obtained by Equation (15):

$$\mathbf{P}_{G,k} = \begin{bmatrix} \sigma_{x_G}^2 & \sigma_{x_G,y_G}^2 & \sigma_{x_G,\theta_G}^2 \\ \sigma_{y_G,x_G}^2 & \sigma_{y_G}^2 & \sigma_{y_G,\theta_G}^2 \\ \sigma_{\theta_G,x_G}^2 & \sigma_{\theta_G,y_G}^2 & \sigma_{\theta_G}^2 \end{bmatrix} \quad (15)$$

and the process noise is modeled by the diagonal matrix (16).

$$\mathbf{Q}_G = \begin{bmatrix} \sigma_{v_{G,x}}^2 & 0 & 0 \\ 0 & \sigma_{v_{G,y}}^2 & 0 \\ 0 & 0 & \sigma_{v_{G,\theta}}^2 \end{bmatrix} \quad (16)$$

where  $\sigma_{v_{G,x}}^2$ ,  $\sigma_{v_{G,y}}^2$  and  $\sigma_{v_{G,\theta}}^2$  are the variances related to each value of the global state vector.

The global matrix observations obtained by the *a priori* information are shown in Equation (17) for the spherical case. Its derivative represents the global measurement model matrix and is defined in Equation (18):

$$h(\tilde{\mathbf{X}}_{G,k}) = \begin{bmatrix} h_1(\tilde{\mathbf{X}}_{G,k}) \\ \vdots \\ h_M(\tilde{\mathbf{X}}_{G,k}) \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_{G,k}^- - xb_{G,1})^2 + (\hat{y}_{G,k}^- - yb_{G,1})^2 + (z_{MR} - zb_{G,1})^2} \\ \vdots \\ \sqrt{(\hat{x}_{G,k}^- - xb_{G,1})^2 + (\hat{y}_{G,k}^- - yb_{G,1})^2 + (z_{MR} - zb_{G,m})^2} \end{bmatrix} \quad (17)$$

$$\mathbf{H}_{G,k} = \begin{bmatrix} \frac{\partial h_1(\tilde{\mathbf{X}}_{G,k})}{\partial(\tilde{\mathbf{X}}_{G,k})} \\ \vdots \\ \frac{\partial h_M(\tilde{\mathbf{X}}_{G,k})}{\partial(\tilde{\mathbf{X}}_{G,k})} \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1(\tilde{\mathbf{X}}_{G,k})}{\partial(\hat{x}_{G,k}^-)} & \frac{\partial h_1(\tilde{\mathbf{X}}_{G,k})}{\partial(\hat{y}_{G,k}^-)} & \frac{\partial h_1(\tilde{\mathbf{X}}_{G,k})}{\partial(\hat{\theta}_{G,k}^-)} \\ \vdots & \vdots & \vdots \\ \frac{\partial h_M(\tilde{\mathbf{X}}_{G,k})}{\partial(\hat{x}_{G,k}^-)} & \frac{\partial h_M(\tilde{\mathbf{X}}_{G,k})}{\partial(\hat{y}_{G,k}^-)} & \frac{\partial h_M(\tilde{\mathbf{X}}_{G,k})}{\partial(\hat{\theta}_{G,k}^-)} \end{bmatrix} \quad (18)$$

Regarding the hyperbolic case, the global matrix observation is given by Equation (19):

$$h(\tilde{\mathbf{X}}_{G,k}) = \begin{bmatrix} h_1(\tilde{\mathbf{X}}_{G,k}) \\ \vdots \\ h_M(\tilde{\mathbf{X}}_{G,k}) \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_{G,k}^- - xb_{G,2})^2 + (\hat{y}_{G,k}^- - yb_{G,2})^2 + (z_{MR} - zb_{G,2})^2} \\ -\sqrt{(\hat{x}_{G,k}^- - xb_{G,1})^2 + (\hat{y}_{G,k}^- - yb_{G,1})^2 + (z_{MR} - zb_{G,1})^2} \\ \vdots \\ \sqrt{(\hat{x}_{G,k}^- - xb_{G,M})^2 + (\hat{y}_{G,k}^- - yb_{G,M})^2 + (z_{MR} - zb_{G,M})^2} \\ -\sqrt{(\hat{x}_{G,k}^- - xb_{G,1})^2 + (\hat{y}_{G,k}^- - yb_{G,1})^2 + (z_{MR} - zb_{G,1})^2} \end{bmatrix} \quad (19)$$

For the spherical case, as the distance measurements are directly used, we can assume that the noise for each measurement is Gaussian and uncorrelated, so  $\mathbf{R}_G$  is a diagonal matrix as defined in Equation (20), whose diagonal elements are the variance in the distance measurements:

$$\mathbf{R}_G = \begin{bmatrix} \sigma_{w_G}^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_{w_G}^2 \end{bmatrix} \quad (20)$$

For the hyperbolic case, we also assume that the noise is Gaussian, but now, as differences of distances are used, the noise is correlated [17] and  $\mathbf{R}_G$  is the matrix shown in Equation (21):

$$\mathbf{R}_G = \begin{bmatrix} \sigma_{w_G}^2 & 0.5 \cdot \sigma_{w_G}^2 & 0.5 \cdot \sigma_{w_G}^2 \\ 0.5 \cdot \sigma_{w_G}^2 & \ddots & 0.5 \cdot \sigma_{w_G}^2 \\ 0.5 \cdot \sigma_{w_G}^2 & 0.5 \cdot \sigma_{w_G}^2 & \sigma_{w_G}^2 \end{bmatrix} \quad (21)$$

### 3.3. Local Initialization for the LRLPS Area (Lines 13 and 14 in Algorithm 1)

When the MR moves into an LRLPS area, the first iteration of the  $n$ -local vector (22) is obtained using the same procedure as for the GRLPS case. The aim is to minimize the Equations (23) or (24) depending on the case, spherical or hyperbolic, respectively:

$$\hat{\mathbf{X}}_{L,n,0} = [\hat{x}_{L,n,0} \quad \hat{y}_{L,n,0} \quad \hat{\theta}_{L,n,0}]^T \quad (22)$$

$$[\hat{x}_{L,n,0} \quad \hat{y}_{L,n,0}]^T = \arg \min \sum_{m=1}^M (|d_{p_{L,n,0}, b_{L,n,m}} - \hat{d}_{p_{L,n,0}, b_{L,n,m}}|) \quad (23)$$

$$[\hat{x}_{L,n,0} \quad \hat{y}_{L,n,0}]^T = \arg \min \sum_{m=2}^M (|\Delta d_{p_{L,n,0}, b_{L,n,m}} - \Delta \hat{d}_{p_{L,n,0}, b_{G,m}}|) \quad (24)$$

where  $m = 1, 2, \dots, M$  represents each beacon of the LRLPS and the distances are calculated by Equations (25) and (26) in each case.

$$\hat{d}_{p_{L,n,0}, b_{L,n,m}} = \sqrt{(\hat{x}_{L,n,0} - x_{b_{L,n,m}})^2 + (\hat{y}_{L,n,0} - y_{b_{L,n,m}})^2 + (z_{MR} - z_{b_{L,n,m}})^2} \quad (25)$$

$$\Delta \hat{d}_{p_{L,n,0}, b_{L,n,m}} = \sqrt{(\hat{x}_{L,n,0} - x_{b_{L,n,m}})^2 + (\hat{y}_{L,n,0} - y_{b_{L,n,m}})^2 + (z_{MR} - z_{b_{L,n,m}})^2} \quad (26)$$

$$-\sqrt{(\hat{x}_{L,n,0} - x_{b_{L,n,1}})^2 + (\hat{y}_{L,n,0} - y_{b_{L,n,1}})^2 + (z_{MR} - z_{b_{L,n,1}})^2}$$

### 3.4. Local Navigation inside an LRLPS Area (Lines 16 and 17 of Algorithm 1)

If the MR navigates inside a LRLPS area, the local state vector and the covariance matrix are updated with an extended Kalman filter (EKF) particularized to the specific LRLPS and run in parallel to the global one and other LRLPS in the case that the corresponding coverage areas intersect. The Equations are described as follows:

Prediction stage:

$$\hat{\mathbf{X}}_{L,n,k}^- = f(\hat{\mathbf{X}}_{L,n,k-1}, \mathbf{u}_k) \quad (27)$$

$$\mathbf{P}_{L,n,k}^- = \mathbf{A}_{L,n,k} \cdot \mathbf{P}_{L,n,k-1} \cdot \mathbf{A}_{L,n,k}^T + \mathbf{Q}_L \quad (28)$$

Update stage:

$$\mathbf{K} = \mathbf{P}_{L,n,k}^- \cdot \mathbf{H}_{L,n,k}^T \cdot [\mathbf{H}_{L,n,k} \cdot \mathbf{P}_{L,n,k}^- \cdot \mathbf{H}_{L,n,k}^T + \mathbf{H}_{L,n,k}^T + \mathbf{R}_L]^{-1} \quad (29)$$

$$\hat{\mathbf{X}}_{L,n,k} = \hat{\mathbf{X}}_{L,n,k}^- + \mathbf{K} \cdot (\mathbf{Z}_{L,n,k} - h(\hat{\mathbf{X}}_{L,n,k}^-)) \quad (30)$$

$$\mathbf{P}_{L,n,k} = (\mathbf{I} - \mathbf{K} \cdot \mathbf{H}_{L,n,k}) \cdot \mathbf{P}_{L,n,k}^- \tag{31}$$

The *a priori* information of the *n*-local state vector is defined by Equation (32), in which, we have used the odometry information ( $\Delta d_{odo}$ ,  $\Delta \theta_{odo}$ ) in order to estimate this state:

$$\hat{\mathbf{X}}_{L,n,k}^- = \begin{bmatrix} \hat{x}_{L,n,k-1} + \Delta d_{odo} \cdot \cos(\hat{\theta}_{L,n,k-1} + \Delta \theta_{odo}) \\ \hat{y}_{L,n,k-1} + \Delta d_{odo} \cdot \sin(\hat{\theta}_{L,n,k-1} + \Delta \theta_{odo}) \\ \hat{\theta}_{L,n,k-1} + \Delta \theta_{odo} \end{bmatrix} \tag{32}$$

The Expression (33) is obtained taking the derivative of the matrix (32) with respect to each one of the local state variables:

$$\mathbf{A}_{L,n,k} = \begin{bmatrix} \frac{\partial f(\hat{\mathbf{X}}_{L,n,k}^-, \mathbf{u}_k)}{\partial \hat{x}_{L,n,k}} \\ \frac{\partial f(\hat{\mathbf{X}}_{L,n,k}^-, \mathbf{u}_k)}{\partial \hat{y}_{L,n,k}} \\ \frac{\partial f(\hat{\mathbf{X}}_{L,n,k}^-, \mathbf{u}_k)}{\partial \hat{\theta}_{L,n,k}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta d_{odo} \cdot \sin(\hat{\theta}_{L,n,k-1} + \Delta \theta_{odo}) \\ 0 & 1 & \Delta d_{odo} \cdot \cos(\hat{\theta}_{L,n,k-1} + \Delta \theta_{odo}) \\ 0 & 0 & 1 \end{bmatrix} \tag{33}$$

The local covariance matrix (34) represents the variance for each local state variable and is updated in every iteration of the local EKF:

$$\mathbf{P}_{L,n,k} = \begin{bmatrix} \sigma_{x_{L,n}}^2 & \sigma_{x_{L,n},y_{L,n}}^2 & \sigma_{x_{L,n},\theta_{L,n}}^2 \\ \sigma_{y_{L,n},x_{L,n}}^2 & \sigma_{y_{L,n}}^2 & \sigma_{y_{L,n},\theta_{L,n}}^2 \\ \sigma_{\theta_{L,n},x_{L,n}}^2 & \sigma_{\theta_{L,n},y_{L,n}}^2 & \sigma_{\theta_{L,n}}^2 \end{bmatrix} \tag{34}$$

Regarding the process noise, it is modeled by a diagonal matrix (35):

$$\mathbf{Q}_{L,n} = \begin{bmatrix} \sigma_{v_{L,n,x}}^2 & 0 & 0 \\ 0 & \sigma_{v_{L,n,y}}^2 & 0 \\ 0 & 0 & \sigma_{v_{L,n,\theta}}^2 \end{bmatrix} \tag{35}$$

The local matrix observations obtained by the *a priori* information is shown in Equation (36) for the spherical case. Its derivative represents the global measurement model matrix and is defined in Equation (37):

$$h(\hat{\mathbf{X}}_{L,n,k}^-) = \begin{bmatrix} h_1(\hat{\mathbf{X}}_{L,n,k}^-) \\ \vdots \\ h_M(\hat{\mathbf{X}}_{L,n,k}^-) \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_{L,n,k}^- - x_{b_{L,n,1}})^2 + (\hat{y}_{L,n,k}^- - y_{b_{L,n,1}})^2 + (z_{MR} - z_{b_{L,n,1}})^2} \\ \vdots \\ \sqrt{(\hat{x}_{L,n,k}^- - x_{b_{L,n,m}})^2 + (\hat{y}_{L,n,k}^- - y_{b_{L,n,m}})^2 + (z_{MR} - z_{b_{L,n,m}})^2} \end{bmatrix} \tag{36}$$

$$\mathbf{H}_{L,n,k} = \begin{bmatrix} \frac{\partial h_1(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{\mathbf{X}}_{L,n,k}^-)} \\ \vdots \\ \frac{\partial h_M(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{\mathbf{X}}_{L,n,k}^-)} \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{x}_{L,n,k}^-)} & \frac{\partial h_1(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{y}_{L,n,k}^-)} & \frac{\partial h_1(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{\theta}_{L,n,k}^-)} \\ \vdots & \vdots & \vdots \\ \frac{\partial h_M(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{x}_{L,n,k}^-)} & \frac{\partial h_M(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{y}_{L,n,k}^-)} & \frac{\partial h_M(\hat{\mathbf{X}}_{L,n,k}^-)}{\partial(\hat{\theta}_{L,n,k}^-)} \end{bmatrix} \quad (37)$$

In the hyperbolic case, the observation matrix is defined by Equation (38):

$$h(\hat{\mathbf{X}}_{L,n,k}^-) = \begin{bmatrix} h_1(\hat{\mathbf{X}}_{L,n,k}^-) \\ \vdots \\ h_M(\hat{\mathbf{X}}_{L,n,k}^-) \end{bmatrix} = \begin{bmatrix} \sqrt{(\hat{x}_{L,n,k}^- - xb_{L,n,2})^2 + (\hat{y}_{L,n,k}^- - yb_{L,n,2})^2 + (z_{MR} - zb_{L,n,2})^2} \\ -\sqrt{(\hat{x}_{L,n,k}^- - xb_{L,n,1})^2 + (\hat{y}_{L,n,k}^- - yb_{L,n,1})^2 + (z_{MR} - zb_{L,n,1})^2} \\ \vdots \\ \sqrt{(\hat{x}_{L,n,k}^- - xb_{L,n,M})^2 + (\hat{y}_{L,n,k}^- - yb_{L,n,M})^2 + (z_{MR} - zb_{L,n,M})^2} \\ -\sqrt{(\hat{x}_{L,n,k}^- - xb_{L,n,1})^2 + (\hat{y}_{L,n,k}^- - yb_{L,n,1})^2 + (z_{MR} - zb_{L,n,1})^2} \end{bmatrix} \quad (38)$$

For the spherical case, the observation noise is Gaussian and uncorrelated Equation (39):

$$\mathbf{R}_{L,n} = \begin{bmatrix} \sigma_{w_{L,n}}^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_{w_{L,n}}^2 \end{bmatrix} \quad (39)$$

In the hyperbolic case, the noise is Gaussian, but correlated, and it can be simplified as indicated in Expression (40) according to [17]:

$$\mathbf{R}_{L,n} = \begin{bmatrix} \sigma_{w_{L,n}}^2 & 0.5 \cdot \sigma_{w_{L,n}}^2 & 0.5 \cdot \sigma_{w_{L,n}}^2 \\ 0.5 \cdot \sigma_{w_{L,n}}^2 & \ddots & 0.5 \cdot \sigma_{w_{L,n}}^2 \\ 0.5 \cdot \sigma_{w_{L,n}}^2 & 0.5 \cdot \sigma_{w_{L,n}}^2 & \sigma_{w_{L,n}}^2 \end{bmatrix} \quad (40)$$

### 3.5. Global Navigation in a LRLPS Area (Line 20 of Algorithm 1)

One of the major challenges of this algorithm is to solve the case in which the MR is navigating inside LRLPS areas and the position of the MR is required according to the global reference system. In this case, the global vector Equation (43) is updated using the increments of distance and angle between the current point ( $k$ ) Equation (41) and the previous point ( $k-1$ ) Equation (42) of the local state vector. In the case of having the coverage of several LRLPSs, the global state vector is updated considering only the one which has the least trace of its covariance matrix.

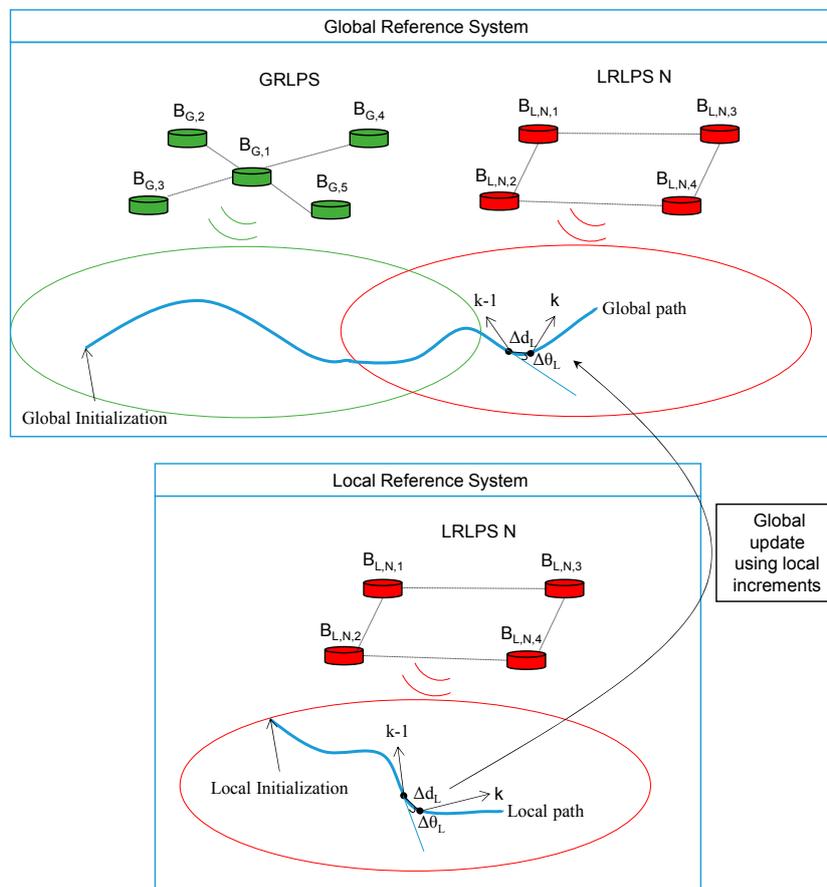
$$\Delta \hat{d}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})} = \sqrt{(\hat{x}_{L,n,k} - x_{L,n,k-1})^2 + (\hat{y}_{L,n,k} - y_{L,n,k-1})^2} \quad (41)$$

$$\Delta \hat{\theta}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})} = \sqrt{(\hat{\theta}_{L,n,k} - \hat{\theta}_{L,n,k-1})^2} \quad (42)$$

$$\hat{\mathbf{X}}_{G,k} = \begin{bmatrix} \hat{x}_{G,k-1} + \Delta \hat{d}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})} \cdot \cos(\hat{\theta}_{G,k-1} + \Delta \hat{\theta}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})}) \\ \hat{y}_{G,k-1} + \Delta \hat{d}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})} \cdot \sin(\hat{\theta}_{G,k-1} + \Delta \hat{\theta}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})}) \\ \hat{\theta}_{G,k-1} + \Delta \hat{\theta}_{L|n=\min \text{trace}(\mathbf{P}_{L,n,k})} \end{bmatrix} \quad (43)$$

Figure 3 shows graphically how the global positioning (related to the global reference system) is performed from the local positioning (obtained regarding the local reference system of an LRLPS). The filter that computes the local positioning algorithm gives positions that are translated and rotated regarding to the global reference system, although the increments of distance and angle between two successive instants of time can be translated from the local path to the global one.

**Figure 3.** Diagram that shows the update of the global position from the increments of distance and angle between two consecutive instants of time in the local reference system.



### 3.6. Global Navigation with Only Odometry (Lines 22 and 23 in Algorithm 1)

Of course, when the MR is navigating without any coverage from an LRLPS or a GRLPS, the global state vector is updated using only the odometry information, and the covariance matrix is equal to the *a priori* estimation, as was shown in Equation (13).

Finally, in the cases in which the robot is covered partly by GRLPS and partly by LRLPS, the global state vector is updated using the global beacons. In parallel, the local state vector for the particular LRLPS is updated using the measurements from the local beacons (to be used just in case of losing the global coverage).

#### 4. Simulated Results

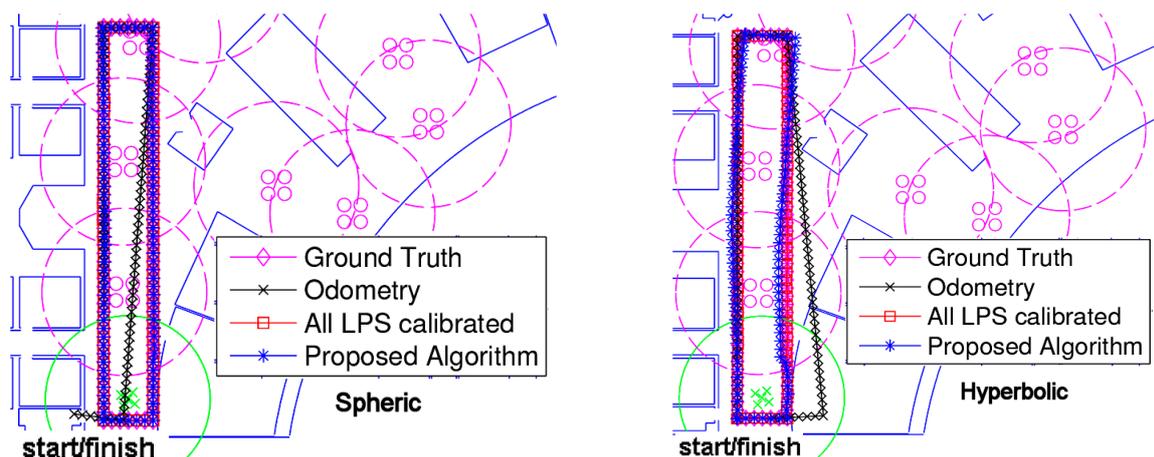
In order to evaluate the performance of the algorithm, different trajectories have been simulated using the workspace shown in Figure 2. We have compared the proposed algorithm based on GRLPS and LRLPS to a case in which all LPSs were previously calibrated. Results are simulated for both the spherical and the hyperbolic cases.

In the first test, we simulated a trajectory where the MR follows a rectangular path. In this simulation, we have introduced Gaussian noises for the odometry and for the measurement of distances. The typical deviation of the odometry error for the MR increment of distance has been set to zero, while for the increment of angle, we have used a relatively high value (compared with real systems) in order to evaluate the effect of errors when the MR rotates in the corners of the rectangular path. Finally, for distance measurements, we have used a typical error of 1 cm, obtained from real measurements in an Ultrasonic Local Positioning System (ULPS), as seen in Equation (44):

$$\sigma_{\Delta d} = 0 \text{ m}, \quad \sigma_{\Delta\theta} = 0.07 \text{ rad}, \quad \sigma_{w_G} = \sigma_{w_{L,n}} = 0.01 \text{ m} \quad (44)$$

Figure 4 shows the trajectory with only odometry information (lines in black), the ground truth of the trajectory (lines in magenta), the navigation estimation supposing all beacons previously calibrated (lines in red) and the navigation estimation using the proposed algorithm (lines in blue). Obviously, the navigation errors supposing all LPS calibrated are less than the ones using the proposed algorithm, but in the last case, better than the case with only odometry information. The performance of each method can be better seen in Figure 4, where the accumulative mean error evolution is shown.

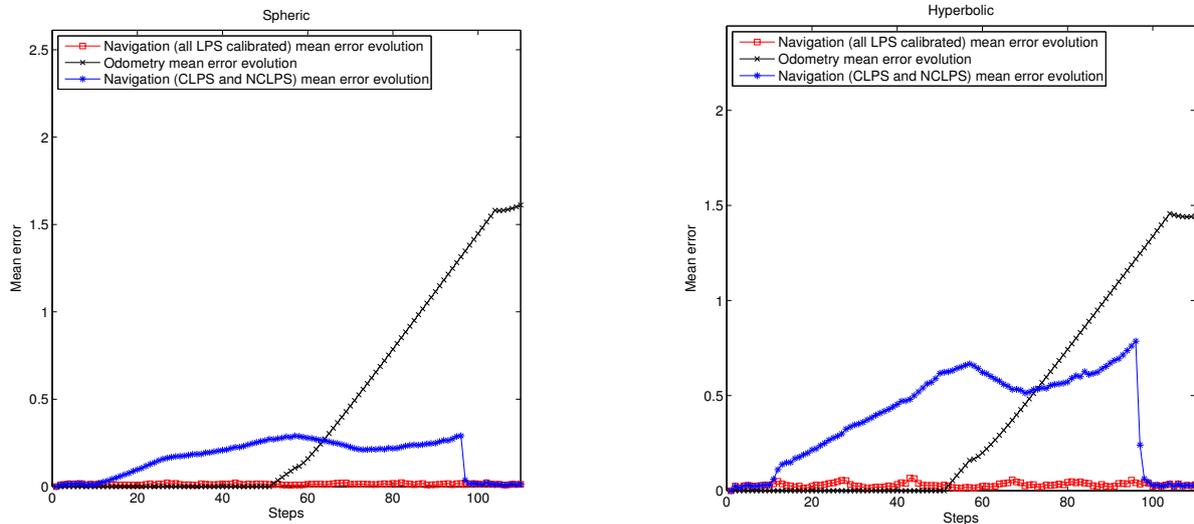
**Figure 4.** Evaluation of the algorithm for a rectangular trajectory. **(Left)** Spherical case; **(right)** hyperbolic case.



In order to evaluate the accumulative error, we have simulated 50 iterations per step of the trajectory assuming the errors described in Equation (44). Note that in Figure 4, only one of these iterations was represented. After all of the iterations, the simulation results are shown in Figure 5, where Step 1 corresponds to the starting point and Step 110 to the finishing point. It can be observed that the navigation errors assuming all LPS calibrated are close to zero and are not accumulative. The odometry has an accumulative error that gets higher right after the first turning of the MR (it is very sensitive to angle errors). Concerning the proposed algorithm, it has a relatively low accumulative

error, even when the MR navigates inside LRLPS areas (less than 40 cm for the spherical case and double for the hyperbolic).

**Figure 5.** Accumulative mean error evolution. (Left) Spherical case; (right) hyperbolic case.

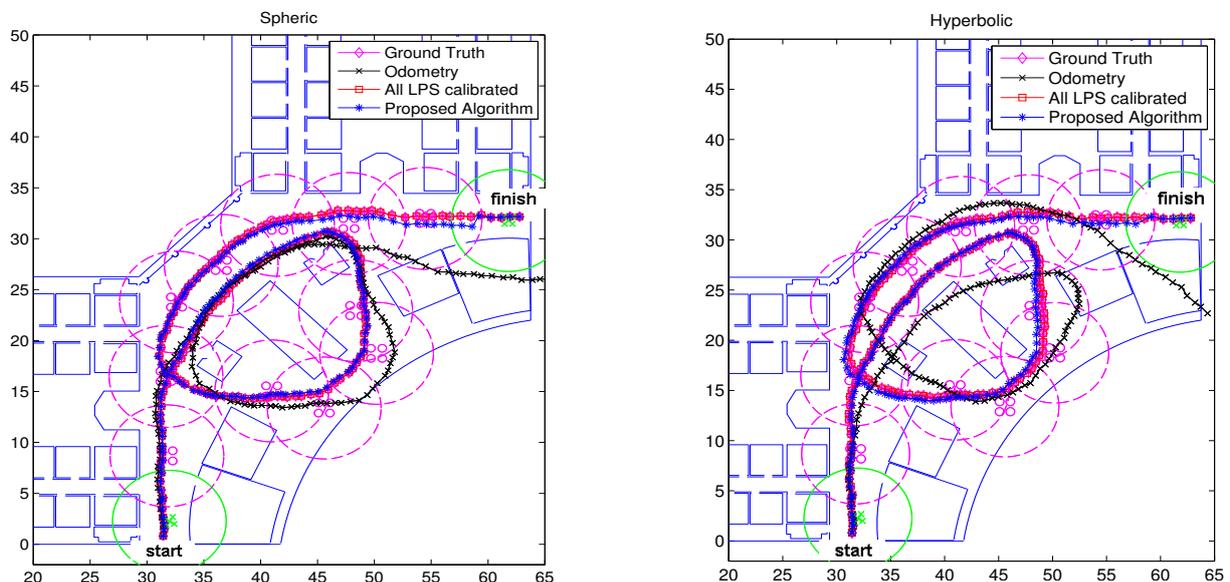


Let us pay attention to the navigation error evolution when the MR returns to GRLPS at the end of the trajectory. This happens roughly around Step 100. The error figures go suddenly to very low values again. This is because of the reduced absolute errors that can be obtained under the GRLPS coverage areas [10].

In the second test, an arbitrary route was defined to be followed by the MR. In this case, the odometry errors have been changed, considering typical errors obtained for our real MR and ultrasonic distance measurements based on empirical tests Equation (45):

$$\sigma_{\Delta d} = 0.07 \text{ m}, \quad \sigma_{\Delta\theta} = 0.05 \text{ rad}, \quad \sigma_{w_G} = \sigma_{w_{L,n}} = 0.01 \text{ m} \quad (45)$$

**Figure 6.** Simulated trajectory. (Left) Spherical case; (right) hyperbolic case.

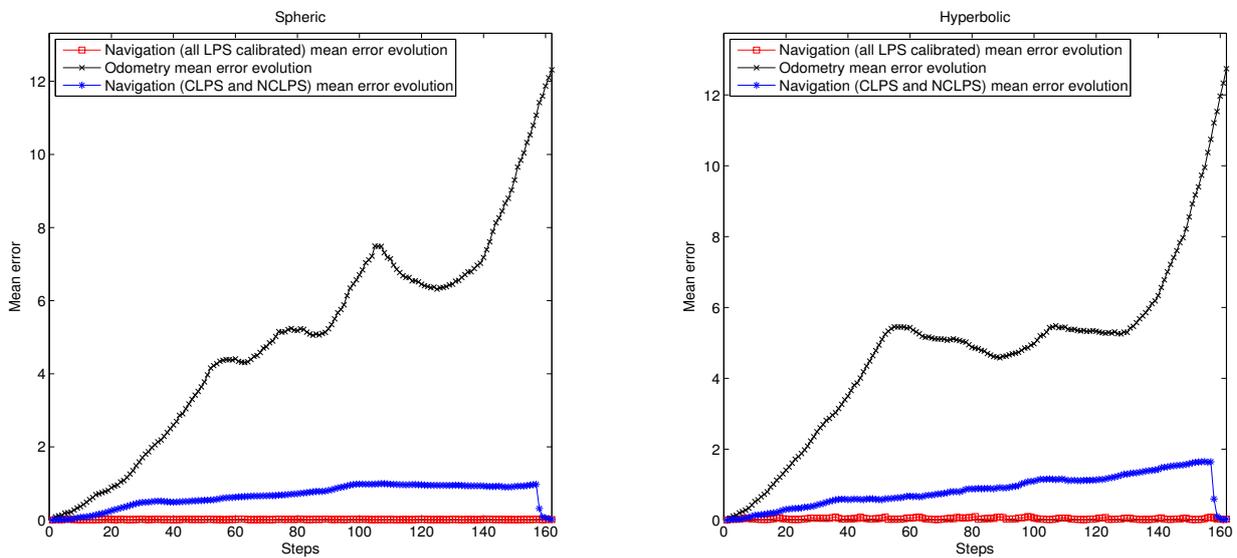


The defined trajectory covers almost all of the working area, and it was thought to be a practical route.

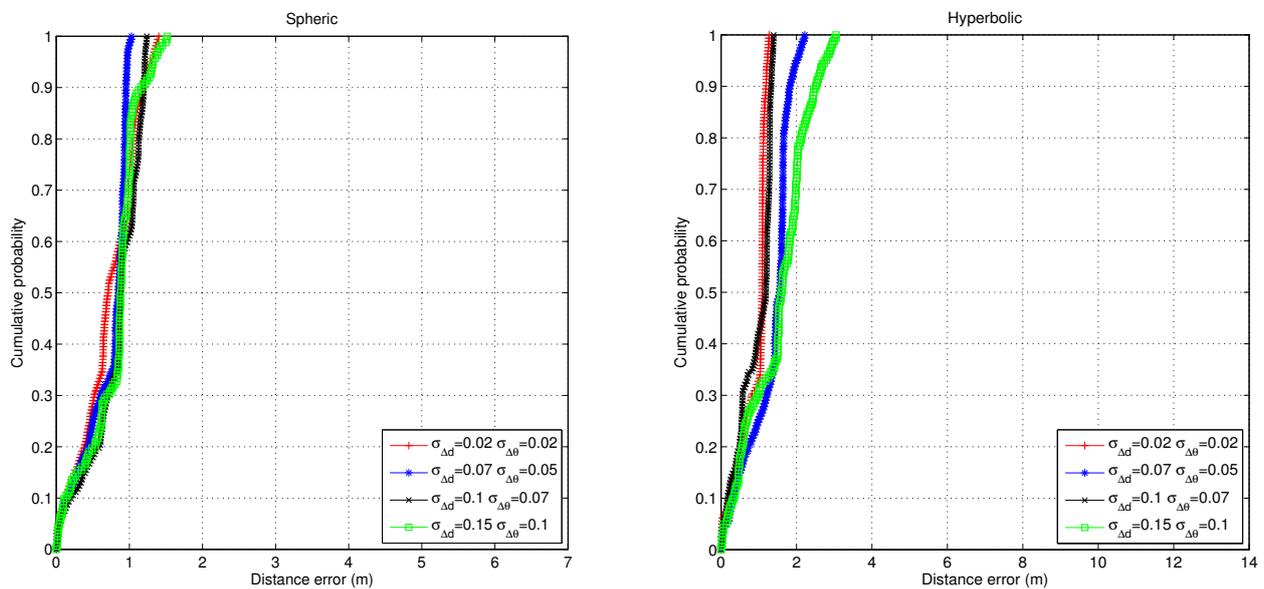
Figure 6 shows the results to compare the performance between the proposed algorithm using GRLPS and LRLPS and the case in which all of the LPSs were calibrated, as well as the case with only odometry.

Figure 7 shows the accumulative error for the trajectory in this simulation. Now, the trajectory is longer than in the previous simulation, and the accumulative error for the case of using only odometry increases until more than ten meters; while for the proposed algorithms, it remains below 1 m (and better for the spherical case). Finally, Figures 8 and 9 show the cumulative distribution function (cdf) in different error conditions.

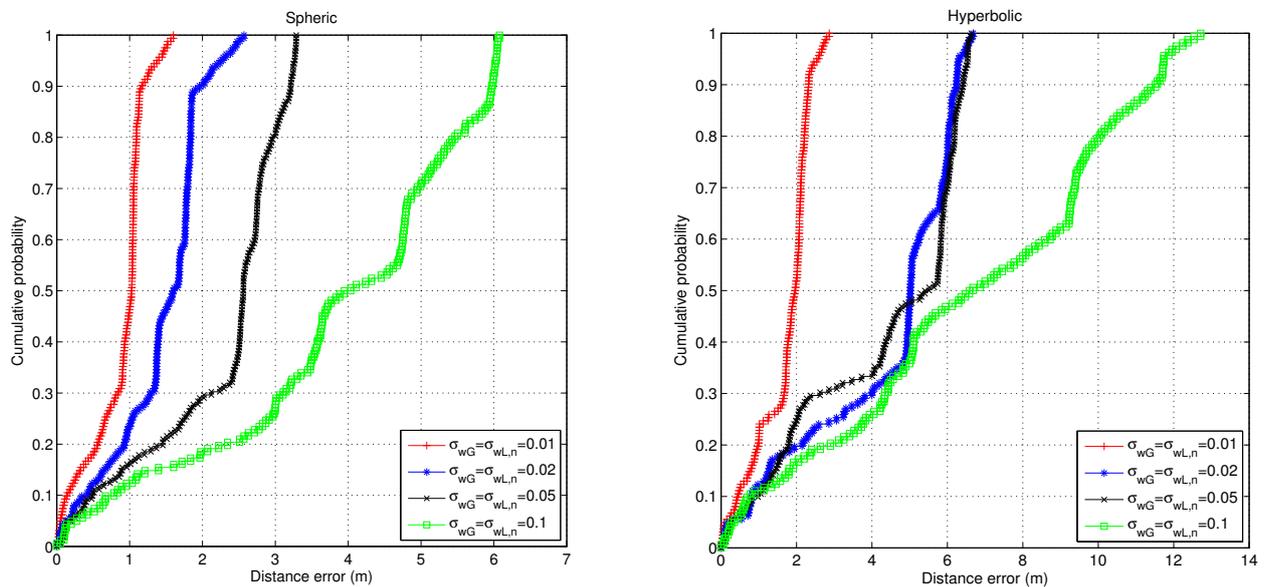
**Figure 7.** Accumulative mean error evolution for trajectories in Figure 6. **(Left)** Spherical case; **(right)** hyperbolic case.



**Figure 8.** cdf of the navigation errors for different odometry error conditions assuming that the ultrasonic distance errors remain constant. **(Left)** Spherical case; **(right)** hyperbolic case.



**Figure 9.** cdf of the navigation errors for different ultrasonic distance error conditions assuming that the odometry errors remain constant. **(Left)** Spherical case; **(right)** hyperbolic case.

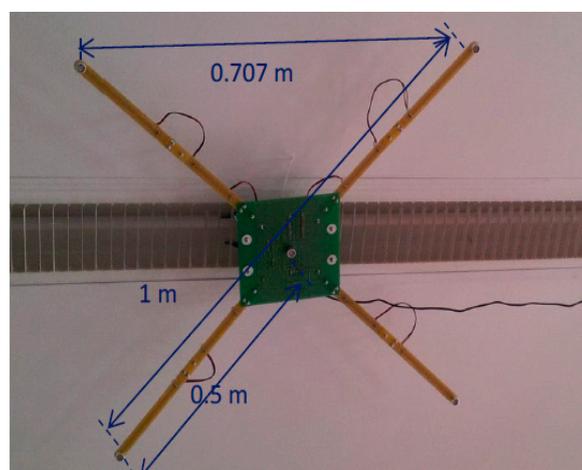


For the first test, it can be observed that the variation of odometry errors does not degrade the system performance very much when the ultrasonic distance errors remain constant (Figure 7), whereas when the odometry errors are constant and the ultrasonic distance errors change, the navigation errors increase considerably, especially in the hyperbolic case. Note that in any case, the results for the hyperbolic trilateration show twice the errors that are obtained using the spherical trilateration.

## 5. Real Results

In order to evaluate the algorithm behavior in a real scenario, we have used two ultrasonic LPSs. One of them will work as a GRLPS, while the other will operate as an LRPLS. The layout and hardware of both LPSs are the same. Each LPS has the structure shown in Figure 10.

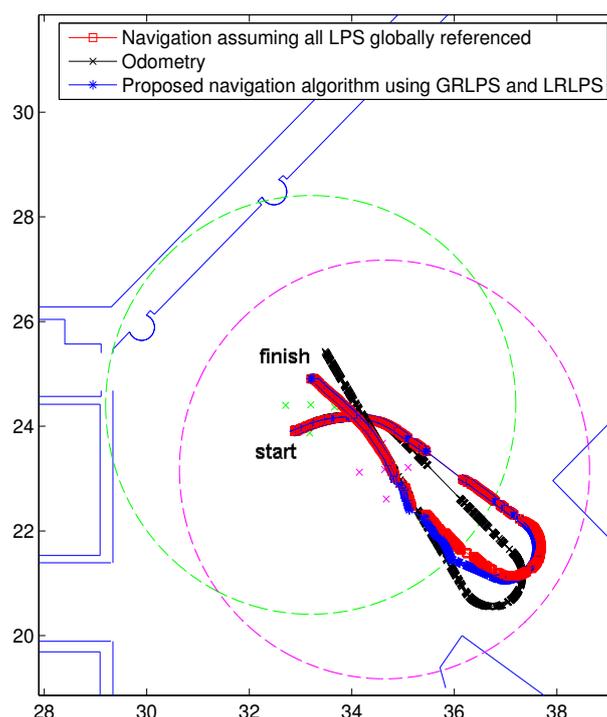
**Figure 10.** Structure of a real ULPS.



There are five emitters (beacons) placed, four of them on the corners of a square and the fifth one in its center. All of them emit a 1023-bit Kasami code in order to increase the detection performance. Full details of the system are explained in [10].

Figure 11 shows the test trajectory performed with a Pioneer 3-DX MR [18]. The coverage area of the GRLPS is shown in green, while the coverage area of the LRLPS is shown in magenta. To estimate the MR location, it is only possible to use the hyperbolic case, since there is no synchronization between the beacons and the receiver.

**Figure 11.** Real results with a GRLPS and an LRLPS.



In this Figure 11, the red line (with squares) is the estimated path assuming all LPSs were globally referenced. As the position errors in this case are close to the ground truth, this path will be used as the reference one to be compared with the results obtained in other conditions. The black line (with crossings) represents the trajectory described by the odometry of the robot, while the blue line (with asterisks) illustrates the results obtained using our proposed algorithm. It can be observed that using this algorithm, the estimated trajectory is close to the one assuming that all LPS were calibrated.

## 6. Use of the System for Human Tracking

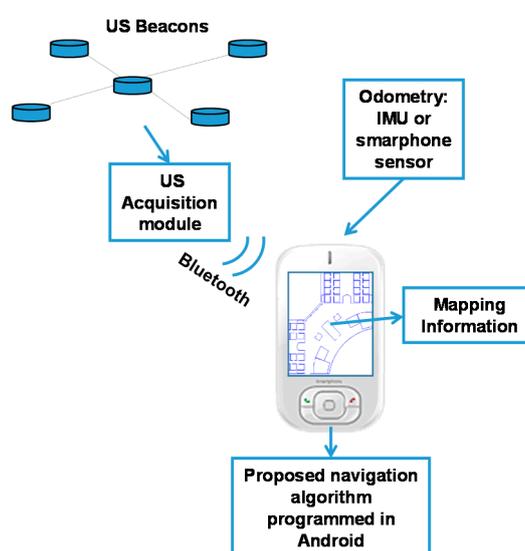
The approach presented in this paper is suitable for human navigation, provided that the equipment is conveniently adapted.

Regarding the US reception stage, the MR uses a receiver system developed by Avisoft [19] connected to a laptop. However, this receiver is too huge to be carried by a person, and thus, in [20] we proposed an ultrasonic signal acquisition module for mobile devices (smartphones, tablets).

Regarding the odometry data obtained from people walking, different approaches have been published. For instance, taking advantage of the smartphone sensors, some authors, as in [21], propose a pedestrian dead reckoning technique based on the accelerometer sensor, which permits the recognition of some step patterns. On the other hand, Inertial Measurement Units (IMUs) allow getting better estimations than smartphones; although the equipment required is more intrusive. Each IMU is usually composed of three accelerometers and three gyroscopes and is mounted on the foot. An example of this approach can be found in [22].

As future work, we intend to use the ultrasonic signal acquisition module and a pedestrian dead reckoning method in order to apply the navigation algorithm presented in this paper for human tracking and navigation. A major challenge will be the adaptation of our algorithm to a smartphone due to real-time restrictions. Figure 12 shows a scheme of the adaptation for human tracking.

**Figure 12.** Proposed scheme for human tracking.



## 7. Conclusions

In this paper, we have proposed a novel algorithm that allows navigating with increased performance inside an extensive area covered by a set of LPSs, some of them with a global reference (GRLPSs) and most with a local reference system (LRLPSs). The main advantage of the proposed algorithm is the possibility of navigating in extensive indoor environments without calibration of all of the needed LPSs, useful in cases of changing environments or when a fast installation is required.

In the case that the MR navigates inside an LRLPS coverage area, the accumulative position error increases slowly (compared with the case of only using odometry), and usually, it can be admissible for MR indoor navigation; this error is reset (decreases to a low value) when the MR is inside a GRLPS coverage area. The performance of the system has been tested with different simulated and real trajectories. Finally, we have studied the possibility to adapt the system to human tracking as future work.

## Acknowledgments

This work was possible thanks to the Spanish Ministry of Economy and Competitiveness (LORIS project, Ref. TIN2012-38080-C04-01) and the PhD fellowship program of University of Alcalá (Madrid).

## Author Contributions

David Gualda, Jesús Ureña and Juan C. García designed the proposal; David Gualda programmed the algorithm and obtained simulation results; Alejandro Lindo contributed obtaining and analyzing real results and the paper was written by David Gualda, Jesús Ureña and Juan C. García.

## Conflicts of Interest

The authors declare no conflicts of interest.

## References

1. Want, R.; Hopper, A.; Falcao, V.; Gibbons, J. The active badge location system. *ACM Trans. Inf. Syst. (TOIS)* **1992**, *10*, 91–102.
2. Priyantha, N.B.; Miu, A.K.L.; Balakrishnan, H.; Teller, S. The cricket compass for context-aware mobile applications. In Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, Rome, Italy, 16–21 July 2001; pp. 1–14.
3. Hazas, M.; Hooper, A. Broadband ultrasonic location system for improved indoor positioning. *IEEE Trans. Mob. Comput.* **2006**, *5*, 536–547.
4. Medina, C.; Segura, J.C.; de la Torre, Á. Ultrasound Indoor Positioning System Based on a Low-Power Wireless Sensor Network Providing Sub-Centimeter Accuracy. *Sensors* **2013**, *13*, 3501–3526.
5. Ladd, A.M.; Bekris, K.E.; Marceau, G.; Rudys, A.; Wallach, D.S.; Kavraki, E.E. Using wireless Ethernet for localization. *Intell. Robots Syst.* **2002**, *1*, 402–408.
6. Tarrío, P.; Bernardos, A.M.; Casar, J.R. Weighted Least Squares Techniques for Improved Received Signal Strength Based Localization. *Sensors* **2011**, *11*, 8569–8592.
7. Boonsriwai, S.; Apavatjirut, A. Indoor WIFI localization on mobile devices. In Proceedings of the 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Krabi, Thailand, 15–17 May 2013; pp. 1–5.
8. Kartowisastro, I.H. Local Positioning System of a Mobile Robot: A Practical Perspective. In Proceedings of the 9th International Conference on Control, Automation, Robotics and Vision, (ICARCV '06), Singapore, 5–8 December 2006; pp. 1–6.
9. Villadangos, J.M.; Ureña, J.; García, J.J.; Mazo, M.; Hernández, Á.; Jiménez, A.; Ruíz, D.; Marziani, C.D. Measuring Time-of-Flight in an Ultrasonic LPS System Using Generalized Cross-Correlation. *Sensors* **2011**, *11*, 10326–10342.
10. Ruiz, D.; Garcia, E.; Ureña, J.; de Diego, D.; Gualda, D.; Garcia, J.C. Extensive Ultrasonic Local Positioning System for navigating with mobile robots. In Proceedings of the 10th Workshop on Positioning Navigation and Communication (WPNC), Dresden, Germany, 20–21 March 2013; pp. 1–6.

11. Ward, A.; Jones, A.; Hopper, A. A new location technique for the active office. *IEEE Pers. Commun.* **1997**, *4*, 42–47.
12. Deffenbaugh, M.; Bellingham, J.G.; Schmidt, H. The relationship between spherical and hyperbolic positioning. In Proceedings of the MTS/IEEE on Prospects for the 21st Century (OCEANS '96), Fort Lauderdale, FL, USA, 23–26 September 1996; Volume 2, pp. 590–595.
13. Kalman, R.E. A New Approach to Linear Filtering and Prediction Problems. *J. Basic Eng.* **1960**, *82*, 35–45.
14. Wan, E.A.; van der Merwe, R. The unscented Kalman filter for nonlinear estimation. In Proceedings of the Adaptive Systems for Signal Processing, Communications, and Control Symposium, Lake Louise, AB, Canada, 1–4 March 2000; pp. 153–158.
15. Simon, D. *To Game Theory Approach to Constrained Minimax State Estimation*; Technical Report; Department of Electrical Engineering, Cleveland State University: Cleveland, OH, USA, 2005.
16. Ristic, B.; Arulampalam, S.N. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*; Artech House: London, UK, 2004.
17. Chan, Y.T.; Ho, K.C. A simple and efficient estimator for hyperbolic location. *IEEE Trans. Signal Process.* **1994**, *42*, 1905–1915.
18. Pioneer 3-DX. Available online: <http://www.mobilerobots.com/ResearchRobots/PioneerP3DX.aspx> (accessed on 10 September 2014).
19. Avisoft Bioacoustics, Ultrasound Gate 116 Hm Product Specification. Available online: <http://www.avisoft.com/usg/usg116hm.htm>, 2013 (accessed on 10 September 2014).
20. Lindo, A.; Perez, M.C.; Urena, J.; Gualda, D.; Garcia, E.; Villadangos, J.M. Ultrasonic signal acquisition module for smartphone indoor positioning. In Proceedings of the 19th IEEE International Conference on Emerging Technologies and Factory Automation, Barcelona, Spain, 16–19 September 2014; pp. 1–6.
21. Pratama, A.R.; Widyawan; Hidayat, R. Smartphone-based Pedestrian Dead Reckoning as an indoor positioning system. In Proceedings of the 2012 International Conference on System Engineering and Technology (ICSET), Bandung, Indonesia, 11–12 September 2012; pp. 1–6.
22. Jimenez, A.R.; Seco, F.; Zampella, F.; Prieto, J.C.; Guevara, J. PDR with a Foot-Mounted IMU and Ramp Detection. *Sensors* **2011**, *11*, 9393–9410.