

RF-based Relative Position Estimation in Mobile Ad-Hoc Networks with Confidence Regions

Luis Oliveira and Luis Almeida

IT/Faculdade de Engenharia
Universidade do Porto, Portugal

Abstract. Relative localisation of mobile robots can provide useful information to applications, from formation control, to joint exploration and inspection. One way to obtain relative localisation is to measure distances between the multiple robots. In this scope, distance estimates based on RF ranging data can be beneficial for small/inexpensive communicating robots that have no other means of measuring distances, or as disambiguation of multiple hypothesis in high accuracy localisation systems. In this work, we present a technique of estimating the relative positions of simple mobile robots in a small team using the distance information that can be captured by a wireless transceiver, only. Simulation results with a team of five mobile robots show that we can estimate their relative positions with an average accuracy of 1.3 meters without any fixed reference and using RF information, only. The main contribution of our work is that we can provide consistent reliability information as the covariance of the obtained positions.

Keywords: Relative localisation, RF-ranging, Multidimensional Scaling.

1 Introduction

Accurate localisation is a key factor on most mobile robots applications which, due to the need of interaction with the world, has been mostly focused on absolute localisation. Particularly, substantial attention has been given to simultaneous localisation and mapping (SLAM) algorithms, using sensors as diverse as vision [15] [1], lasers [19], or a combination of those [20], usually fused with dead reckoning. Such sensors provide high precision measurements, consequently they are used to provide high precision localisation. Despite that, lasers are bulky (for example to be carried by quad-rotors) and vision systems require high computational capacity. A viable alternative is to use external systems such as GPS or previously built infrastructures, however, the first may be undesirable since it is only available in outdoor environments and may be rendered unusable in forests and street canyons. Similarly, the latter are usually an undesired alternative, mainly because they are costly to deploy, and are useless if a catastrophic event changes the environment significantly.

Alternatively it is possible to localise robots relatively to each other, and although this type of localisation may seem insufficient to support interactions with the environment, in multi-robot systems it is of key importance to support fusion of information

from different robots, allowing robot cooperation/coordination as well as improvement of global localisation, e.g. using lasers to measure distance and angle to other robots and fusing that information with dead reckoning [21], or using relative positions to fuse SLAM maps [3].

In this work, we propose a relative position estimation technique that can be employed to localise small (approximately up to 10 elements) dynamic multi-robot teams. The sensing is performed by the wireless communications interface, which reports the Received Signal Strength Indicator (RSSI) and is able to measure the Round-Trip Time-of-Flight. No extra hardware is necessary. Despite the fact that RSSI readings are unreliable, they can still provide helpful information about the distance between robots and are unequivocal information since the robot can know who it is “talking” with. The main contributions of this work are:

- we provide a novel relative position tracking technique using RF-based ranging estimates, only
- we make no assumption on the dynamics of the robots, i.e. all robots are considered mobile with unknown velocity
- we provide positioning reliability information as the covariance of the positions

2 Related work and proposal

Relative localisation of mobile robots (or more generically mobile nodes) can provide useful information to applications. Specifically, estimates of the topology of a network based on collected ranging data, can be beneficial to applications as diverse as coarse formation control, cooperative sensing and area coverage, and disambiguation of multiple hypothesis in high accuracy localisation systems.

A common approach to robot localisation is SLAM. A famous example is the Fast-SLAM algorithm [11], where a particle filter is employed to track several possible paths of the robot, and extended Kalman filters to estimate the positions of landmarks. Another example is Wifi-SLAM [8] where an automatic fingerprinting technique that exploits landmarks on the radio map is proposed. By fusing RSSI information with IMU data, it is possible to detect loop closure and to build the environment map and locate the user. Another interesting work is presented in [6], where authors propose a range only SLAM. The work in [6] proposes a technique using dead reckoning to track robot movement and ultrasound ranging equipment to measure the distance between itself and some beacons. The beacons have unknown positions but are able to measure the distance between themselves. MapCraft [24] assumes that a physical map containing the walls and doors is already available. Then, data from different sensors is fused and matched to the map to estimate positions. Despite all that, these techniques require the ability of measuring some static features, such as landmarks, walls, or the RSSI fingerprints of certain access-points. Conversely, in the work presented in [16] a method of estimating the positions of moving nodes in an anchor-less scenario is proposed. The authors use relative velocity, calculated based on dead-reckoning, together with RSSI measurements to provide a position estimate of a team of mobile nodes without resorting to anchor nodes.

Another very popular solution of calculating nodes position from ranging information is the MultiDimensional Scaling (MDS) algorithm [2, 4] that minimises the dissimilarities of a connectivity matrix up to a rigid formation. In order to improve results under unknown line-of-sight/non-line-of-sight (LOS/NLOS) conditions and scarce ranging information, [5] uses another variant of MDS based on Weighted Least Squares algorithm, whose weights are assigned according to the reliability of the ranging measurements. The work of [17] proposes a method of estimating nodes positioning using a Maximum Likelihood Estimator (MLE). RSSI and ToA measurements of static nodes are collected using the location of four anchor nodes. In order to obtain distance measurements from RSSI values, the parameters of the path loss model are estimated using data collected prior to experiments. However, assuming known position of anchor nodes is undesirable, since they can be unavailable in many scenarios. Therefore, in [10, 14], the authors propose a method for deriving the network topology from the RSSI data using MDS. The method presented is not a physical accurate localisation system, mainly because the work does not consider any propagation model. Other approaches involve iterative methods, such as the work in [25] that proposes solving an expectation-maximisation problem that jointly estimates the path loss model and the relative positions of the nodes, using MDS as the initial condition. Similarly, the authors in [22] propose to use a gradient descent algorithm to solve a minimisation problem that finds the topology that minimises the error of the distance, but this problem has non-linear constraints and many local minima. Finally, the technique of communication-based relative localisation has been also been extensively studied in the field of wireless sensor networks (WSN), such as [9][5][17]. However, the assumptions valid in most WSN scenarios can seldom be extended to mobile robot, particularly the absence of, or very limited, mobility that simplifies the problem.

Our paper differs from the previously referred works in several aspects. First of all, we do not use sensors other than the RF transceiver module to compute approximate relative positions between the nodes of a small team of mobile robots. Moreover, we make no assumption on the dynamics of the robots, and do not assume pre-installed anchor nodes or robots with known positions. Finally, in addition to the positions estimate, we provide an estimate of the covariance of the positions, allowing the user to define confidence regions around the estimate. To the best of the authors knowledge, this is one of the first works computing confidence regions associated to MDS-computed position estimates.

3 Estimating the positions

In this section we explain the details of our proposal. First of all we explain the data that is collected from the robots, and how we use it to obtain a measurement of the positions. Then we explain how we filter that data and generate positions estimates.

3.1 Collecting distance information from robots

The work presented in [13] proposes an RF-only, anchor-less technique that performs online estimation of the distance between mobile robots without previous knowledge.

The authors use RSSI/Time-of-Flight measurements to perform online estimation of the path loss model. Then, the corresponding model is used to estimate the distance using the RSSI, or the RSSI and Time-of-Flight when available. Moreover, an Extended Kalman Filter (EKF) was used to perform the estimation. Consequently, the result is both the estimated distance between robots and the corresponding estimated variance. Despite that, the work in [13] approaches the problem we aim to solve up to the estimation of distances, only. The details of that work cannot be included here due to space constraints, but Fig. 1 explains the respective process.

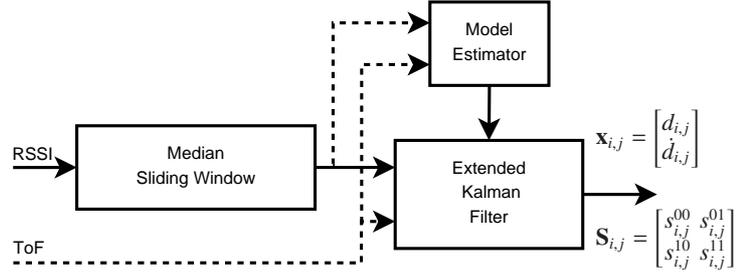


Fig. 1. RF-based ranging for each link: Dashed lines apply only when ToF data is available (adapted from [13]); In this figure, $\mathbf{x}_{i,j}$ is the estimated state (containing the distance and respective derivative) between robots i and j ; $\mathbf{S}_{i,j}$ is the respective covariance matrix

Taking advantage of this work, we build and share amongst the robots two matrices based on the output of the EKF: the matrix $\mathbf{D}_{n \times n} = d_{ij}$, Eq. (1), where d_{ij} is the distance estimate between robots i and j as estimated by robot i , and the matrix $\mathbf{V}_{n \times n} = s_{ij}^{00}$, Eq. (2), where s_{ij}^{00} is the variance of the distance estimate between robots i and j , as estimated by robot i .

$$\mathbf{D}_{n \times n} = \begin{bmatrix} 0 & d_{12} & d_{13} & \cdots & d_{1n} \\ d_{21} & 0 & d_{23} & \cdots & d_{2n} \\ d_{31} & d_{32} & 0 & \cdots & d_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & d_{n3} & \cdots & 0 \end{bmatrix} \quad (1)$$

$$\mathbf{V}_{n \times n} = \begin{bmatrix} 0 & s_{12}^{00} & s_{13}^{00} & \cdots & s_{1n}^{00} \\ s_{21}^{00} & 0 & s_{23}^{00} & \cdots & s_{2n}^{00} \\ s_{31}^{00} & s_{32}^{00} & 0 & \cdots & s_{3n}^{00} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{n1}^{00} & s_{n2}^{00} & s_{n3}^{00} & \cdots & 0 \end{bmatrix} \quad (2)$$

Finally, the algorithms we will use require these matrices to be symmetrical, which seldom happens due to different transmission power in different robots, non omnidirectional antennas, etc.. That being said, in order to create and feed a symmetric distance matrix to the those algorithms, we define matrix $\mathbf{G}_{n \times n}$ according to Eq. 3, and matrix $\mathbf{W}_{n \times n}$ according to Eq. 4.

$$\mathbf{G}(i, j) = \begin{cases} \mathbf{D}(i, j), & \mathbf{V}(i, j) < \mathbf{V}(j, i) \\ \mathbf{D}(j, i), & \text{otherwise} \end{cases} \quad (3)$$

$$\mathbf{W}(i, j) = \begin{cases} \mathbf{V}(i, j), & \mathbf{V}(i, j) < \mathbf{V}(j, i) \\ \mathbf{V}(j, i), & \text{otherwise} \end{cases} \quad (4)$$

3.2 Estimating positions from distances

MultiDimensional Scaling (MDS) [2] is relatively simple to implement technique that can be used to compute the robots relative positions. MDS is a technique used in multivariate analysis that transfers a known $n \times n$ matrix of dissimilarities to n points of an b -dimensional Euclidean space in such a way that the pairwise distances between points are compatible with the dissimilarities matrix. Consequently, by limiting b to two, i.e. 2-dimensional positions, and by using the positive semi-definite matrix $\mathcal{G}_{n \times n}$ containing the pairwise distances between all robots, we can write Eq. (5). Where the MDS function returns the 2-dimensional $\mathcal{M}_{n \times 2}$ containing the positions of the robots.

$$\mathcal{M} = \text{MDS}(\mathcal{G}) \quad (5)$$

With respect to defining relative positions for a team of mobile nodes, MDS already sorts out certain ambiguities that are inherent to the relative localisation process, e.g. eigenvector switching. However, that has the consequence that a small perturbation in the distances matrix can bring totally different results for the coordinates, such as map flips. Since the nodes position is only recovered up to rigid motion, orientation of the team cannot be determined just with pair-wise distances, neither can symmetry relationships. To obtain relative positions estimates that vary smoothly, we carry out the following adjustments of the coordinates provided by the MDS (considering only the result presented in 2D space, i.e. $b = 2$).

Let $\mathcal{M}_{n \times 2} = (\mathbf{m}_1; \mathbf{m}_2; \dots; \mathbf{m}_n)$ denote the coordinates determined with MDS, where $\mathbf{m}_i = (m_i^x, m_i^y)$ is the 2D position of robot i ; $\mathcal{L}_{n \times 2} = (\mathbf{l}_1; \mathbf{l}_2; \dots; \mathbf{l}_n)$ denote a set of arbitrary reference positions, where $\mathbf{l}_i = (l_i^x, l_i^y)$ is the 2D position of robot i ; and $\mathcal{F}_{n \times 2} = (\mathbf{f}_1; \mathbf{f}_2; \dots; \mathbf{f}_n)$ denote the final coordinates, where $\mathbf{f}_i = (f_i^x, f_i^y)$ is the 2D position of robot i .

We consider the robot making these calculations (herein referred by 1) as being in the origin. Then, because of the flip ambiguity, we generate $\mathcal{M}'_{n \times 2} = (\mathbf{m}'_1; \mathbf{m}'_2; \dots; \mathbf{m}'_n)$, Eq. (6), that represents the mirror image of the output of MDS along the y -axis.

$$\mathcal{M}' = \mathcal{M} \times \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \quad (6)$$

In order to remove the rotation ambiguity, for each robot, we calculate the angle that would be required to align it with the reference $\phi = \text{atan2}(\mathcal{L}) - \text{atan2}(\mathcal{M})$. Where atan2 represents the four quadrant arctangent. Similarly, we calculate $\phi' = \text{atan2}(\mathcal{L}) - \text{atan2}(\mathcal{M}')$. Using those two hypotheses we choose the best coordinate set according to Eq. (7). By selecting the set with the smallest standard deviation, we are selecting the topology with more similarities to the reference, i.e. the topology in which all robots require approximately the same rotation to match the reference. Finally, we remove the points whose residuals exceed one standard deviation (ϕ_e) from ϕ and ϕ' , i.e. if

$\text{abs}(\phi_e - \text{mean}(\phi)) > \text{std}(\phi)$ or if $\text{abs}(\phi_e - \text{mean}(\phi^I)) > \text{std}(\phi^I)$, and we calculate α , the clockwise rotation angle that minimises the square error of the angle between the estimate and the reference (Eq. (8)).

$$\mathcal{T} = \begin{cases} \mathcal{M}, & \text{std}(\phi) < \text{std}(\phi^I) \\ \mathcal{M}^I, & \text{otherwise} \end{cases} \quad (7)$$

$$\alpha = \begin{cases} \min [(\phi - \alpha) \cdot (\phi - \alpha)], & \text{std}(\phi) < \text{std}(\phi^I) \\ \min [(\phi^I - \alpha) \cdot (\phi^I - \alpha)], & \text{otherwise} \end{cases} \quad (8)$$

The last step to calculate the final coordinate \mathcal{F} , is rotating the selected topology as in Eq. (9). Note that ϕ or ϕ^I can contain angles similar in rotation but different in value, i.e., any values separated by 2π . Therefore, we analyse the residuals both between $[-\pi, \pi]$ and $[0, 2\pi]$, choosing the one with less standard deviation.

$$\mathcal{F} = \mathcal{T} \times \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \quad (9)$$

3.3 Kalman filter

In order to estimate the relative positions of a team of robots, we implemented a Kalman filter (KF) [23]. The state vector is given in Eq. (10), where $\mathcal{P}_{2n \times 1}$ is the state vector, and $\mathbf{p}_i = (p_i^x, p_i^y)$ is the estimated 2D position of robot i . The equations of the state space model are provided below.

$$\mathcal{P}_{2n \times 1} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \mathbf{p}_3 \ \dots \ \mathbf{p}_n]' \quad (10)$$

The prediction equation is Eq. (11), where $\omega(k) \sim \mathcal{N}(0, \mathbf{R})$ is the process noise at instant k . Finally, we measure the state \mathcal{P} directly by sampling the output of MDS, Eq. (12) where, $\overline{\mathcal{P}}(k)$ is the measurement and $v(k) \sim \mathcal{N}(0, \mathbf{Q})$ is the measurement noise.

$$\mathcal{P}(k) = \mathcal{P}(k-1) + \omega(k) \quad (11)$$

$$\overline{\mathcal{P}}(k) = \mathcal{P}(k) + v(k) \quad (12)$$

In order to estimate the process noise ω , we apply an heuristic inspired on the technique used in [14] to estimate relative velocities. However, instead of using the speed between robots, we use the estimated variances of the distances between robots, Eq. (14), allowing us to use the uncertainty of the distance measurements as a measurement of the state progression. In detail, we use the estimated variances of the distances between robots (\mathbf{W}), and compute a unit vector $\mathbf{u}_{i,j}$, Eq. (13), pointing from the position of the robot i to the position of robot j . Then, the state variance of robot i (\mathbf{r}_i) is calculated as the sum of the absolute value of those vectors projected in the x and y axes, where $\mathbf{r}_i = (r_i^x, r_i^y)$. Finally, we divide the velocity by n , to remove the multiple inclusions of the same variance, obtaining the diagonal matrix \mathbf{R} , Eq.(15).

$$\mathbf{u}_{i,j} = (\mathbf{u}_{i,j}^x, \mathbf{u}_{i,j}^y) = (\mathbf{p}_i(k-1) - \mathbf{p}_j(k-1)) / |\mathbf{p}_i(k-1) - \mathbf{p}_j(k-1)| \quad (13)$$

$$\mathbf{r}_i = \frac{1}{n} \times \sum_{j=1..n, j \neq i} \mathbf{W}(i, j) \cdot \left(\left| \mathbf{u}_{i,j}^x \right|, \left| \mathbf{u}_{i,j}^y \right| \right) \quad (14)$$

$$\mathbf{R}_{n \times n} = \begin{bmatrix} r_1^x & 0 & 0 & \cdots & 0 & 0 \\ 0 & r_1^y & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & r_n^x & 0 \\ 0 & 0 & 0 & \cdots & 0 & r_n^y \end{bmatrix} \quad (15)$$

3.4 Integrating measurements

In order to facilitate the understanding of the remaining of this paper, we will now assume the output of MDS to be in the localisation system state format of the state, i.e. $\begin{bmatrix} x_1 & y_1 & x_2 & y_2 & \cdots & x_n & y_n \end{bmatrix}' = \text{MDS}(\mathbf{G})$.

Then, in order to compute the process noise we use a Monte Carlo (MC) approach, according to which we add random noise to the distance inputs, and repeatedly calculate the output, thus sampling the localisation function. With a sufficient number of runs we can determine the impact of the noise on the output distribution. Therefore, we execute MDS with the symmetrical matrix \mathbf{G} obtaining the positions $\overline{\mathcal{P}}(k)$, and adjust the robots positions to minimise the error with relation to the current state estimate. Then, we execute the multidimensional scaling algorithm q times, Eq. (17), where q can be configured according to the precision required and the computational power available. For each of those executions we use as input $\mathbf{G} + a \times \mathbf{H}$, where $a \sim \mathcal{N}(0, 1)$, and \mathbf{H} is the matrix of standard deviations, obtained from the element-wise square root of \mathbf{W} . To obtain meaningful results after each execution, we adjust the robots positions to minimise the error with relation to $\overline{\mathcal{P}}(k)$. Finally we obtain the covariance matrix $\mathbf{Q} = \text{cov}(\mathbf{M}_z)$, where $z = [1..q]$.

$$\mathbf{H} = \begin{bmatrix} \sqrt{W_{11}} & \sqrt{W_{12}} & \sqrt{W_{13}} & \cdots & \sqrt{W_{1n}} \\ \sqrt{W_{21}} & \sqrt{W_{22}} & \sqrt{W_{23}} & \cdots & \sqrt{W_{2n}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sqrt{W_{n1}} & \sqrt{W_{n2}} & \sqrt{W_{n3}} & \cdots & \sqrt{W_{nn}} \end{bmatrix} \quad (16)$$

$$\mathbf{M}_z = \text{MDS}(\mathbf{G} + a \times \mathbf{H}) \quad (17)$$

4 Simulation results

In this section, we first describe the simulation setup, Fig. 2, namely how we generate measurements and how we share the distance estimates between the robots. Then we present the results we obtained using our simulator on ground-truth (GT) collected from the CyberRescue@RTSS2009 competition [7].

4.1 Generating RSSI and ToF measurements

In order to realistically simulate our proposal, we modelled the sensors measurements from the real experiments performed in [13]. Namely, we generate RSSI values taking into account the hardware 2dBm resolution, Eq. (18), where $\rho(d)$ is the medium propagation using the model in Eq. (19). For the RSSI the parameters are: $\sigma_\rho^2 = 20$, $\rho_0 = -39.6955$, $\alpha = 1.1558$, and $a_\rho \sim \mathcal{N}(0, 1)$. In addition, we also generate ToF measurements from real distance according to Eq. (20). In this case $a_d \sim \mathcal{N}(0.3842, 1)$ and $\sigma_d^2 = 0.4$.

$$\bar{\rho} = -2 \times \text{round}\left(0 \leq -\left(\rho(d) + a_\rho \times \sqrt{\sigma_\rho^2} + 35\right)/2 \leq 31\right) - 35 \quad (18)$$

$$\rho(d) = \rho_0 - 10\alpha \log_{10}(d) \quad (19)$$

$$\bar{d}(d) = d + a_d \times \sqrt{\sigma_d^2} \quad (20)$$

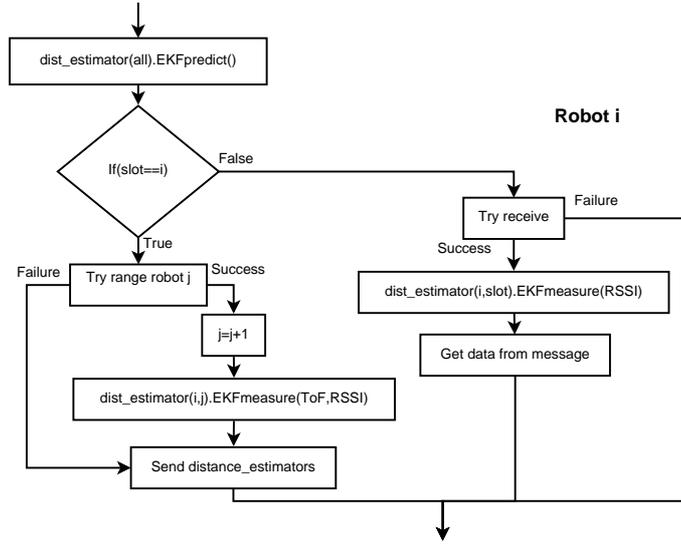


Fig. 2. Simulation loop for robot i – Predict all Kalman filters; If robot is a receiver, try to receive from sender and measure RSSI; If robot is the sender, try to range robot j and measure ToF and RSSI, finally estimate positions

4.2 Simulating the communications

In order to simulate the delays of the information reaching a robot, we have devised a simple communication protocol. A TDMA schedule with slot size of 50ms was created

such that each robot i transmits in slot i , therefore, at each slot the simulator will run the loop in Fig.2 for each robot. A round robin ranging schedule was implemented such that every i^{th} slot, robot i tries to range one of the other robots, then, in the next i^{th} slot it will try to range the next robot, and so on. Still in the same slot, robot i will broadcast a message containing its distance estimates and distance variance, which the other robots will try to receive. The communications were programmed with a probability of success of 94% from measurements carried out by [12] in real conditions similar to our simulation, i.e. TDMA rounds with no external interference.

4.3 Results

In order to perform experimental evaluation of our proposal, we used logs collected in a simulated robots competition (CyberRescue@RTSS2009 [7]) using the Cyber-Physical Systems Simulator (CPSS) [18]. We used the GT positions collected in the 9 logs available on the website, as the path the robots travel through. Each log is composed of five robots moving in a 28 m by 14 m arena. For the purpose of this work, the walls were not considered for non-line of sight and reflection effects.

We ran the robots through our simulator, using the paths obtained from the logs to generate simulated measurements. The simulator output was used to run our positions estimator from the perspective of robot 1, and we set to 10 the number of MDS executions required for the measurement of the topology in the Monte Carlo approach. To be able to compare our generated positions to the log GT, we applied the same transformation techniques to the GT data using the positions estimate as the reference, placing them in the same reference frame as our data. Therefore, from this point on, when we mention GT, we are referring to the transformed GT.

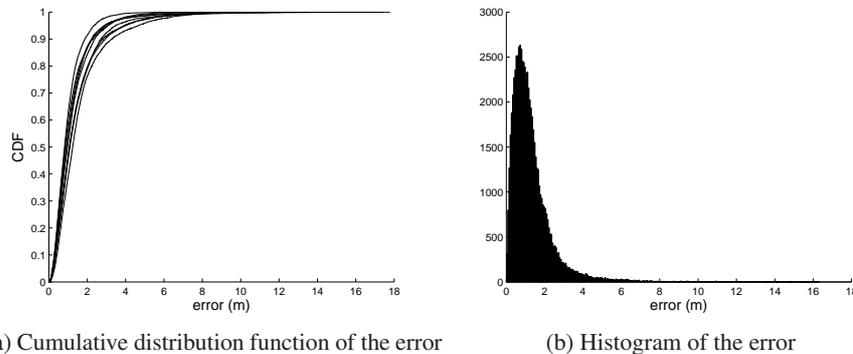


Fig. 3. Error of the position estimates – 90% of the errors are under 5 m for all cases and below 2 m in the best case, with an overall average of 1.3 m

In order to measure the accuracy of our proposal we calculated the error of the estimate, as the difference between the estimated position and the GT position. An

histogram of such error is presented in Fig. 3a, where we can see that under realistic simulated measurements, 90% of the errors are under 5 m for all the cases and under 2 m for the best case, with an overall average at 1.3 m

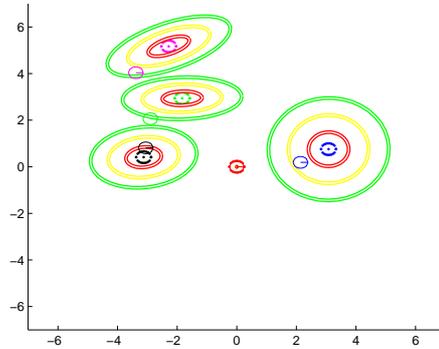


Fig. 4. A zoomed in snapshot of the simulation field – full line robots represent the ground-truth, dotted robots represent the estimations surrounded by the 1-standard deviation ellipse(red), by the 2-standard deviation ellipse(yellow), by the 3-standard deviation ellipse(green)

Nevertheless, the information we were more interested in is the confidence measurement. To analyse the confidence we calculate the percentage of GT positions contained inside different scales of the covariance ellipse. Figure 4 depicts a snapshot of one of the simulations where we can see the top left robot inside of the 3-standard deviation ellipse, the center left robot just on the 3-standard deviation ellipse, the bottom left robot inside of the 1-standard deviation ellipse, and the right robot inside of the 2-standard deviation ellipse. The total percentage of GT inside different scales of the covariance ellipse, from 1 to 10 times the standard deviation, is presented in Fig. 5, for all 9 simulations. We can see that for 3 times the standard deviation, over 80% of the GT fall inside the covariance ellipse. For 5 times, this value raises to about 95%.

5 Conclusions and future work

In this work we presented a novel technique that uses RF-based range estimates, only, to track the relative positions of a fully mobile team of robots providing reliability information as the covariance of the positions. We believe that this is one of the first works to propose confidence values to position estimates obtained through MDS, particularly for relative localization purposes without any anchors. Using RF-only information, our method could determine the relative localization of a team of 5 freely moving robots with an average error of 1.3m in a region of 28×14 m. Results show that our approach can consistently provide similar performance across different experiments.

Currently we are implementing this approach on a real robots team, particularly a RoboCup MSL team, with which we plan to validate our proposal in a real scenario.

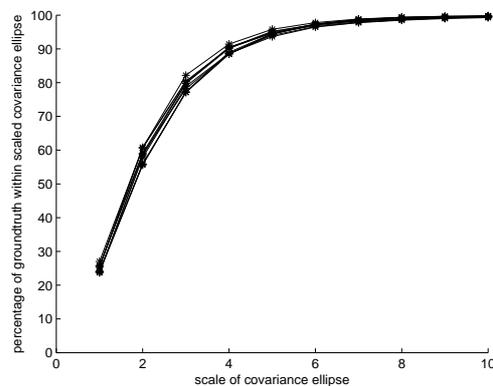


Fig. 5. Percentage of GT positions contained inside of the covariance ellipses at different scales, from 1 to 10 times the standard deviation – For 3 times the standard deviation, over 80% of the times, the GT falls inside the covariance ellipse

Acknowledgements

This work was partially supported by the Portuguese Government through FCT (Fundação para a Ciência e a Tecnologia) grant SFRH/BD/74292/2010 and by the Portuguese and Brazilian Governments through the CAPES-FCT bilateral project PICC. The authors would also like to acknowledge the valuable comments provided by Prof. Traian Abrudan from Oxford University.

References

1. Biswas, J., Veloso, M.: Depth camera based indoor mobile robot localization and navigation. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on. pp. 1697–1702 (May 2012)
2. Borg, I., Groenen, P.J.F.: Modern Multidimensional Scaling: Theory and Applications (Springer Series in Statistics). Springer, 2nd edn. (Aug 2005)
3. Burgard, W., Moors, M., Fox, D., Simmons, R., Thrun, S.: Collaborative multi-robot exploration. In: Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on. vol. 1, pp. 476–481. IEEE (2000)
4. Chen, Z.X., Wei, H.W., Wan, Q., Ye, S.F., Yang, W.L.: A supplement to multidimensional scaling framework for mobile location: A unified view. Signal Processing, IEEE Transactions on 57(5), 2030–2034 (may 2009)
5. Destino, G., De Abreu, G.: Weighing strategy for network localization under scarce ranging information. Wireless Communications, IEEE Transactions on 8(7), 3668–3678 (jul 2009)
6. Djughash, J., Singh, S., Kantor, G., Zhang, W.: Range-only slam for robots operating cooperatively with sensor networks. In: Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on. pp. 2078–2084 (May 2006)
7. Facchinetti, T., Vedova, M.D.: Cyberrescue@rtss2009 (2009), <http://robot.unipv.it/cyberrescue-RTSS09>

8. Ferris, B., Fox, D., Lawrence, N.D.: WiFi-SLAM using gaussian process latent variable models. In: IJCAI. vol. 7, pp. 2480–2485 (2007)
9. Jamaa, M.B., Koubaa, A., Kayani, Y.: Easyloc: RSS-based localization made easy. *Procedia Computer Science* 10(0), 1127 – 1133 (2012)
10. Li, H., Almeida, L., Wang, Z., Sun, Y.: Relative positions within small teams of mobile units. In: *Mobile Ad-Hoc and Sensor Networks. Lecture Notes in Computer Science*, vol. 4864, pp. 657–671. Springer Berlin / Heidelberg (2007)
11. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B., et al.: FastSLAM: A factored solution to the simultaneous localization and mapping problem. In: *AAAI/IAAI*. pp. 593–598 (2002)
12. Oliveira, L., Almeida, L., Santos, F.: A loose synchronisation protocol for managing RF ranging in mobile ad-hoc networks. In: *RoboCup 2011: Robot Soccer World Cup XV, Lecture Notes in Computer Science*, vol. 7416, pp. 574–585. Springer Berlin Heidelberg (2012)
13. Oliveira, L., Di Franco, C., Abrudan, T.E., Almeida, L.: Fusing time-of-flight and received signal strength for adaptive radio-frequency ranging. In: *Advanced Robotics (ICAR), 2013 16th International Conference on*. pp. 1–6. IEEE (2013)
14. Oliveira, L., Li, H., Almeida, L., Abrudan, T.E.: RSSI-based relative localisation for mobile robots. *Ad Hoc Networks* 13, Part B(0), 321 – 335 (2014)
15. Olufs, S., Vincze, M.: Embedded vision-based Monte-Carlo robot localisation without additional sensors. In: *AFRICON, 2013*. pp. 1–6 (Sept 2013)
16. Pan, T.W., Hou, T.C.: Localization of moving nodes in an anchor-less wireless sensor network. In: *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*. pp. 3112–3116 (April 2012)
17. Patwari, N., Hero, A.O., I., Perkins, M., Correal, N., O’Dea, R.: Relative location estimation in wireless sensor networks. *Signal Processing, IEEE Transactions on* 51(8), 2137 – 2148 (aug 2003)
18. Pereira, A., Nuno Lau, T.F.: Cyber-physical systems simulator (2014), <http://sourceforge.net/projects/cps/>
19. Pinto, M., Paulo Moreira, A., Matos, A., Sobreira, H., Santos, F.: Fast 3d map matching localisation algorithm. *Journal of Automation and Control Engineering*, June pp. 110–114 (2013)
20. Pinto, M., Sobreira, H., Moreira, A.P., Mendona, H., Matos, A.: Self-localisation of indoor mobile robots using multi-hypotheses and a matching algorithm. *Mechatronics* 23(6), 727 – 737 (2013)
21. Schneider, F.E., Wildermuth, D.: Influences of the robot group size on cooperative multi-robot localisation analysis and experimental validation. *Robotics and Autonomous Systems* 60(11), 1421 – 1428 (2012), towards *Autonomous Robotic Systems 2011*
22. Shioda, S., Shimamura, K.: Cooperative localization revisited: Error bound, scaling, and convergence. In: *Proceedings of the 16th ACM International Conference on Modeling, Analysis Simulation of Wireless and Mobile Systems*. pp. 197–206. MSWiM ’13, ACM, New York, NY, USA (2013)
23. Thrun, S., Burgard, W., Fox, D.: *Probabilistic robotics*. MIT press (2005)
24. Xiao, Z., Wen, H., Markham, A., Trigoni, N.: Lightweight map matching for indoor localisation using conditional random fields. In: *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*. pp. 131–142. IPSN ’14, IEEE Press, Piscataway, NJ, USA (2014)
25. Yin, F., Li, A., Zoubir, A., Fritsche, C., Gustafsson, F.: RSS-based sensor network localization in contaminated gaussian measurement noise. In: *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013 IEEE 5th International Workshop on*. pp. 121–124 (Dec 2013)