Robot Localization System in a Hard Outdoor Environment

Teresa Conceição[1], Filipe Neves dos Santos[1], Paulo Costa[1], António Paulo Moreira[1]

INESC TEC - INESC Technology and Science (formerly INESC Porto) and Faculty of Engineering, University of Porto, Porto, Portugal,
ee12082@fe.up.pt,fbsantos@inesctec.pt,paco@fe.up.pt,amoreira@fe.up.pt

Abstract. Localization and mapping of autonomous robots in a hard and unstable environment (Steep Slope Vineyards) is a challenging research topic. Typically, the commonly used dead reckoning systems can fail due to the harsh conditions of the terrain and the Global Position System (GPS) accuracy can be considerably noisy or not always available. One solution is to use wireless sensors in a network as landmarks. This paper evaluates a ultra-wideband time-of-flight based technology (Pozyx), which can be used as cost-effective solution for application in agricultural robots that works in harsh environment. Moreover, this paper implements a Localization Extended Kalman Filter (EKF) that fuses odometry with the Pozyx Range measurements to increase the default Pozyx Algorithm accuracy.

1 Introduction

Localization and mapping is one of the key features of a truly autonomous robot application, which must be aware of its environment in order to locate itself and navigate through its surroundings. Although there are several accurate technologies for this purpose on regular stable conditions, the problem remains a complex challenge when facing variant atmospheric conditions or complex environments. Areas with dense vegetation for instance can be affected by signal blockage and multi-path interference reducing the efficiency of the widely used Global Position System (GPS) and not even the usually reliable dead reckoning systems can give an accurate localization since they also suffer from harsh terrain shape.

A range-based Localization system is then proposed to deal with this conditions. In our previous work [[7]] a redundant localization solution for mountain vineyards based on the identification of natural features by Laser Range Finder (LRF) measurements is presented which can cope with the lack of access to Global Navigation Satellite System (GNSS). Despite being a slight improvement, natural feature detection is still highly dependent on environmental conditions and was also proved to be less reliable on the row transitions of the vineyards since it has no vine masts or trunks (the identified natural landmarks) on sight. Additionally, in [[2]], an artificial landmarking mapping procedure is proposed, called Beacons Mapping Procedure (BMC). BMC can map automatically each

Radio Frequency Identification (RFID) tags that can be placed in the begin/end of each vineyard row. These artificial beacons are an input for our hybrid SLAM approach (VineSLAM) [[7]] and a redundant localization information for the agricultural robots.

In this paper, Pozyx, a low-cost time-of-flight technology with ultra-wideband (UWB) based communication is deeply characterized as a complementary alternative tool to the previous works. A further implementation based on the fusion of the sensor's range measurements with dead reckoning by the Extended Kalman Filter (EKF) algorithm is proposed and compared against its own localization algorithm. The obtained results are presented and discussed and allow to present formulations for better results of accurate and reliable localization systems.

2 Background

Nowadays, the main approaches to the Robot Localization problem include:

- **GPS (Global Position System)**: gives a precise localization and it's the most commonly used since the service is globally available, precise and not too expensive. It is however limited to an outdoor environment without any obstruction in the line of sight and a good satellite coverage so in some cases it is not a viable option.
- **Dead Reckoning**: uses Odometry and the Inertial Measurement Unit (IMU) to integrate the position of the robot from encoders and/or inertial sensors such as accelerometers and gyroscopes. Its error is cumulative therefore it can lead to divergency after some time.
- **Range-Based**: measure angle and/or distances to natural or artificial landmarks. It can calculate the position with different technologies such as Radio Frequency (RF) beacons, Radio Frequency Identification (RFID) tags or Laser Rangefinder (LRF) sensors.
- **Visual Odometery**: localize the robot by sensing its surroundings with a single camera or stereo vision system. It's not so commonly used as the other approaches but the growing popularity of evolutionary and learning algorithms have given this technique a lot of research attention lately.

Although these technologies can be used independently, proper fusion of them in a multi-sensor environment give greater accuracy and more robust systems. They are usually filtered and combined with algorithms highly developed for non-linear systems such as Particle Filters, the Extended Kalman Filter(EKF) or the Unscented Kalman Filter (UKF) [[9]]. A Particle Filter is more flexible and appropriate when the noise doesn't follow a Gaussian distribution, which isn't an issue for this specific case. The EKF was then chosen because it's less complex and more computationally efficient.

2.1 Range-Based Localization

Range-based Localization is widely used in Wireless Sensors Network Localization. It typically uses reference points (anchors) and its distances to one or more Nodes of Interest (NOI) in order to estimate the required positions. These reference points are usually static either mapped a priori or with an unknown position which is dynamically determined. The distances and angles between the nodes and landmarks in the network can be measured with different technologies such as Radio Frequency (RF), Laser or UWB. The methods used to perform the calculations are preferably Time Of Flight (TOF), Time Difference of Arrival (TDOA), Angle of Arrival (AOA) and Received Signal Strength Indicator (RSSI).

2.2 UWB based Localization by Pozyx

The Ultra-Wideband is a wireless radio technology mostly used in communications which has been lately receiving more attention in positioning applications. In [[3]] González combines UWB range measurements with the vehicle odometry with a Particle Filter approach in an indoor environment. Further characterization for both indoor and outdoor is done in [[4]]. Due to its increased bandwidth it not only avoids interference with other types of RF signals but also deals with obstacles and walls since the signal can go through them. Therefore, it is seen as an emergent ranging technology being robust to multipath and Non-Line-of-sight situations. Apart from that, each tag of a UWB system is uniquely identified and aware of every other tag so the data association problem is automatically solved. Its principle is based essentially on sending radio signals, which travel at the speed of light, from a mobile transceiver (tag) to a set of known anchors, measuring the Time of Flight (ToF) and consequently the distances to this set of points. For this work, we selected Pozyx System [[5]], which is a hardware solution for accurate position and motion sensing based UWB system. One can set its UWB configuration settings according to four different parameters which can impact the whole performance of the system:

- **Channel**: The communication between devices can be made in 6 independent UWB channels. Generally, lower channel numbers have lower frequencies which increase the communication range.
- **Bitrate**: The bitrate choice is either 110kbit/sec, 850kbit/sec or 6.81Mbit/sec. A higher value will result in faster communication but reduced range sensitivity.
- **Pulse Repetition Frequency (PRF)**: the increase or decrease of the rates have the same impact on the system network as the bitrate setting although in a much lower scale.
- **Preamble Length**: This setting has 8 different options: 4096, 2048, 1536, 1024, 512 , 256 , 128 , or 64 symbol. Smaller values and therefore shorter messages come with faster communication but reduced operating range.

Two localization approaches will be taken into consideration: one using only the raw measurements of the sensor filtered with an EKF algorithm; the other based on the Pozyx Localization feature given by its firmware. The later can use different dimensions (2D, 2.5D, 3D) and different algorithms.

3 Problem Formulation

Generally the classical Robot Range-Based Localization problem can be defined as estimating the robot pose $X = \begin{bmatrix} x \; y \; \theta \end{bmatrix}^T$ given a set of distances measurements $Z_{Bi}(k) = \{r_{Bi}, i \in \{1...N_B\}$ to a given number of beacons N_B mapped in $M_B = [x_{Bi} y_{Bi}], i \in \{1...N_B\}$ both X and M_B defined in a global reference frame $GxGy$. In this particular case the following assumptions are made:

- The beacons position in M_B is static and known a prior;
- At time step k_t the Odometry values are available for $k \in \{k_{t-1}...k_t\}$. This will be used to provide the input $u(k) = \begin{bmatrix} \Delta x_{odom} \; \Delta y_{odom} \; \Delta\theta_{odom} \end{bmatrix}^T$ to the system in which each $u(k)$ refers to the variation of the odometry between times k and $k-1$ relative to the robot reference frame $RxRy$;
- Due to the uniquely identification of every node in the UWB system,the data association between each beacon i and the measurement $\{r_{Bi}\}$ is clearly defined at all moments.

3.1 Pozyx Range-Only + EKF

The raw range measurements of the anchors themselves can't return any robot location, hence a localization algorithm has to be chosen. The trilateration solution yields good results theoretically but practically it generates a region solution because of the associated measurement noise. Furthermore, it requires at least 3 anchors to outcome a position and although we're working with 4, they are not always available (due to timeout in the communication). Thus, the well-known EKF Localization was chosen as a good approach to the problem given the conditions.

3.2 Extended Kalman Filter (EKF) Localization

The Extended Kalman Filter is the version of the Kalman Filter that overcomes the non-linearity of some system dynamics. It proceeds by continually updating a linearization around the previous state estimation and by approximating the state densities by Gaussian densities. The EKF algorithm (1) used for the Localization is based on "EKF Localization with Known Correspondences" presented in [9]. It can fuse an Odometry model with the Sensor measurements going through essentially two steps: the **Prediction** phase which estimates the actual pose by the defined State Transition Model and the **Correction** phase which corrects the predicted state inputting the Range Measurements in the Observation Model.

State Transition Model An Odometry Model based on the one introduced by [**eliazar˙learning˙2004**] was chosen ((2))also considering the fact that the platform we'll be working on, AGROB, is a unidirectional traction robot.

$$X_{k+1} = f(X_k, u_{k+1}) + N(0, Q) \tag{1}$$

$$f(X_k, u_{k+1}) = X_k + \begin{bmatrix} \mathit{\Delta} x_{Odom}(k+1) cos(\theta + \frac{\mathit{\Delta}\theta_{odom}}{2}) \\ \mathit{\Delta} x_{Odom}(k+1) sin(\theta + \frac{\mathit{\Delta}\theta_{odom}}{2}) \\ \mathit{\Delta}\theta_{odom} \end{bmatrix} \tag{2}$$

Apart from the transition function f, the model noise N is represented by a Gaussian distribution with zero mean and Q covariance, assumed constant. In order to linearize the process, EKF uses the Taylor expansion which inputs the uncertainty by computing both the Jacobian of f with respect to X_k (Gx_k 3) and to u_k (Gu_k 4).

$$Gx_k = \frac{\partial f}{\partial X} = \begin{bmatrix} 1 \; 0 & -\mathit{\Delta} x_{Odom}(k) * sin(\theta(k) + \frac{\mathit{\Delta}\theta_{Odom}(k)}{2}) \\ 0 \; 1 & \mathit{\Delta} x_{Odom}(k) * cos(\theta(k) + \frac{\mathit{\Delta}\theta_{Odom}(k)}{2}) \\ 0 \; 0 & 1 \end{bmatrix} \tag{3}$$

$$Gu_k = \frac{\partial f}{\partial u} = \begin{bmatrix} cos(\theta(k) + \frac{\mathit{\Delta}\theta_{Odom}(k)}{2}) & 0 & \frac{-\mathit{\Delta} x_{Odom}(k) * sin(\theta(k) + \frac{-\mathit{\Delta}\theta_{Odom}(k)}{2})}{2} \\ sin(\theta(k) + \frac{\mathit{\Delta}\theta_{Odom}(k)}{2}) & 0 & \frac{\mathit{\Delta} x_{Odom}(k) * cos(\theta(k) + \frac{\mathit{\Delta}\theta_{Odom}(k)}{2})}{2} \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

Observation Model In what concerns the sensor measurements, the expected ranges are determined by the Euclidean Distance between the predicted position and the known anchor localization (5). Once again the measurements are affected by an additive Gaussian noise with zero mean and covariance constant $R = [\sigma_r^2]$, a parameter related to the sensor error characterization.

$$\widehat{Z_{Bi}} = h(\widehat{X_k}, M_{Bi}) + N(0, R) \tag{5}$$

$$h(\widehat{X_k}, M_{Bi}) = sqrt((x_{Bi} - \widehat{x_k})^2 + (y_{Bi} - \widehat{y_k})^2) \tag{6}$$

As well as for the state transition model, EKF algorithm also needs the Jacobian of h related to X_k (Hz_{ki} 7).

$$Hz_{ki} = \begin{bmatrix} -\frac{x_{Bi} - \widehat{x_k}}{sqrt((x_{Bi} - \widehat{x_k})^2 + (y_{Bi} - \widehat{y_k})^2)} \\ -\frac{y_{Bi} - \widehat{y_k}}{sqrt((x_{Bi} - \widehat{x_k})^2 + (y_{Bi} - \widehat{y_k})^2)} \\ 0 \end{bmatrix} \tag{7}$$

Detection of Outliers Generally, a network of sensors is always prone to noisy measurements. For a proper use of the observations of Pozyx Ranges, the process needs to be able to detect and to cope with this unreliable data whose characteristics deviate from the normal pattern. This phase is called Outlier Detection. There are several ways to approach the problem. In presence of a multivariate data, one of the common methods is using the Mahalanobis' distance (MD) (8) as a statistical measure of the probability of some observation belonging to a certain data set. This method as used in [[10] [8]], consists of computing the normalized distance between one point and all the population. One could simply use the Euclidean Distance but MD takes into account not only the distance to

Algorithm 1 EKF Localization Algorithm with Known Measurement correspondences

```
    Input: $X_{k-1}, u_k, P_{k-1}, R, Q, ZB_k$
    Output: $X_k, P_k$

    PREDICTION
 1: $\widehat{X_k} = f(X_{k-1}, u_k)$
 2: $Gx_k = \frac{\partial f}{\partial X}$
 3: $Gu_k = \frac{\partial f}{\partial u}$
 4: $P_k = Gx_k P_{k-1} Gx_k^T + Gu_k Q Gu_k^T$

    CORRECTION
 5: for $i \leftarrow 1$ to $N_B$ do
 6:     $\widehat{Z_{ki}} = h(\widehat{X_k}, M_{Bi}) = \sqrt{(x_{Bi} - \widehat{x_k})^2) + (y_{Bi} - \widehat{y_k})}$
 7:     $Hz_{ki} = \frac{\partial h}{\partial X}$
 8:     $S_{ki} = [Hz_{ki} \widehat{P_k} Hz_{ki}^T + R]$
 9:     if ! $Outlier\_Detected$ then
10:         $K_i = \widehat{P_k} Hz_{ki}^T S^{-1}$
11:         $\widehat{X_k} = \widehat{X_k} + K_i[Z_{ki} - \widehat{Z_{ki}}]$
12:         $\widehat{P_k} = [I - K_i Hz_{ki}]\widehat{P_k}$

13: $X_k = \widehat{X_k}$
14: $P_k = \widehat{P_k}$
```

the mean but also the direction, i.e covariance. As a matter of fact, MD is equal to the Euclidean distance only when the covariance matrix is the identity (all the variables are independent and with the same variance).

$$Md(v) = \sqrt{(v - \mu)^T S^{-1} (v - \mu)} \tag{8}$$

Since there is only one sensor measurement at a time we will assume that the current estimated range calculated by the state estimation at the prediction step is the "statistical population". In order words, $\widehat{Z_k}$ will be the mean μ of the population and S will be a factor representing a combination of the uncertainty of the state and the actual sensor measurement (line 8 in algorithm 1).

Algorithm 2 Outlier Detection for EKF

```
Input: $S_{ki}, \widehat{Z_{ki}}, Z_{ki}$
Output: $isOutlier$
$Md(Z_{ki}) = \sqrt{(Z_{ki} - \widehat{Z_{ki}})^T S_{ki}^{-1} (Z_{ki} - \widehat{Z_{ki}})}$
if $Md(Z_{ki}) > threshold$ then
    return true
else
    return false
```

Choosing the threshold can be done by simply analyzing the data or by determining some probabilistic statistical parameter. Here the threshold was set accordingly to the χ^2 probability table so that the observations with less than 2.5% of probability were cut out.

4 Methodology

The work methodology consisted of a previous Range Error Characterization Test to define the Pozyx Range behavior that would be helpful for further implementing other algorithms. Then the two localization approaches were tested simultaneously in the same outdoor environment as the previous test. Finally, all the data acquired was post-processed with MATLAB as well as the testing and tuning of the EKF algorithm.

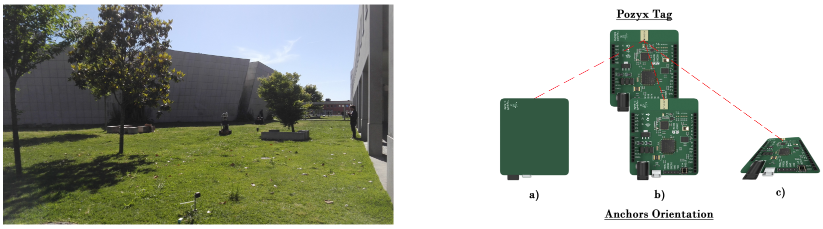


Fig. 1: Tests Environment on the left. Different antenna orientation for Range Test on the Right: a - Vertical Front, b- Vertical Back, c- Horizontal

For all the tests it was used the Pozyx Library for Arduino [6] combined with a ROS publisher node. The transmission of data was done by serial port using the Termios library [1] the Arduino to the ROS node which after validation published the information received with a rate of 10Hz in a native ROS Pose message [2] and a message created for publishing an array of ranges. A ROS message filter was also used to synchronize all the acquired data.

4.1 Range Error Characterization Test

To get a full characterization of the Pozyx system and its error a first experimental analysis was made by measuring the ranges from a static anchor to the moving sensor in increasingly discrete distances (1-10,15,20,25,30,40,50,60 meters). The Range test was made with both sensor and anchor about 20 cm from the floor and with 3 different antenna orientations (Fig.1). To yield better results in the future the antenna and the tag should be positioned higher so it can prevent some unwanted reflections/multipaths from the floor.

[1] Library to deal with I/O interfaces `http://pubs.opengroup.org/onlinepubs/7908799/xsh/termios.h.html`

[2] ROS PoseStamped message: `http://docs.ros.org/api/geometry_msgs/html/msg/PoseStamped.html`

4.2 Localization Tests

The Localization System was tested with AGROB v.16, a cost effective outdoor robot for application on steep slope vineyard monitoring tasks [[1]] The previous environment was used again, spreading this time all the 4 Pozyx anchors in the corners of a square with 20 meters of size. The tested trajectory consisted in a squared shape going through all the corners and then into the inside (results in Fig. 3. To analyze the results a Laser Scan (computed by the Hector Slam ROS package) was considered as the ground truth since the GPS data acquired was not reliable and precise.

4.3 Pozyx Configuration

The Pozyx System needs to be configured before the tests. As previously explained the UWB configuration settings can be set according to four different parameters which may impact the whole performance of the system. In this case the following configuration was used: **Channel 2** (Center frequency 3993.6MHz and bandwidth 499.2 MHz) - the channel with the better antenna gain as informed by the manufacturers; **Bitrate** 110kBit/s - in this first pre-eliminary study we were interested in having a good range performance and the rate of transmission was not so important since most of the algorithm run offline; **PRF**: 64MHz - the default setting was used because its value is not said to influence the performance drastically. **Preamble length**: 1024 symbols - the default was also chose in order to have a mid-term value. Regarding the Positioning system, this preliminary work on Pozyx will only analyze 2D dimensions with the **UWB-only algorithm** especially used in Non-Line of Sight measurements and that does not require continuous positioning accuracy.

5 Experimental Results and Discussion

5.1 Pozyx Error Characterization

When characterizing the performance of a system it is important to be aware that an error can have different sources and can be of different types. We'll focus on both accuracy and precision analysis, which are usually linked to systematic and random errors respectively. Although both are important, in this particular test, characterizing the precision of the sensor is of particular importance since it will be used to tune the EKF. Hence, more than the actual accuracy values we'll be looking for patterns in the standard deviations.

As we can deduce from Table 1 and Fig. 2 the error characterization highly depends on the antenna orientation and the distance to the sensor. Histogram (c) of Fig.2 shows that the error follows approximately a normal distribution under 30m when in its best orientation (Vertical Up). On the other hand, when the sensor's antenna is facing back the tag (Histogram (d) of Fig.2) there are more outliers so the error distribution doesn't seem so perfect. The same applies for distances higher than 30m in which the error metrics tend to be worse.

Table 1: Refers to the performance results for different antenna orientations. Under 10 meters the results don't follow any specific pattern and are always around the same values so only their average is presented.

Distance	Vertical Front			Vertical Back			Horizontal Up		
(m)	Error (mm)	Std (mm)	Timeouts (%)	Error (mm)	Std (mm)	Timeouts (%)	Error (mm)	Std (mm)	Timeouts (%)
<= 10	77.56	33.15	0.56	180.13	116.89	2.01	183.31	82.93	0.67
15	62.603	36.12	7.093	425.92	1060.23	30.08	196.90	318.79	60.91
20	147.05	37.64	18.20	1204.65	2105.25	51.90	2794.25	5245.79	98.86
25	192.33	35.31	25.36	90.51	78.54	40.45	157.11	179.69	94.61
30	164.00	43.06	5.371	624.21	1482.76	0.57	920.40	1463.53	71.40
40	1008.17	918.99	14.73	4905.02	732.32	38.67			
50	119.6	49.49	10.06						
60	2432.03	1094.83	6.49						

Regarding the Pozyx orientation, it is known that an antenna has not the same gain/distance models for different orientations so it comes as no surprise that error values are slightly different too. The performance of the antenna when facing the tag in a vertical front position is clearly better than for the other two orientations whose timeout communication errors are way higher and data received after 30/40 m is close to zero.

Through the navigation, the sensor embedded to the robot will be having different orientations towards the anchors. Thus, in future work, it will be interesting to do a more intensive study of this differences and to include a bearing measurement in the sensor observation model for the EKF. For now, we'll be focusing on the results for the vertical front orientation to define the error variables of the filter.

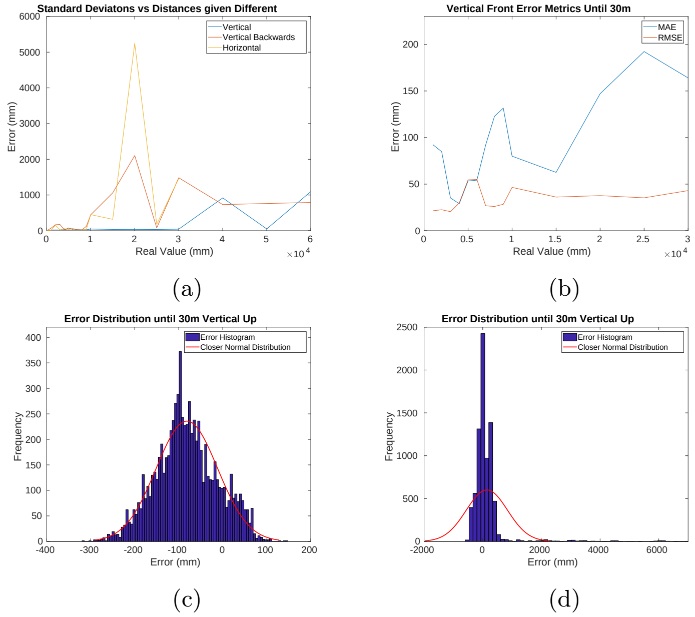
5.2 EKF Tuning

An important step in the post-processing step was the EKF tuning. Changing its parameters affects the convergence of the algorithm as well as the smoothness of the resulting trajectory estimation. Considering the previous noise characterization, R and Q were adjusted so that the estimation results presented weren't neither to noise sensitive neither too slow.

As it can be seen from (d) in Fig. 3, ranging measurement doesn't result in many outliers (except for the timeout ones which are instantly observed and don't need further calculations to be detected). Therefore, the effect of including the Outlier Detection Filter is not a big improvement simply because there are not many outliers to get rid of. To further test the effectiveness of the filter more tests with wider areas should be done.

5.3 Pozyx Positioning vs Pozyx Range + EKF

Fig.3 shows that both the procedures yield reasonable results, especially when compared to the Odometry localization that is clearly affected by some wheel



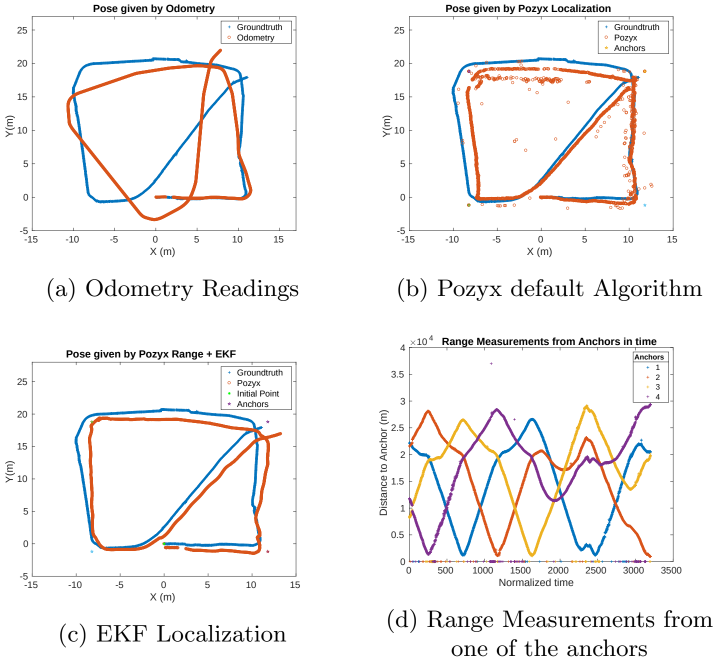
(a) (b)

(c) (d)

Fig. 2: Error Statistics: (a) Standard Deviations for every orientation; (b) Error and standard deviations of Vertical Front orientation under 30 m; (c) and (d) Error Distribution Under 30m for Vertical Front and Vertical Back orientation respectively

slippage in the curves. However, the native Pozyx Localization has a lot more outliers than the EKF Localization.

Lastly, Fig.4 give some notes on the right positioning of the anchors relative to the sensor. On the Pozyx Positioning algorithm there are a lot more outliers than in the different trajectory test of Fig.3 not only because of the bigger distance to some of the anchors (higher error and more timeouts) but also due to its position relatively to the sensor. Since it only uses the UWB ranges in a Trilateration process, if some of the references are in a straight line in the direction of the sensor then there will be an amplification of the error just like in the GPS principle of poor Geometric Dilution of Precision(GDOP). Thus, one can conclude that the anchors should be spread out in different directions around the robot's trajectory area so it can cover different directions. Contrarily to the Pozyx algorithm, the EKF Localization is not based on this method so it doesn't seem to be so affected by the poor GDOP.



(a) Odometry Readings

(b) Pozyx default Algorithm

(c) EKF Localization

(d) Range Measurements from one of the anchors

Fig. 3: After the data acquisition the Pozyx Ranges were inputted in the EKF and compared to the Pozyx Positioning Algorithm and to the considered Laser Scan ground truth.

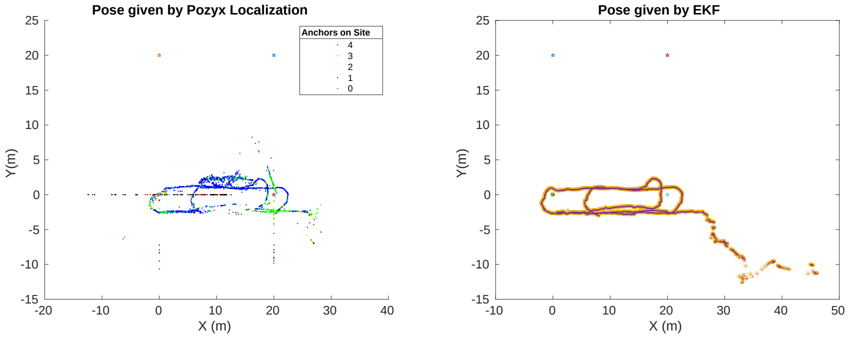


Fig. 4: Pozyx Algorithm vs EKF Localization on poor GDOP.

6 Conclusions and Future Work

The proposed EKF implementation presented much better results than the UWB-only native Pozyx Algorithm leading to less noisy smoother trajectories, however one needs to carefully tune it to assure the right convergence. This tunning should also consider not only the system characterization but also the environment. The Pozyx Algorithm, in turn, although with undesired outliers, can still be a good alternative to the standalone Odometry whose errors are cumulative and prone to fail at curves. Furthermore, the usual positioning error of other range technologies is in the range of 1-5m for Bluetooth and WIFI and 6-10m for GPS [5]. Our results showed that not only the error is in the cm range (and not meters) but also that working ranges go until at least 60 m.

In future work, it would be interesting to study more intensively the effect of the different UWB parameters in the distance measurements as well as testing the maximum ranges for each configuration. For a better characterization of the system, the RSSI readings from the sensor should be used to analyze the antenna orientation effects on the robot Localization. A better error and noise modeling would probably permit to expand the localization to larger and harder environments and higher working distances.

Regarding the Localization algorithms, further improvements can be made by testing 2D, 2.5D and 3D dimensions and the other Pozyx Algorithms as well as fuse them with the IMU data provided by the sensor.

Overall, this first work around Pozyx showed that it can be a good and effective tool to improve the robot localization in an hard outdoor GPS-denied environment given its good accuracy/cost trade-off. The still not so explored tools of the system can still help to better characterize the UWB technology and understand the effects of its configuration and the environment in the final localization.

Acknowledgment

This work is financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme within project ”POCI-01-0145-FEDER-006961”, and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) as part of project UID/EEA/50014/2013.

References

[1] *AGROB V1x – Robotics for Agriculture and Forestry – That works in steep slope viticulture.* URL: `http://agrob.inesctec.pt/`.

[2] Marcos Duarte et al. “Agricultural Wireless Sensor Mapping for Robot Localization”. In: *Robot 2015: Second Iberian Robotics Conference: Advances in Robotics, Volume 1.* Ed. by Luís Paulo Reis et al. Cham: Springer International Publishing, 2016, pp. 359–370. ISBN: 978-3-319-27146-0. DOI: 10.1007/978-3-319-27146-0_28. URL: https://doi.org/10.1007/978-3-319-27146-0_28.

[3] J. González et al. “Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach”. In: *Robotics and Autonomous Systems* 57.5 (2009), pp. 496–507. ISSN: 0921-8890. DOI: http://dx.doi.org/10.1016/j.robot.2008.10.022. URL: http://www.sciencedirect.com/science/article/pii/S0921889008001747.

[4] Kexin Guo et al. “Ultra-Wideband-Based Localization for Quadcopter Navigation”. In: *Unmanned Systems* 04.01 (2016), pp. 23–34. DOI: 10.1142/S2301385016400033. eprint: http://www.worldscientific.com/doi/pdf/10.1142/S2301385016400033. URL: http://www.worldscientific.com/doi/abs/10.1142/S2301385016400033.

[5] *Pozyx - centimeter positioning for Arduino.* URL: https://www.pozyx.io.

[6] *pozyxLabs/Pozyx-Arduino-library.* URL: https://github.com/pozyxLabs/Pozyx-Arduino-library.

[7] Filipe Neves dos Santos et al. “Towards a Reliable Robot for Steep Slope Vineyards Monitoring”. In: vol. 83. 3. Sept. 2016, pp. 429–444. DOI: 10.1007/s10846-016-0340-5. URL: https://doi.org/10.1007/s10846-016-0340-5.

[8] Heber Sobreira et al. “Robust mobile robot localization based on a security laser: an industry case study”. In: *Industrial Robot: An International Journal* 43.6 (2016), pp. 596–606. DOI: 10.1108/IR-01-2016-0026. eprint: https://doi.org/10.1108/IR-01-2016-0026. URL: https://doi.org/10.1108/IR-01-2016-0026.

[9] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics.* MIT press, 2005.

[10] Blaise Omer Yenké et al. “Adaptive Scheme for Outliers Detection in Wireless Sensor Networks”. In: *International Journal of Computer Networks and Communications Security* (2017), pp. 105–114. URL: http://www.ijcncs.org/published/volume5/issue5/p3_5-5.pdf.