Sensor Fusion Algorithm Selection for an Autonomous Wheelchair Based on EKF/UKF Comparison

Bibiana Fariña*, Jonay Toledo, and Leopoldo Acosta
Universidad de La Laguna, Tenerife, Spain; Email: bfarinaj@ull.edu.es, {jonay, leo}@isaatc.ull.es
*Correspondence: bfarinaj@ull.edu.es

***Abstract*—This paper compares two sensorial fusion algorithms based on their characteristics and performance when applied to a localization system for an autonomous wheelchair in dynamic environments. The mobile robot localization module is composed by three sensors: Encoders attached to the wheels, LIDAR and IMU. The information provided by each one is combined according to their covariance obtaining the most reliable pose estimation possible. For this purpose, it focuses on the study of two fusion algorithms, the Extended and Unscented Kalman filters, detailing their properties and operation. Both methods are implemented in the wheelchair for its comparison. The experiments carried out demonstrate how the localization results with UKF are more precise than using the EKF in a non-linear system and shows similar pose estimation when using a constant velocity model, despite the fact that the UKF needs longer execution time than the EKF.**

***Keywords*— mobile robot, localization, sensor fusion**

I. Introduction

Localization systems are one of the main modules in an autonomous robot. Without a good localization it is impossible to carry out a robust and safe navigation, so the robot will not be able to reach its destination. These systems are highly dependent on the environment for which the robot and its sensorial set are designed, and there is no universally valid solution. Good tracking results can be achieved in controlled and indoor environments, unlike in real and outdoor conditions.

The localization in mobile robots cannot be based on only one sensor. It is necessary to receive information from multiple sensors based on different operating principles, achieving a robust pose against possible sensor failures. Each one must be characterized by its measurement accuracy in order to be processed in a fusion algorithm and obtain a more precise pose in real time. In these algorithms one of the most used techniques is the Kalman Filter, which fuses the information from several sensors, characterized by their noise covariance. Likewise, there are variants of non-linear Kalman filter algorithms, such as Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF). Both of them achieve more robust localization, estimating the state of nonlinear dynamic systems. EKF is based on the Taylor series expansion [1]. However, the main disadvantage is that error increases when dealing with highly non-linear systems. The UKF fits the probability density distribution of nonlinear equations with fixed parameters through the unscented transform and therefore, avoids the loss of higher order terms in the Taylor series expansion caused by linearization [2].

In this paper, these two last algorithms are implemented in the localization system of an autonomous wheelchair for its comparison. The more suitable in terms of accuracy and computational cost will be selected as the sensor fusion algorithm.

The document is organized as follows. Section II includes a review of previous studies. Section III describes the characteristics of the prototype and the localization system. Section IV explains the fusion algorithms used. The experiments are shown in section V. Finally, the conclusions are summarized in Section VI.

II. Previous Work

Accurate real-time state estimation is a major challenge in mobile robotics to navigate safely and autonomously. This task usually takes the noisy information from sensors and, using fusion tools, achieves a better state approximation. One of the techniques mentioned in current literature is the Kalman Filter, invented in 1960 by Rudolf E. Kalman [3]. It consists of identifying the hidden state (not measurable) of a linear dynamic system subjected to additive white noise. It is a recursive algorithm that can be executed using only the input measurements, the previously computed state, and its uncertainty matrix. Therefore, due to its ability to extract useful information from noisy data and its small computation cost and memory requirements, it is used in many application areas [4].

In situations when the systems are nonlinear, the Kalman filter provides inaccurate errors and can eventually lead to tracking system divergence. Nonlinear problems can be solved with its variants; Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF). S. Konatowski and A. Pieniezny [5] present a

Manuscript received October 2, 2022; revised November 11, 2022; accepted December 4, 2022.

doi: 10.18178/ijmerr.12.1.1-7

comparison of both methods. It demonstrates experimentally how for a specific system, both techniques obtain very similar results, however in complex movements a better quality is demonstrated in the UKF. Likewise, D. Hong-de, D. Shao-wu, et al. [6] describe a performance comparison of EKF and UKF for a Classical Orbital Elements estimation, using RADAR measurements. It concludes that UKF handles model non-linearity and uncertainty more effectively than EKF. Furthermore, the non-linear filter variants have been applied to a variety of real-world situations. J. Wei and G. Qinhe [7] implement the EKF to a hydraulic circuit model and, B. Leela and R. Sai [8] present the EKF for the state estimation of a body in free fall towards the ground. As for the UKF, T. Zhu and H. Zheng [9] describe the UKF performance in a vehicle state estimation and, P. Pasek and P. Kaniewski [10] use this algorithm for a personal navigation system by range measurements between ultrawideband radio modules for positioning.

In this paper, both filters will be applied to an autonomous wheelchair, analysing its advantages and disadvantages, and selecting the most appropriate one.

III. Wheelchair Design

The fusion algorithms under study are tested on an intelligent wheelchair (Fig. 1). This wheelchair navigates safely and autonomously thanks to its different modules [11]. It can reach a defined destination by the most appropriate path, making decisions itself and avoiding obstacles while moving. For this purpose, it needs a localization system consisting of several sensors to estimate its pose in each time instant.

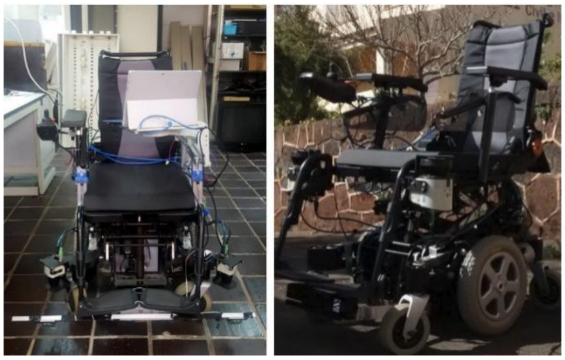


Figure 1. Autonomous wheelchair prototype.

This robot has the following sensors implemented for self-localization:

- **Encoders**: The wheels use incremental encoders coupled to the motors. The system captures the wheels movement in real time and obtains the robot pose, knowing the radii and their distance, using the odometric equations for differential locomotion. In this case, the encoder resolution is 8800 pulses per turn, with a resolution of 0.04 degrees.
- **IMU:** The Inertial Measurement Unit (MPU9250) is situated under the seat and uses a sensor set to estimate inertial magnitudes; an accelerometer, a gyroscope, and an electronic compass. For our study, the accelerometer and the magnetometer are discarded due to its low accuracy. We consider the information provided by the gyroscope, giving an accurate robot angular velocity.
- **LIDAR:** The Laser Imaging Detection and Ranging (pair of Sick TiM 551) calculates the difference between two consecutive laser scans to estimate the movement between them with a scan matching technique. The method consists of the ICP (Iterative Closet Point) algorithm [12] that finds the transformation (rotation and translation) that aligns the points of two consecutive scans and optimize it iteratively by the square error minimization. As a result, it gets the robot pose increment in a time interval.

A. Localization System

The localization system is designed with non-linear fusion algorithms which receives the state of three different sensors and their covariances as inputs; LIDAR, IMU and Encoders. The algorithm combines the sensor outputs depending on their covariances and a predefined model to estimate the robot current state. The system has been developed in a robotic operating system (ROS) and the theoretical operation of the filters used are described in section IV.

Fig. 2 shows the scheme for the localization modules implemented for its later comparison in the autonomous wheelchair. In each iteration the state (X_k) is estimated by each sensor information and the use of a fusion algorithm, which can be the EKF or UKF.

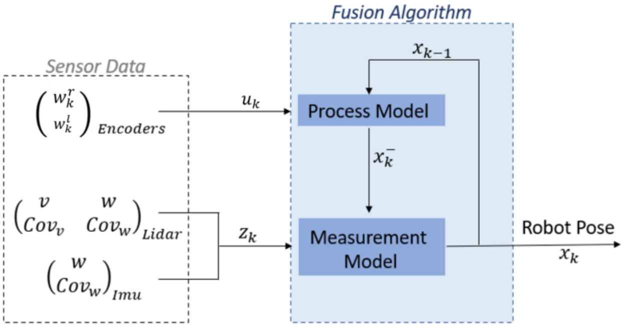

Fusion Algorithm
Sensor Data
$\begin{pmatrix} w_k^r \\ w_k^l \end{pmatrix}_{Encoders}$
u_k
x_{k-1}
Process Model
x_k^-
$\begin{pmatrix} v & w \\ Cov_v & Cov_w \end{pmatrix}_{Lidar}$
z_k
Measurement Model
Robot Pose
x_k
$\begin{pmatrix} w \\ Cov_w \end{pmatrix}_{Imu}$

Figure 2. Localization system schema implemented in the wheelchair.

By implementing both filters, we can compare their performance. The experiments will show which algorithm is more suitable for this robot localization module.

IV. Filter Algorithms

The wheelchair localization can be defined as a non-linear system with difference equations describing the estimation and observation models with additional noise,

$$\begin{gathered} X_k = f(X_{k-1}, u_k) + n_{w_k} \\ Z_k = h(X_k) + n_{v_k} \end{gathered} \tag{1}$$

Where k is the discrete-time instant, X_k is the space state we want to estimate and Z_k are the actual measurements, that is, the output sensor information. f(·) is the process function that estimate the new state from the previous one (X_{k-1}) and the model input (u_k). h(·) is the measurement function which converts internal states to measurement outputs. The vectors n_w and n_v are two zero-mean white Gaussian noise distributions with covariance matrix Q_k and R_k, respectively.

The movement model used is defined by data information of each wheel encoder and the odometry equations corresponding to a robot with differential kinematics presented in (1). The wheel velocities (v^r, v^l) are obtained using the number of each encoder ticks ($count^{l/r}$) that the electronics receive in time period (Δt), the encoder resolution (encRes) in one turn and, the wheel radii (r_r, r_l). The angular wheel velocities (w^r, w^l) can be calculated attending to this relation $v^{r/l} = w^{r/l} \cdot r^{r/l}$.

The linear v and angular w robot speeds are obtained based on $v^{r/l}$ and the wheel distance D. The orientation angle θ and the pose components (x, y) are estimated by integration.

$$\begin{gathered} v^l = w^l r_l = \frac{2\pi r_l count_k^l}{encRes\triangle t}; \quad v^r = w^r r_r = \frac{2\pi r_r count_k^r}{encRes\triangle t} \\ w = \frac{(v^l - v^r)}{D}; \quad v = \frac{(v^l + v^r)}{2} \end{gathered} \tag{2}$$

Considering the odometric process for differential robots and following the general scheme exposed in (1), it is possible to design our system model. The state vector X_k is composed by the robot pose (x, y, θ) and the linear and angular velocities (v, w). The process function f(·) used in our problem is defined in (3). The next state in k instant is estimated depending on the previous one in k-1 and the odometric formulas, using the angular speed of each wheel as inputs, $u_k(w^r_k, w^l_k)$. With this information we can estimate the next velocity states and its pose.

$$\begin{bmatrix} x \\ y \\ \theta \\ v \\ w \end{bmatrix}_k = \begin{bmatrix} x_{k-1} + v_{k-1} \triangle t\cos(\theta_{k-1} + w_{k-1} \triangle t) \\ y_{k-1} + v_{k-1} \triangle t\sin(\theta_{k-1} + w_{k-1} \triangle t) \\ \theta_{k-1} + w_{k-1} \triangle t \\ \frac{w_k^r r_r + w_k^l r_l}{2} \\ \frac{w_k^r r_r - w_k^l r_l}{D} \end{bmatrix} \tag{3}$$

Regarding the observation model, LIDAR speed measurements ($z(v)_{LIDAR}$, $z(w)_{LIDAR}$) and the angular IMU velocity ($z(w)_{IMU}$) are incorporated as Z_k measurements. The sensor outputs directly represent the state and therefore, we can define the function h(·) by a matrix that maps the real state in the observation space and is formed by 0 and 1. In equation (4) the two models used for both, the LIDAR $h(\cdot)_L$ and the IMU $h(\cdot)_I$, are presented.

$$h_L = \begin{bmatrix} 0 & 0 & 0 & 0 & w_k \end{bmatrix} \quad h_I = \begin{bmatrix} 0 & 0 & 0 & v_k & 0 \\ 0 & 0 & 0 & 0 & w_k \end{bmatrix} \tag{4}$$

A. EKF

The Extended Kalman Filter (EKF) estimates the robot state in real time using nonlinear system models. For this purpose, the filter linearizes it around the current covariance and mean, by the Taylor series. It consists in calculating the Jacobian matrix of the transition and observation models, that have to be differentiable. The equation (5) shows the Jacobian matrix A in time instant k for our model system that depends on the state X_k.

$$A_k = \begin{bmatrix} 1 & 0 & A_{13} & A_{14} & A_{15} \\ 0 & 1 & A_{23} & A_{24} & A_{25} \\ 0 & 0 & 1 & 0 & A_{35} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5}$$

$$\begin{aligned} A_{13} &= -v_k \triangle t\sin(\theta_k + w_k \triangle t) \\ A_{14} &= \triangle t\cos(\theta_k + w_k \triangle t) \\ A_{15} &= -v_k \triangle t^2\sin(\theta_k + w_k \triangle t) \\ A_{23} &= v_k \triangle t\cos(\theta_k + w_k \triangle t) \\ A_{24} &= \triangle t\sin(\theta_k + w_k \triangle t) \\ A_{25} &= v_k \triangle t^2\cos(\theta_k + w_k \triangle t) \\ A_{35} &= \triangle t \end{aligned}$$

The algorithm can be divided following the same structure of the Kalman filter for linear systems, having two stages: prediction and correction steps. In Eq. (6) are presented the equations that describe the EKF. X_k^- and P_k^- are the state and covariance estimated a priori, where Q_k is the process noise. In the correction step, H_k is the Jacobian matrix of the observation function h(·) forming, in this case, a matrix whose elements are 0 and 1. K_k is the filter gain, R_k the measurement noise and (X_k, P_k) the current state calculated and its covariance.

$$\begin{gathered} X_k^- = f(X_{k-1}, u_k) \\ P_k^- = A_k P_{k-1} A_k^T + Q_k \\ K_k = P_k^- H_k^T [H_k P_k^- H_k^T + R_k]^{-1} \\ X_k = X_k^- + K_k[Z_k - h(X_k^-)] \\ P_k = [I - K_k H_k] P_k^- \end{gathered} \tag{6}$$

B. UKF

The Unscented Kalman Filter (UKF) is a stochastic filter whose goal is to estimate the new state in non-linear systems. This algorithm is based on the unscented transformation [13], a method for calculating the statistics of a random variable which undergoes a nonlinear transformation. The state distribution is represented using a set of sample points, called sigma points, the state mean and its covariance.

The algorithm is an iterative process where sigma points are calculated in previous instant k-1 to estimate the current robot state in k time. The first step of the algorithm is to define the weights associated considering the propagation of a random variable X (that represents the state) of dimension N (in our case N = 5) through a nonlinear function. Eq. (7) describes the weight vectors, W_m and W_c, belonging to the mean and covariance, respectively. Each one has a length of 2N+1. Regarding the parameters used, their values are defined by the user.

A determines the spread of the sigma points, κ is another scaling parameter and β is used to incorporate prior knowledge of the distribution.

$$\begin{gathered} \lambda = \alpha^2(N+\kappa) - N \\ W_c^0 = \frac{\lambda}{N+\lambda} + (1-\alpha^2+\beta); \quad W_c^i = \frac{1}{2(N+\lambda)} \\ W_m^0 = \frac{\lambda}{N+\lambda}; \quad W_m^i = \frac{1}{2(N+\lambda)} \quad i = 1 \ldots 2N \end{gathered} \tag{7}$$

Once the system weights have been calculated and state initial conditions are defined, the algorithm can be divided following the same structure of Kalman filters, having two stages: prediction and correction steps.

The prediction process is,

$$\begin{gathered} S_{k-1}^0 = X_{k-1} \\ S_{k-1}^j = X_{k-1} + \eta\sqrt{P_{k-1}}, \quad S_{k-1}^{j+N} = X_{k-1} - \eta\sqrt{P_{k-1}} \\ S_k^f = f(S_{k-1}, u_k); \\ X_k^- = \sum_{i=1}^{2N} W_m^i S_{i,k}^f \\ P_k^- = \sum_{i=1}^{2N} W_c^i \left(S_{i,k}^f - X_k^-\right)\left(S_{i,k}^f - X_k^-\right)^T + Q_k \end{gathered} \tag{8}$$

The respective sigma points (S_{k-1}) are calculated from the previous state (X_{k-1}, P_{k-1}), where j=0...N and $\eta = \sqrt{\lambda + N}$. S forms a matrix whose dimension is (2N+1, N) where the first element corresponds to X, the next N elements to X+ $\eta\sqrt{P}$ and the following N to X- $\eta\sqrt{P}$. Once S_{k-1} has been calculated, they are passed through the non-linear function obtaining the set S^f_k. With this information and the respective weights, the state estimated a priori (X_k^-) and its covariance (P_k^-) are calculated, knowing that Q is the process noise.

The correction step is,

$$\begin{gathered} S_k^z = h(S_k); \quad \hat{Z}_k = \sum_{i=1}^{2N} W_m^i S_{i,k}^z \\ Pzz = \sum_{i=1}^{2N} W_c^i \left(S_k^z - \hat{Z}_k\right)\left(S_k^z - \hat{Z}_k\right)^T + R_k \\ P_{xz} = \sum_{i=1}^{2N} W_c^i \left(S_k - X_k^-\right)\left(S_k^z - \hat{Z}_k\right)^T \\ K_k = P_{xz} P_{zz}^{-1} \\ x_k = X_k^- + K_k\left(Z_k - \hat{Z}_k\right) \\ P_k = P_k^- - K_k P_{zz} K_k^T \end{gathered} \tag{9}$$

In this part, S is recalculated from the state estimated in the previous step X_k^-, defining S_k. This data is introduced in the observation matrix, obtaining the set S^z_k. With this information, the estimated measurement $\hat{Z}_k$, the P_{zz} covariance, the cross covariance P_{xz} and the filter gain K are calculated. The current robot poses, defined by the state and its covariance (X_k, P_k), is obtained from the previous data and the sensor measurements (Z_k).

Regarding the UKF implementation, the variables that control the propagation of the sigma points have been defined as; α=0.001 and κ = 0. Likewise, the variable β is equal to 2.

V. Results

The comparison of both fusion algorithms has been performed through several experiments, implementing the localization systems in a wheelchair and executing different routes. Fig. 3 shows the map where they were carried out. It is a corridor where the chair goes to the end, makes a U-turn and ends at the same departure point. In order to compare the real travel with the estimated ones, we have a Velodyne HDL 32 as ground truth. It has been installed in the upper part of the chair and consists of 32 lidars with a precision of 2 cm. It allows to calculate the path with high reliability and to create a point cloud as the environment map using the LOAM slam algorithm [14].

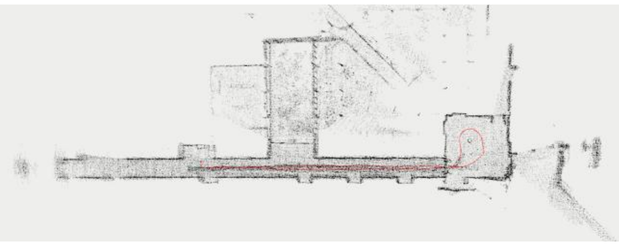


Figure 3. A-LOAM Map. Red line is the wheelchair pose.

To test the algorithms, four different approaches have been taken; EKF with non-linear model, as explained in section IV, EKF with constant velocity model considering odometric data as another measurement signal, UKF with nonlinear model and UKF with linear constant velocity model. The constant velocity model is used sometimes as a universal solution to integrate multiple measurements without the need to implement a specific model and avoiding the matrix calculus.

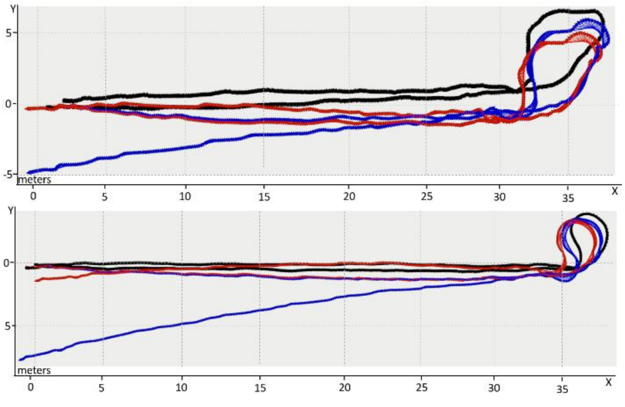


Figure 4. localization results using EKF (blue) and UKF (red) using the odometry model and the velodyne (black) as reference.

Regarding the odometric model, Fig. 4 shows two representative examples of the results obtained when testing both filters in two different paths around the study area. The black line is the ground truth result, the red line is the chair pose estimated by the UKF and the blue one, the route calculated by the EKF. As can be observed a priori, the UKF achieves a better pose in both tests, approaching the velodyne results. It verifies what is mentioned in [15] and [16] where demonstrate a considerable advantage of the UKF in terms of precision and more insensitive to measurement errors in non-linear systems.

In Fig. 5, we can analyze the results obtained by both filters with the linear constant velocity system. If we compare the results of the previous nonlinear model, with this case, we can see how the estimated pose of both filters in the linear model accuracy are less, diverging from the desired position.

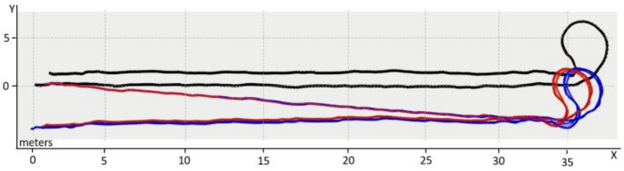


Figure 5. Localization results using EKF (blue) and UKF (red) using the constant velocity model and the velodyne (black) as reference.

The operation validation has been conducted by analyzing six experiments changing the characteristics and paths. Each one has been repeated ten times collecting the data obtained by each filter. The error between the pose estimated by the algorithms and the one provided by the velodyne is calculated using the RMSE (Root Mean Squared Error). Table I shows the results obtained using the nonlinear model and the linear one. The data shows the RMSE of the pose collected at each instant ($(\delta_D)_{RMS}$, $(\delta_\theta)_{RMS}$) and the final pose error mean estimated in the experiments ($(\delta_D)_{F.\ Pose}$, $(\delta_\theta)_{F.\ Pose}$). In the odometry model, even though the results from both filters are similar, there is an improvement in terms of precision in the UKF due to the smaller mean error when compared to the EKF. Likewise, we analyzed how the filters would work in the constant velocity model, where the information from the three sensors is considered as measurements. As one would expect, both algorithms produce similar behaviors with this model, with almost identical mean errors and with no appreciable difference between them. Furthermore, higher errors are obtained in this last case compared to the nonlinear model.

TABLE I. ROOT MEAN SQUARED ERROR OF BOTH MODELS

		Odometry Model	Constant velocity
EKF	$(\delta_D)_{RMS}$	2.849247 m	3.375612 m
	$(\delta_\theta)_{RMS}$	0.150283 rad	0.144008 rad
	$(\delta_D)_{F.\ Pose}$	6.112 m	4.544326 m
	$(\delta_\theta)_{F.\ Pose}$	0.22761rad	0.19934 rad
UKF	$(\delta_D)_{RMS}$	1.89529 m	3.465438 m
	$(\delta_\theta)_{RMS}$	0.095109 rad	0.10883 rad
	$(\delta_D)_{F.\ Pose}$	3.400253 m	4.89109 m
	$(\delta_\theta)_{F.\ Pose}$	0.13795 rad	0.13129 rad

Once the validity of our odometric system model has been verified. The advantages of using the UKF against the EKF have been compared by studying the error obtained in a route using the Normalized Estimation Error Squared (NEES) metric that considers the state covariance and is defined as,

$$e_i = \left(X_i - \hat{X}_\iota\right)^T P_i^{-1}\left(X_i - \hat{X}_\iota\right) \qquad (10)$$

where e_i is de NEES error in an instant i, X_i is the state of reference, $\hat{X}_i$ is the state estimated for the filter and P_i is the respective estimated covariance. Fig. 6 shows the NEES results of the linear and angular velocities estimated by both filters at each instant with respect to time. The graphics prove once again the best results of the UKF with respect to the EKF.

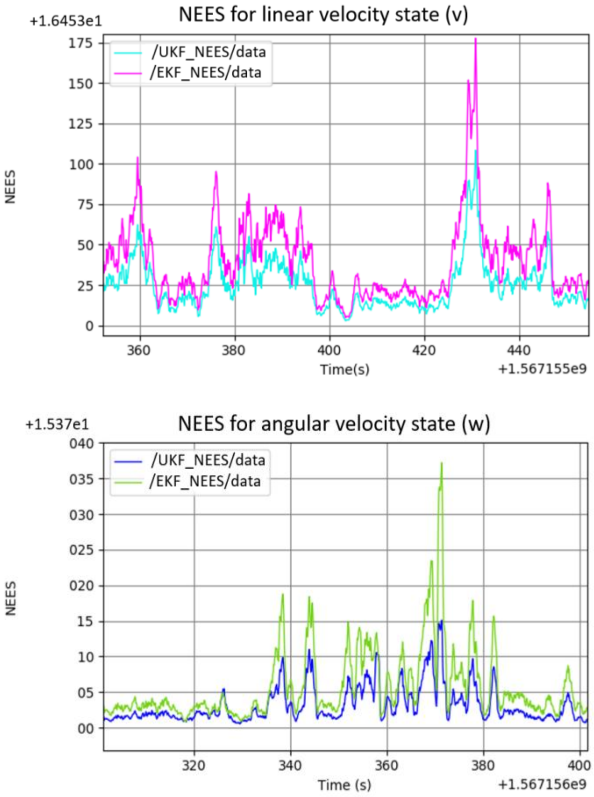


Figure 6. NEES for velocity states in a specific path.

The execution time in the estimation and measurement parts has been considered for the comparison of both Kalman filters. Table II shows the time they take to run over a path of approximately 80 meters. As we can observe, the execution time in the UKF is greater than the EKF, which is twice as fast. So, despite obtaining better results with the UKF in nonlinear systems and estimating very similar poses for linear problems, the UKF takes longer to execute than the EKF. However, if we focus on the average time that each algorithm takes in each loop iteration, this difference is not significant, calculating the robot pose in real time without notable delays between both algorithms.

TABLE II. EXECUTION TIME OF BOTH ALGORITHMS

	EKF	UKF
Average time per iteration (ms)	0.1347 ms	0.4051 ms
Execution time (s)	0.9364 s	2.9985 s

VI. Conclusion

This paper describes a localization module applied to an autonomous wheelchair. The system consists of fusing the information from three sensors (IMU, LIDAR and encoders) to estimate the robot's pose in real time. It studies the advantages and disadvantages of applying the non-linear variants of the Kalman filter, such as the EKF and UKF. The experimental results demonstrate how for the nonlinear estimation model designed by the odometric equations, the UKF is able to achieve more accurate real-time poses compared to the EKF. However, if we focus on the time, the UKF requires a longer time than the EKF to execute. Furthermore, if we consider a constant velocity linear model and assume the outputs of the three sensors as measurements, the behavior of both filters is very similar, obtaining the same results with lower accuracy than when a nonlinear model is used.

Consequently, due to the high reliability required in the localization system for an autonomous wheelchair, accuracy is a priority over computation time. We can assume for the case we are considering, that the UKF, despite its execution time, obtains better results than the EKF in non-linear systems providing a precise pose closer to the real robot state. Furthermore, the time increment in each iteration is not representative. So, the localization algorithm selected as the most appropriate is the UKF using the nonlinear odometry model.

Conflict of Interest

The authors declare no conflict of interest.

Author’s Contribution

This work was carried out under the general supervision of L. Acosta who conducted all the research. J. Toledo Toledo was responsible for the dataset and the tests with the wheelchair. B.Fari ña designs the system and the code implementation. All authors had contributed equally to the writing of the paper and approved the final version.

References

[1] K. Chadaporn, J. Baber, and M. Bakhtyar, “Simple example of applying extended kalman filter,” in *Proc. 1st International Electrical Engineering Congress*, 2014.

[2] E. A. Wan and R. V. D. Merwe, “The unscented Kalman filter for nonlinear estimation,” in *Proc. the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 2000.

[3] Y. Pei, S. Biswas, D. S. Fussell, and K. Pingali, *An Elementary Introduction to Kalman Filtering*, arXiv, 2017.

[4] C. Urrea and R. Agramonte, “Kalman filter: Historical overview and review of its use in robotics 60 years after its creation,” *J. Sensors*, vol. 2021, pp. 9674015:1-9674015:21, 2021.

[5] S. Konatowski and A. Pieniezny, “A comparison of estimation accuracy by the use of KF, EKF, UKF filters,” *Computational Methods and Experimental Measurements XIII*, 2007.

[6] D. Hong-de, D. Shao-wu, C. Yuan-cai, and W. Guang-bin, “Performance comparison of EKF/UKF/CKF for the tracking of ballistic target,” *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 10, November 2012.

[7] J. Wei and G. Qinhe, “The application of EKF in state estimation of the hydraulic circuit,” *2009 International Forum on Computer Science-Technology and Applications*, 2009.

[8] B. Leela and R. Sai, “Application of extended kalman filter for a free falling body towards earth,” *International Journal of Advanced Computer Science and Applications-IJACSA*, vol. 2, pp. 134-140, January 2011.

[9] T. Zhu and H. Zheng, “Application of unscented Kalman filter to vehicle state estimation,” in *Proc. 2008 ISECS International Colloquium on Computing, Communication, Control, and Management*, 2008.

[10] P. Pasek and P. Kaniewski, “Unscented Kalman filter application in personal navigation,” *Radioelectronic Systems Conference 2019*, 2020.

[11] B. Fari ña, J. Toledo, J. Estevez, and L. Acosta, “Improving robot localization using doppler-based variable sensor covariance calculation,” *Sensors*, vol. 20, p. 2287, April 2020.

[12] S. Bonnabel, M. Barczyk, and F. Goulette, “On the covariance of ICP-based scan-matching techniques,” in *Proc. 2016 American Control Conference (ACC)*, 2016, pp. 5498-5503.

[13] M. Gasperin and D. Juricic, “Transformation in nonlinear system identification,” *IFAC Proceedings Volumes*, vol. 44, pp. 4428-4433, 2011.

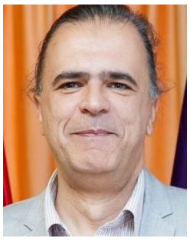
[14] J. Zhang and S. Singh, “LOAM: Lidar odometry and mapping in real-time,” *Robotics: Science and Systems*, 2014.

[15] J. S. Nanda and S. Ghosh, “Performance comparison of EKF and UKF for estimation of COE using Kozai mechanism,” in *Proc. 2021 2nd International Conference on Range Technology (ICORT)*, 2021.

[16] S. Konatowski, P. Kaniewski J. Matuszewski, “Comparison of Estimation Accuracy of EKF, UKF and PF Filters,” *Annual of Navigation*, vol. 23, December 2016.

Bibiana Fari ña received his B.Sc. Degree electronic engineering from University of La Laguna, Spain in 2018. She received his M.Sc. in Robotics in Polytechnic university of Madrid. She is currently completing his PhD's degree in Robotics at the University of La Laguna. She focuses on localization fusion algorithms for autonomous mobile robots. Her research interests include machine learning, mobile robots, smart sensor systems, sensor fusion.

Jonay Toledo received the Ph.D. degree in automatic control from the University of La Laguna in 2008. He is currently a Full Professor with the University of La Laguna. His research interests include mobile robots, autonomous vehicles, automatic control, and embedded systems. He has authored several conference and journal papers in robotics, automatic control and artificial intelligence.



Leopoldo Acosta received the Ph.D. degree in automatic control from the University of La Laguna in 1991. He is currently a Full Professor with the University of La Laguna, Spain. He has been involved in several competitive nationally funded research projects related to artificial intelligence and robotics, six of them as the Project Leader. He focuses on robotics, control engineering and intelligent systems.