

FIDD Bearing-Only SLAM

Rodrigo Munguia^{1,2} and Antoni Grau¹

¹Department of Automatic Control, UPC, c/ Pau Gargallo, 5 E-08028 Barcelona Spain,

²Universidad de Guadalajara, Mexico,
{rodrigo.munguia,antoni.grau}@upc.edu

Abstract- Simultaneous Localization and Mapping (SLAM) is perhaps the most fundamental problem to solve in robotics in order to build truly autonomous mobile robots. The sensors have a large impact on the algorithm used for SLAM. In this work a novel method, called *Filtered Inverse Depth Delayed (FIDD) Initialization* which is intended for initializing new features in Bearing-Only SLAM systems. Unlike range sensors which provide range and angular information, a bearing sensor (e.g. cameras) measures only the bearing (angular information) of features. Therefore depth information (range) cannot be obtained in a single step. This fact has propitiated the emergence of a new family of SLAM algorithms: the Bearing-Only SLAM methods, which mainly rely in especial techniques for features system-initialization in order to enable the use of bearing sensors (as cameras) in SLAM systems. The proposed approach is based in an inverse depth parameterization and delayed initialization scheme. The main idea is to incorporate to the SLAM process, an extra uncorrelated filter, which progressively incorporates the new bearing measurements needed to estimate the full state of each new feature. Several simulations are given in order to show the performance of the proposed approach.

I. INTRODUCTION

The on-line robot estimation position from measurements of self-mapped features is a class of problem called, in the robotics community, as Simultaneous Localization and Mapping (SLAM) problem, which is one of the fundamental problems of robotics. The SLAM consists in incrementally building a consistent map of the environment and at the same time localizing the position of the robot while it explores its world. The robot's sensors have a large impact on the algorithm used for SLAM. Early SLAM approaches focused on the use of range sensors as sonar rings or lasers. Nevertheless there are some disadvantages with the use of range sensors in SLAM: correspondence or data association is difficult; they are expensive and generally limited to 2D maps and computational overhead due to large number of features (see [1, 2] for a complete review).

The aforementioned issues have propitiated that recent work is moving towards the use of cameras as the primary sensing modality. Cameras have become more and more interesting for the robotic research community, because they yield a lot of information allowing reliable data association. Cameras are well adapted for embedded systems: they are light, cheap and power saving. Using vision, a robot can localize itself using common objects as landmarks.

On the other hand, while range sensors (i.e. laser) provide range and angular information, a camera is a projective sensor which measures the bearing of images features. Therefore

depth information (range) cannot be obtained in a single frame. This fact has propitiated the emergence of a new family of SLAM methods: The Bearing-Only SLAM methods, which mainly rely in especial techniques for features system-initialization in order to enable the use of bearing sensors (as cameras) in SLAM systems.

It is important to note that Bearing-Only SLAM is not limited to the use of cameras as primary sensors, in an authors' previous work [3], a Sound-Based SLAM system is proposed where sound sources are used as map features and thus showing the viability on the inclusion of the hearing sense in SLAM and the use of alternative bearing sensors.

In recent years several important improvements and variants to this kind of methods have appeared [4, 5]. Different schemes for increasing the working space [6, 12] also have appeared. Nevertheless the initialization process of new features is still the most important problem for addressing in Bearing-Only SLAM in order to improve the robustness.

In [7] a multi-hypothesis method based on a particle filter to represent the initial depth of a feature is proposed. This work gives good results. However its application in large environments is not straightforward, as it would require a huge number of particles. In [8] is proposed a delayed multi-hypothesis method based in a sum of Gaussian mixture for depth estimation, but it uses odometry as an additional sensor. The work in [9] is based in the FastSLAM algorithm, where the pose of the robot is represented by particles and a set of Kalman filters refine the estimation of the features, this approach is unable to code distant points.

In [10], transition from partially to fully initialized features need not to be explicitly tackled, making it suitable for direct use in EKF framework for sparse mapping. In this approach the features are initialized in the first frame observed (undelayed initialization) with an initial fixed inverse depth (ID) and uncertainty, determined heuristically to cover ranges from nearby to infinity, so distant points can be coded. Due to the clarity and scalability of this method, this approach is a good option for monocular-SLAM implementation. On the other hand, in experiments using the above method, it often happens that the inverse depth becomes negative after a Kalman update, due to the observation noise that predominates over the update of the depth. Moreover, when an initial metric reference is used in order to recover/set the scale of the map (very relevant for robotics applications), initial fixed parameters (inverse depth and uncertainty) must be tuned in order to ensure convergence.

The issues mentioned above suggest us that the initial inverse depth and their associated initial uncertainty of the new features added to the map could be treated before to be added to the system state instead of use fixed initial depth and uncertainty. In authors' previous work [11] a delayed version of [10] is proposed. In this case, initial depth and uncertainty of each feature are dynamically estimated prior to add the new landmark in the stochastic map. The experimental results of [11] shows that wait until some information is gathered priors to add a new feature to the stochastic map, can improves the robustness of the ID Bearing-Only SLAM method. On the other hand, the combination of the direct triangulation technique used in [11] and the implicit uncertainty of the sensor readings can lead in some cases to suboptimal estimations of the new feature's state.

In this work, we present a novel method, called (FIDD) *Filtered Inverse Depth Delayed Initialization* which is intended for initializing new features in Bearing-Only SLAM systems. This novel method is inspired by improves the performance of the approach presented in our previous work [11]. The main idea is to incorporate to the SLAM process, an extra uncorrelated filter, which progressively incorporates the new bearing measurements needed to estimate the full state of each new feature.

II. INVERSE DEPTH (ID) BEARING-ONLY SLAM

A. Sensor Motion Model

Let us consider a bearing sensor, with a limited field of view, moving freely in 2DOF. The sensor state \hat{x}_v is defined by:

$$\hat{x}_v = [x_v, y_v, \theta_v, v_x, v_y, v_\theta]^T \quad (1)$$

where $[x_v, y_v, \theta_v]$ represents the center position and orientation of the sensor and $[v_x, v_y, v_\theta]$ denoting linear and angular velocity.

At every step it is assumed an unknown linear an angular acceleration with zero mean and known covariance Gaussian processes, a^W and α^W , producing an impulse of linear and angular velocity:

$$n = \begin{pmatrix} V_x^W \\ V_y^W \\ V_\theta^W \end{pmatrix} = \begin{pmatrix} a_x^W \Delta t \\ a_y^W \Delta t \\ \alpha_\theta^W \Delta t \end{pmatrix} \quad (2)$$

The sensor motion prediction model is:

$$f_v = \begin{bmatrix} x_{v(k+1)} \\ y_{v(k+1)} \\ \theta_{v(k+1)} \\ v_{x(k+1)} \\ v_{y(k+1)} \\ v_{\theta(k+1)} \end{bmatrix} = \begin{bmatrix} x_{v(k)} + (v_{x(k)} + V_x^W) \Delta t \\ y_{v(k)} + (v_{y(k)} + V_y^W) \Delta t \\ \theta_{v(k)} + (v_{\theta(k)} + V_\theta^W) \Delta t \\ v_{x(k)} + V_x^W \\ v_{y(k)} + V_y^W \\ v_{\theta(k)} + V_\theta^W \end{bmatrix} \quad (3)$$

An Extended Kalman Filter propagates the sensor pose and velocity estimates, as well as feature estimates.

B. Features Definition and Measurement

The complete state \hat{x} that includes the features \hat{y} is made of:

$$\hat{x} = [\bar{x}_v, \hat{y}_1, \hat{y}_2, \dots, \hat{y}_n]^T \quad (4)$$

where a feature \hat{y} represents a feature i defined by the 4-dimension state vector:

$$\hat{y}_i = [x_i, y_i, \theta_i, \rho_i]^T \quad (5)$$

which models a point located at:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} + \frac{1}{\rho_i} m(\theta_i) \quad (6)$$

where x_i, y_i is the sensor center coordinates when the feature was first observed; and θ_i represents the azimuth (respect to the world reference W) for the directional vector. The point depth d_i along the ray is coded by its inverse $\rho_i = 1/d_i$.

The use of an inverse depth parameterization for bearing-only SLAM can improve the linearity of the measurement equation even for small changes in the sensor position (corresponding to small changes in the parallax angle), this fact allows a Gaussian distribution to cover uncertainty in depth which spans a depth range from nearby to infinity. It is well known the relevance of a good uncertainty Gaussian representation in a scheme based in EKF.

The different locations of the sensor, along with the location of the already mapped features, are used to predict the feature angle h_i^z (angle describing the direction of the feature in the sensor coordinate frame).

$$h_i = \text{atan2} \left(\frac{1}{\rho_i} \sin \theta_i + y_i - y_v, \frac{1}{\rho_i} \cos \theta_i + x_i - x_v \right) \quad (7)$$

atan2 is a two-argument function that computes the arctangent of y/x given y and x , within a range of $[-\pi, \pi]$. At this stage it is assumed that the bearing sensor is capable of tracking and discriminating between the landmarks, in other words, the data association problem is obviated.

In implementation using real data, features search could be constrained to regions around the predicted h_i . These regions are defined by the innovation covariance matrix

$$S_i = H_i P_{k+1} H_i^T + R \quad (8)$$

where H_i is the Jacobian of the sensor model with respect to the state, P_{k+1} is the prior state covariance, and measurements z are assumed corrupted by zero mean Gaussian noise with covariance R .

As it was stated before, depth information cannot be obtained in a single measurement when bearing sensors are used. To infer the depth of a feature, the sensor must observe it repeatedly as it freely moves through its environment, estimating the angle from the feature to its center. The difference between angles measurements is the feature parallax. Actually, parallax is the key that allows to estimating features depth. In the case of indoor sequences, centimeters are enough to produce parallax, on the other hand, the more distant the feature, the more the sensor has to

travel to produce parallax. Therefore, in order to incorporate new features to the map, especial techniques for features system-initialization are needed in order to enable the use of bearing sensors in SLAM systems.

III. FILTERED & DELAYED ID INITIALIZATION

C. Undelayed Stage

Parallel to the system state (represented by the state vector \hat{x}), a state vector \hat{x}_{can} is used for estimating (via an extra linear Kalman Filter) the feature depth of each landmark \hat{y}_i . The state \hat{x}_{can} is not directly correlated with the map.

When a feature is detected the first time k , some part of the current state \hat{x} and covariance matrix P together with the sensor measurement are stored, this data λ (called candidate points) is composed by:

$$\lambda_i = (x_1, y_1, \theta_1, z_1, \sigma_1^x, \sigma_1^y, \sigma_1^\theta) \quad (9)$$

The values x_1 , y_1 and θ_1 represent the current robot position; σ_1^x , σ_1^y and σ_1^θ represent their associated variances taken from the state covariance matrix P_k and z_1 is the first bearing measurement to the landmark.

Every time a candidate point λ_i is detected-stored, the state \hat{x}_{can} is augmented as:

$$\hat{x}_{\text{can}} = \begin{bmatrix} \lambda_1' \\ \lambda_2' \end{bmatrix} \quad \hat{x}_{\text{can_new}} = \begin{bmatrix} \lambda_1' \\ \lambda_2' \\ \lambda_i' \end{bmatrix} \quad (10)$$

where λ_i' is a 3-dimension vector which relates a portion of \hat{x}_{can} with each candidate point λ_i . For each candidate point λ , note that the semicolon is used for distinguish between data stored statically and data that will be estimated by the filter. λ_i' is composed by:

$$\lambda_i' = [\alpha_i, \Delta\alpha_i, \rho_i]' \quad (11)$$

For each λ_i' , α_i is the estimated parallax, $\Delta\alpha_i$ is the rate of change in parallax and ρ_i is the estimated inverse depth.

The covariance matrix of \hat{x}_{can} , P_{can} is augmented simply by:

$$P_{\text{can_new}} = \begin{pmatrix} P_{\text{can}} & 0 \\ 0 & R_c \end{pmatrix} \quad (12)$$

The three initial values of λ_i are set to zero, and the initial values of R_c have been heuristically determined as: $R_c = \text{diag}(.01, .01, 1)$.

D. Delayed Stage

While the sensor moves through its environment, it can observe repeatedly a candidate λ_i at each iteration generating a new angle measurement z . All these new measurements are successively added to the linear Kalman Filter, responsible for estimating \hat{x}_{can} , (Fig.1 upper plot) in order to infer the landmark depth. For each new measurement z_i of a candidate λ_i , an iteration of the filter is executed.

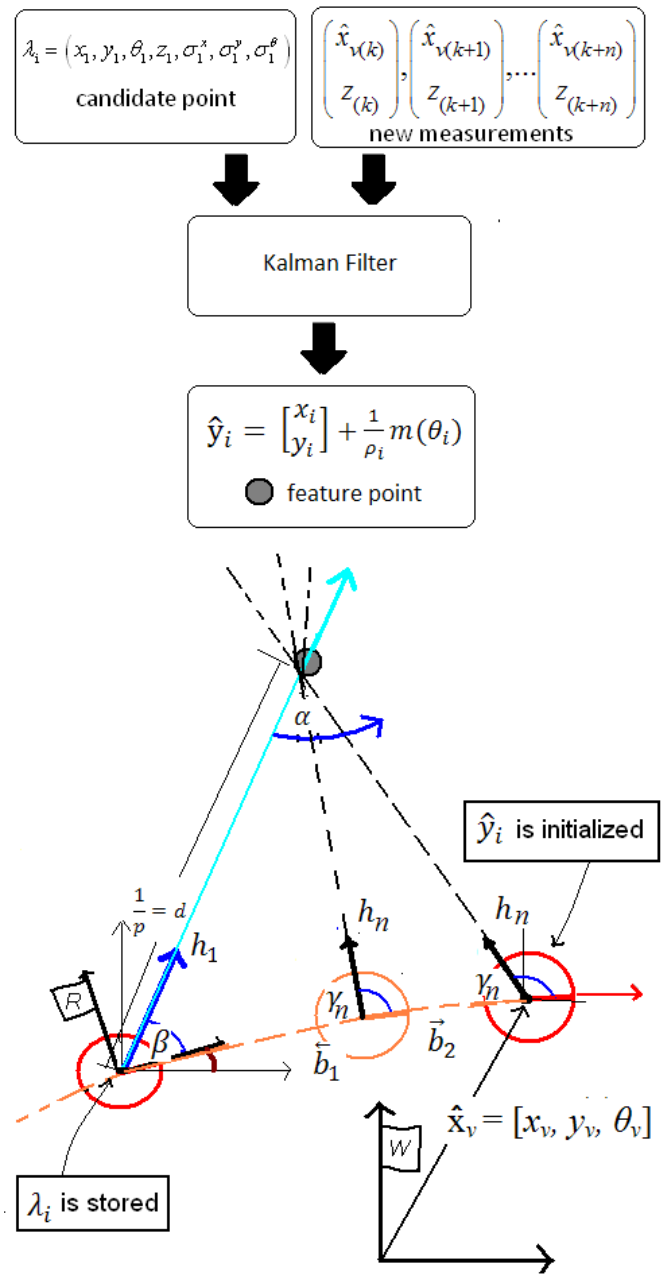


Fig.1. (Upper plot) an extra uncorrelated linear Kalman Filter progressively incorporates the new bearing measurements needed to estimate the full state of each new feature. (Lower plot) parameterization used.

The state transition model for each λ_i' is:

$$\hat{x}_{\text{can_i}(k+1)} = \begin{bmatrix} \alpha_{i(k+1)} \\ \Delta\alpha_{i(k+1)} \\ \rho_{i(k+1)} \end{bmatrix} = \begin{bmatrix} \alpha_{i(k)} + \Delta\alpha_{i(k)} \\ \Delta\alpha_{i(k)} \\ \rho_{i(k)} \end{bmatrix} \quad (13)$$

A process noise $w_k \sim N(0, Q_k)$ is considered. In experiments: $Q_k = \text{diag}(8e^{-7}, 10e^{-9})$ have been used.

The measurement prediction model is directly obtained from the state. On the other hand, the measurements z_{can} used

to update the filter are a function of: i) the feature $\hat{y}_{L(i)}$, ii) the sensor state \hat{x}_v and iii) the current measurement z_i .

$$z_{can} = \begin{bmatrix} z_\alpha \\ z_\rho \end{bmatrix} = f_z(\lambda_i, \hat{x}_v, z_i) \quad (14)$$

z_α and z_ρ are estimated as follows (Fig.1 lower plot):

$$z_\alpha = \pi - (\beta + \gamma) \quad z_\rho = \frac{\sin \alpha}{b \sin \beta} \quad (15)$$

Where $\alpha = z_\alpha$. The angle β is determined by the directional unitary vector h_1 and the vector b_1 defines the base-line b in the direction of the sensor trajectory. The angle γ is determined in a similar way as β but using the directional unitary vector h_n and the vector b_2 defining the base line in the opposite direction of the sensor trajectory by:

$$\beta = \cos^{-1} \left(\frac{h_1 \cdot b_1}{\|h_1\| \|b_1\|} \right) \quad \gamma = \cos^{-1} \left(\frac{h_n \cdot b_2}{\|h_n\| \|b_2\|} \right) \quad (16)$$

where $(h_1 \cdot b_1)$ is the dot product between h_1 and b_1 . The directional vector h_1 , expressed in the absolute frame W , points from the sensor location to the direction when the landmark was observed for the first time, and is computed using the data stored in λ_i denoting the bearing z_i . The directional vector h_n expressed in the absolute frame W is computed in a similar way as h_1 but using the current sensor position \hat{x}_v and the current measurement z_i .

$$h_1 = \begin{bmatrix} \cos(\theta_1 + z_1) \\ \sin(\theta_1 + z_1) \end{bmatrix} \quad h_n = \begin{bmatrix} \cos(\theta_v + z_i) \\ \sin(\theta_v + z_i) \end{bmatrix} \quad (17)$$

b_1 is the vector representing the robot base-line between the robot center position x_1, y_1 stored in λ_i where the point was first detected and the current sensor center (x_v, y_v) . b_2 is equal to b_1 but pointing to the opposite direction. The base-line b is the module of b_2 or b_1 :

$$b = \|b_1\| = \|b_2\| \quad (18)$$

$$b_1 = [(x_v - x_1), (y_v - y_1)] \quad b_2 = [(x_1 - x_v), (y_1 - y_v)]$$

The implicit uncertainties in the estimation of the function f_z are used to compose the error measurement covariance matrix R_{can} :

$$R_{can} = \nabla f_z(P_t) \nabla f_z' \quad (19)$$

where ∇f_z is the Jacobian of f_z with respect to z_{can} . P_t is formed by:

$$P_t = \begin{pmatrix} P_{\hat{x}_v} & 0 & 0 \\ 0 & \lambda_\sigma & 0 \\ 0 & 0 & \sigma_z^2 \end{pmatrix} \quad (20)$$

$P_{\hat{x}_v}$ is the submatrix of P corresponding to the covariance of the sensor state \hat{x}_v . $\lambda_\sigma = (\sigma_1^x, \sigma_1^y, \sigma_1^\theta)$ are the variances stored in λ_i . σ_z is the error variance of the bearing sensor.

R_{can} are used in the Kalman update equations for estimating the innovation covariance matrix $S_{can(i)}$.

E. Full Initialization

The state \hat{x}_{can} encloses the parallax α_i and inverse depth ρ_i estimations for each candidate point λ_i .

Fig. 2 shows the evolution of parameters α_i (upper plot) and $d=1/\rho_i$ (lower plot) and its uncertainty estimated by the linear Kalman filter, for a candidate point with a depth of $d=50$ units. The boundary uncertainties at 3σ are indicated in blue color. The filtered values are depicted in red color. Also note in green color the raw measurements z_{can} (taken from equation 14). In these graphics it can be clearly appreciated how the estimation of depth d is directly influenced by the parallax; for the near feature, only a few steps are needed to producing parallax and thus d converge rapidly to its real value. Also note that the uncertainty is rapidly minimized.

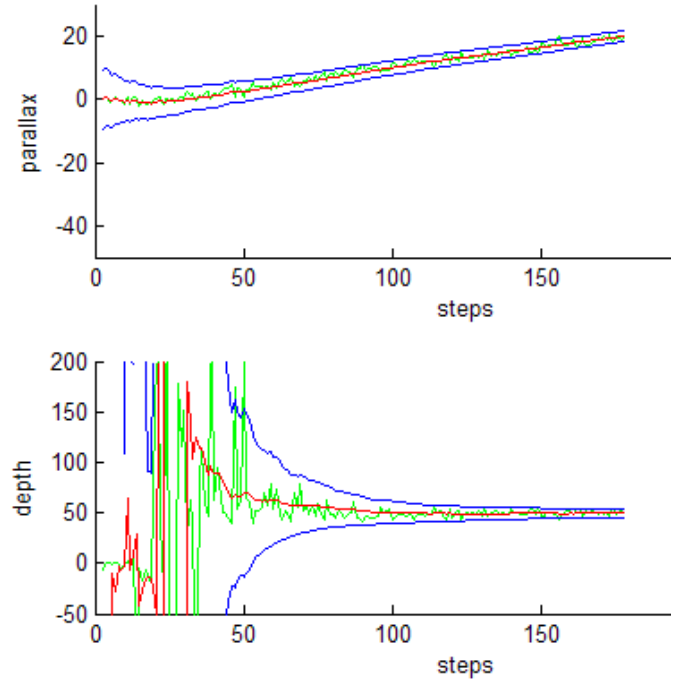


Fig.2. Parallax (Upper plot) and depth (Lower plot) estimate by the Kalman Filter. Note that only a few degrees of parallax are needed in order to minimize the depth uncertainty.

A minimum parallax threshold α_{min} is used for considering a candidate point λ_i as a new feature \hat{y}_i . In experiments we use $\alpha_{min} = 7^\circ$.

Then if $\alpha_i > \alpha_{min}$:

$$\hat{y}_{new} = [x_i, y_i, \theta_i, \rho_i]^T \quad (21)$$

and:

$$\begin{bmatrix} x_i \\ y_i \\ \theta_i \\ \rho_i \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ \theta_1 + z_1 \\ \rho_i' \end{bmatrix} \quad (22)$$

where x_1, y_1, θ_1 and z_1 are obtained directly from the data stored in candidate point λ_i . ρ_i' is taken directly from \hat{x}_{can} .

The covariance matrix P is transformed by the corresponding Jacobian:

$$P_{\text{new}} = J \begin{pmatrix} P & 0 \\ 0 & \sigma_{\rho_i}^2 \end{pmatrix} J' \quad (23)$$

being σ_{ρ_i} the variance of the ρ_i ' and taken directly from P_{can} . When a candidate point λ_i is initialized as \hat{y}_i , then its corresponding values will be removed from the linear Kalman filter responsible for estimating the state \hat{x}_{can} .

IV. EXPERIMENTS

Several simulations have been executed in order to test the performance of the proposed method.

Fig.3 shows both the map and sensor trajectory estimates after a run of 2000 steps of the algorithm. The only information given a priori to the system was the scale reference (the three points in yellow) which was introduced with an associated uncertainty close to zero in the covariance matrix R . Taking into account that there is not an additional sensory input (e.g. odometry), at every step an unknown linear and angular acceleration is introduced with zero mean and known-covariance Gaussian processes (Section A). In simulations $a_x^w = 4\text{m/s}^2$, $a_y^w = 4\text{m/s}^2$ and $a_\theta^w = 2\text{rad/s}^2$ are used. The only sensor input of the system is a noisy sensor capable of measuring the bearing of features, with a limited field of view of 100° (emulating a 2-DOF wide-lens camera). A standard deviation of $\sigma=1^\circ$ is considered for the readings of the sensor.

In the simulation the bearing sensor is moved over a semi-cycled U-like shape trajectory, since our main goal is to observe the effect of the initialization process of new features in the estimation of both map and sensor location, instead of the closing loop problem. About 100 landmarks (in green) simulate the environment of the sensor. Their corresponding feature maps and their uncertainty are indicated in blue. Note the evolution of uncertainty in both sensor and features location. Also note the typical (in SLAM systems) drift in both trajectory and map estimations as the sensor moves far away from its initial location. Fig. 5 shows a close up of the initialization process of three landmarks.

The above experiment was also executed using the ID-Undelayed [10] and the ID-Delayed [11] method. Parameters' initialization for the ID-Undelayed method, values for $\rho_{\text{ini}} = 0.05$ and $\sigma_\rho = 0.025$ were used.

Fig. 4 also illustrates a regular evolution of Euclidean distance (left plot) and orientation (right plot) error, for the three aforementioned methods. The Euclidean distance error is the distance between the real and estimated centers position of the sensor. In the middle plot it can be clearly appreciated that the method proposed in this paper exceeds the performance of both ID-Undelayed and ID-Delayed methods. In the lower plot, it can be seen that the orientation error

tends to be similar for both ID-Delayed and the *Filtered ID Delayed Initialization*, on the other hand oscillates with the ID-Undelayed method.

V. CONCLUSIONS

This work proposes a novel approach called *Filtered ID Delayed Initialization* for initializing new features in SLAM systems based in bearing sensors.

The proposed approach is based in an inverse depth parameterization and delayed initialization scheme. The main idea is to incorporate to the SLAM process, an extra uncorrelated filter, which progressively incorporates the new bearing measurements needed to estimate the full state of each new feature.

The simulations results show that the proposed method can improve the performance of previous approaches, especially in terms of minimize the Euclidean distance error.

ACKNOWLEDGMENT

This research was conducted in the Automatic Control Department, Universitat Politècnica de Catalunya (UPC) and supported by the Spanish Ministry of Science and Innovation, UbROB Project, CICYT number DPI2007-61452.

REFERENCES

- [1] H. Durrant-Whyte, H. Bailey "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine* 2006, Volume 13, pp. 99-16.
- [2] H. Bailey, H. Durrant-Whyte "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine* 2006, Volume 13, pp. 99-16.
- [3] R. Munguía, A. Grau "Single sound source SLAM," *Lecture Notes in Computer Science* 2008, Springer Press vol. 5197, pp 70-77.
- [4] G. Williams, G. Klein, D. Reid "Real-time SLAM relocation," *In Proceedings of the ICCV* 2007.
- [5] D. Chekhlov, M. Pupilli, W. Mayol-Cuevas, A. Calway "Real-time and robust monocular SLAM using predictive multi-resolution descriptors," *In Proc. Int. Symp. Vis. Comput.* 2006..
- [6] R. Munguía, A. Grau "Closing loops with a virtual sensor based on monocular SLAM," *IEEE Transactions on Instrumentation and Measurement* 2009, Volume 58, pp 2377-2385.
- [7] A. Davison "Real-time Simultaneous localization and mapping with a single camera," *In Proceedings of the ICCV* 2003.
- [8] T. Lemaire, S. Lacroix, J. Sola "A practical bearing-only SLAM algorithm." *IEEE International Conference on Intelligent Robots and Systems IROS* 2005.
- [9] E. Eade, T. Drummond "Scalable monocular SLAM," *IEEE Computer Vision and Pattern Recognition CVPR* 2006.
- [10] J.M.M. Montiel, J. Civera, A. Davison "Unified inverse depth parametrization for monocular SLAM," *Robotics: Science and Systems Conference* 2006.
- [11] R. Munguía, A. Grau "Delayed inverse depth monocular SLAM," *17th IFAC World Congress* 2008.
- [12] R. Munguía, A. Grau "Minimizing Drift in Monocular SLAM Real Time Systems" *IEEE International Symposium on Industrial Electronics ISIE*. 2008.

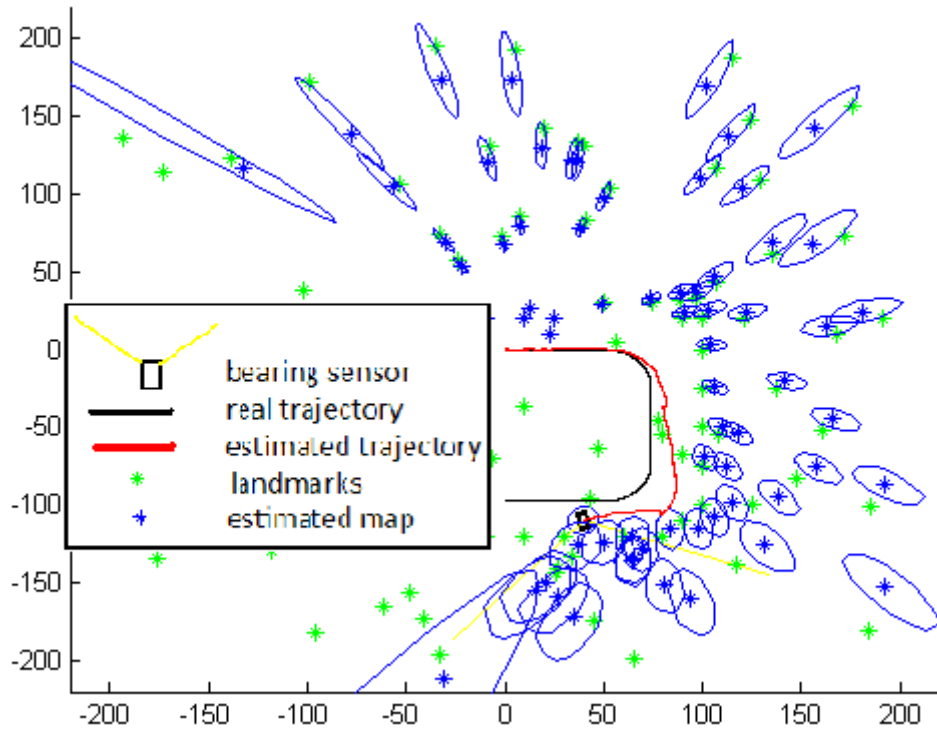


Fig.3 SLAM simulation using the *Filtered ID Delayed Initialization* method (left plot).

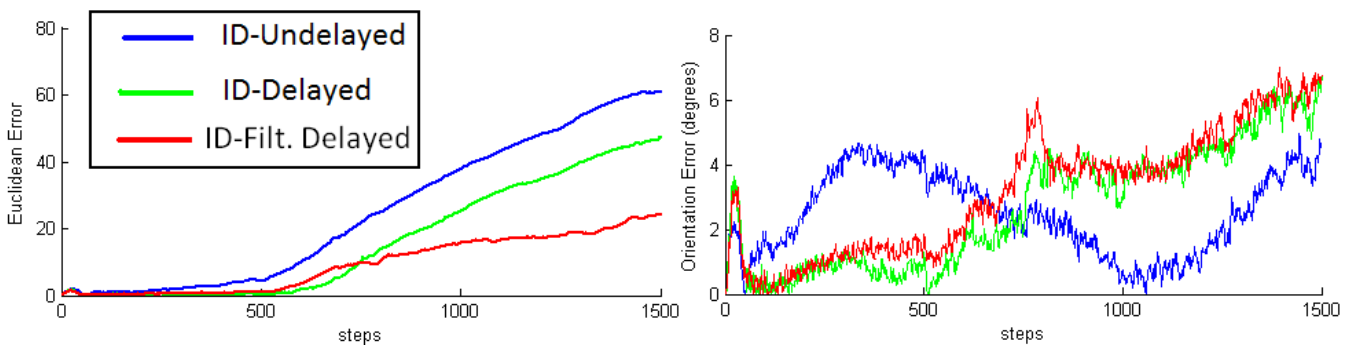


Fig.4 Comparison of the error evolution for ID-Undelayed, ID-Delayed and *Filtered ID Delayed Initialization* (right plot).

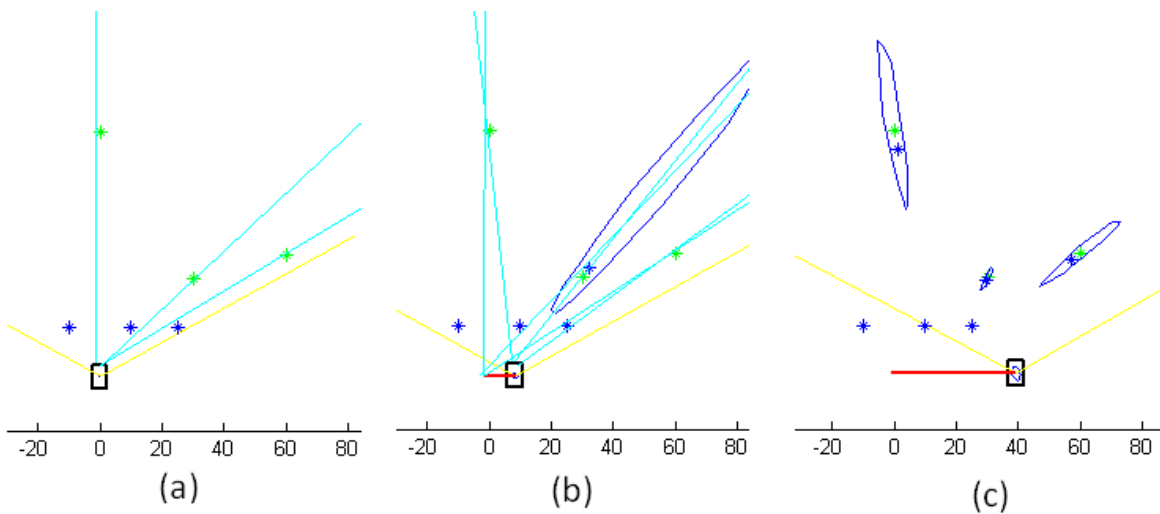


Fig.5. Initialization process of three features maps using the proposed *Filtered ID Delayed Initialization*: At begin of the sequence (plot a) the three landmarks have been detected as candidate points. Note that three feature points have been previously added to the map as a priori known landmarks in order to define-set the scale of the world. Around step 80 (plot b), the small displacement of the sensor to the right produces enough parallax for estimating the depth of the middle landmark; also note that it has been initialized very near to its real position. By the last step, at 400 iterations, (plot c) all the features have been initialized. Note that the uncertainty of the middle feature has been fully minimized.