DIIS - I3A
Universidad de Zaragoza
C/ María de Luna num. 1
E-50018 Zaragoza
Spain

Internal Report: 2008-V09

Parameterization and Initialization of Bearing-Only Information: A Discussion [1]

R. Aragues and C. Sagues

If you want to cite this report, please use the following reference instead:
Parameterization and Initialization of Bearing-Only Information: A Discussion, R. Aragues and C. Sagues, *Fifth International Conference on Informatics in Control, Automation and Robotics (ICINCO 2008), pages 252-261, Madeira - Portugal, May 11-15, 2008.*

[1] This work was supported by projects DPI 2006-07928, IST-1-045062-URUS-STP

PARAMETERIZATION AND INITIALIZATION OF BEARING-ONLY INFORMATION: A DISCUSSION

R. Aragues, C. Sagues
DIIS - I3A, University of Zaragoza
María de Luna, 50018 Zaragoza, Spain
raragues@unizar.es, csagues@unizar.es

Keywords: Bearing-only; Feature parameterization: cartesian, inverse-depth; Delayed/Undelayed initialization.

Abstract: In this paper we discuss feature parameterization and initialization for bearing-only data obtained from vision sensors. The interest of this work refers to the comparison of the bearing-only data representation and initialization techniques. The behavior of the algorithm is analyzed for different robot motions and depth of the features. The results are evaluated in terms of the sensitivity to step size and the performance to ill conditioned situations.
The problem studied refers to robots moving on the plane, sensing the environment and extracting bearing-only information from uncalibrated cameras to recover the position of the landmarks and its own localization.

1 INTRODUCTION

The manipulation of bearing information is an important issue in robotics. Bearing-only data is the kind of information provided by cameras through the projection of landmarks which are in the scene. In order to recover the position of these landmarks in the world, multiple observations taken from different positions must be combined.

Compared with information extracted from other sensors, such as lasers, bearing information is complicated to use. However, the multiple benefits of using cameras have motivated the interest in the researchers. These benefits include the property that cameras are able to sense quite distant features so that the sensing is not restricted to a limited range.

This sensing of the environment in the form of bearing information may be used for many applications such as the computation of the landmark localization in the environment or the calculation of the own robot pose mostly kwnon as SLAM *Simultaneous Localization and Mapping*.

Algorithms which use bearing information must deal with the problem of creating representations for features by the combination of bearing data. The problem of feature parameterization and feature initialization are of big importance here.

With regard to the feature parameterization, the classical approach has been the use of a *cartesian* parameterization (Bailey, 2003), (Kwok and Dissanayake, 2004), (Costa et al., 2004), (Klippenstein et al., 2007). Some approaches prefer a *depth* parameterization, where features are stored as an starting point of the ray where the feature lays, the inclination of the ray and the depth (Davison, 2003). An *inverse-depth* parameterization is an alternative, similar to the *depth* parameterization but using the inverse of the depth instead (Montiel et al., 2006). Some approaches use no explicit feature parameterization and instead represent landmarks as constraints between three robot poses (Trawny and Roumeliotis, 2006).

The problem of depth computation for landmarks is afforded in two separate ways. Some approaches create depth representation from only one bearing assuming an approximate value for it. These techniques are able to cover depths from the position were the landmark was observed until infinity or until a maximum depth within the workspace (Kwok and Dissanayake, 2004), (Davison, 2003), (Montiel et al., 2006). The other approach to depth computation is the combination of observations taken from different robot poses, where triangulation techniques are used to recover the depth (Bailey, 2003), (Klippenstein et al., 2007).

With regard to the feature initialization, *Undelayed* techniques immediately introduce features in the map so that they can be used to improve the robot estimation (Montiel et al., 2006), (Trawny and Roumeliotis, 2006), (Costa et al., 2004), (Kwok and Dissanayake, 2004) while *Delayed* techniques defer the introduction into the map until the features are near-Gaussian (Bailey, 2003), (Klippenstein et al., 2007). *Delayed* techniques often create temporal representations for landmarks which are maintained in separate filters and evolve with the incorporation of new observations of these landmarks until they are finally introduced into the map (Davison, 2003).

The interest of this work refers to the comparison of the bearing-only data representations and initialization techniques, analyzed for different robot motions relative to depth of the landmarks in the scene. Two feature parameterizations are studied. The first is an standard *cartesian* parameterization, where features are described by their (x,y) position. The alternative representation is an adaptation of the *inverse-depth* (Montiel et al., 2006) to the 2D situation. Besides, both *Undelayed* and *Delayed* strategies for feature initialization are used and their performance is compared in different scenarios.

The problem studied in this paper refers to robots moving on the plane, sensing the environment and extracting bearing-only information from uncalibrated images to recover the position of the landmarks and its own localization. As a result of this investigation, some theoretical solutions are proposed, and their validity is supported by an exhaustive experimentation using simulated data. Some preliminary experiments have been carried out using real data from omnidirectional images.

2 BACKGROUND

The problem studied in this paper is related to the use of bearing-only information for the SLAM problem using EKF. The robot moves on the plane and elements in the map are represented by their 2D coordinates. Robot observes landmarks within a field of view of 360° due to the use of omnidirectional cameras and obtains bearing-only measurements. Odometry is used to predict robot motion in every step. The data association problem is not discussed in this paper. The EKF *Extended Kalman Filter* is a widely used technique in these problems and a lot of information can be found in the literature. An innovation test is used to avoid the filter divergence in the presence of poorly initialized features or high innovations. This test computes an individual compatibility for all observation-prediction pairs and then obtains the greatest set of jointly compatible pairs using the JCBB algorithm (Neira and Tardós, 2001).

Along this paper, next notation will be used:

$\mathbf{x} = (\mathbf{x_r}, \mathbf{x_1} ... \mathbf{x_n})$: the state vector containing current robot pose ($\mathbf{x_r}$) and the positions of landmarks ($\mathbf{x_1} ... \mathbf{x_n}$)

$\mathbf{P}$: the covariance matrix.

$\mathbf{x_{r_j}} = (x_{r_j}, y_{r_j}, \theta_{r_j}) \in \mathbb{R}^3$, $\theta_{r_j} \subset [-\pi, \pi]$, for $j = 1..k$: j-th robot pose. When there is no confusion, the subscript j is omitted.

$\mathbf{x_i} = (x_i, y_i) \in \mathbb{R}^2$, for $i = 1..n$: Position of the i-th feature in the map, for *cartesian* parameterization, or $\mathbf{x_i} = (x_i, y_i, \theta_i, \rho_i) \in \mathbb{R}^4$, $\theta_i \subset [-\pi, \pi]$, for $i = 1..n$: when referring to *inverse-depth* parameterization.

z_{ji}: measurement taken from robot pose j to feature i. When only one robot pose is used, z_i refers to the observation of feature i.

$\mathbf{h}$: observation model (EKF).

$\mathbf{g}$: function used to calculate the position of new features when they are introduced into the map.

3 FEATURE PARAMETERIZATION

Cartesian parameterizations represent features by their (x,y) coordinates. This parameterization is very intuitive since the feature position within the map can be easily obtained. The initialization of features in this *cartesian* parameterization is a problematic issue due to the nonlinearity of the triangulation techniques used to recover its position based on the observations taken from different robots poses. It can be easily shown that bearings generate bigger uncertainty as landmark position goes away from the camera. The observation model for a feature $\mathbf{x_i} = (x_i, y_i)$ observed from a robot pose $\mathbf{x_r} = (x_r, y_r, \theta_r)$ is (Bailey, 2003):

$$z_i = h(\mathbf{x_r}, \mathbf{x_i}) = \arctan\left(\frac{y_i - y_r}{x_i - x_r}\right) - \theta_r \qquad (1)$$

Inverse-depth parameterizations represent a feature (x_i) as a ray starting at (x_i, y_i), the position where the feature was firstly observed, with a global bearing θ_i and a depth of $\frac{1}{\rho_i}$. Every feature is stored in the state vector using these four parameters $(x_i, y_i, \theta_i, \rho_i)$. The *cartesian* coordinates of the landmark can be calculated as:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} + \frac{1}{\rho_i}\mathbf{m_i} \qquad (2)$$

where $\mathbf{m_i} = [\cos(\theta_i)\sin(\theta_i)]^T$.

The observation model for a feature $\mathbf{x_i} = (x_i, y_i, \theta_i, \rho_i)$ observed from a robot pose $\mathbf{x_r} = (x_r, y_r, \theta_r)$ is:

$$\mathbf{h} = \mathbf{atan2}(h_y^{xy}, h_x^{xy}) \tag{3}$$

where (h_y^{xy}, h_x^{xy}) are the predictoed coordinates of the feature in the robot reference:

$$\mathbf{h^{xy}} = \begin{pmatrix} h_x^{xy} \\ h_y^{xy} \end{pmatrix} = \mathbf{R_r}\left(\begin{pmatrix} x_i \\ y_i \end{pmatrix} + \frac{1}{\rho_i}\mathbf{m_i} - \begin{pmatrix} x_r \\ y_r \end{pmatrix}\right) \tag{4}$$

where $\mathbf{R_r} = \begin{bmatrix} \cos\theta_r & \sin\theta_r \\ -\sin\theta_r & \cos\theta_r \end{bmatrix}$.

This observation model is not affected if next equation is used instead of equation 4 provided that $\rho_i > 0$:

$$\mathbf{h^{xy}} = \begin{pmatrix} h_x^{xy} \\ h_y^{xy} \end{pmatrix} = \mathbf{R_r}\left(\rho_i\left(\begin{pmatrix} x_i \\ y_i \end{pmatrix} - \begin{pmatrix} x_r \\ y_r \end{pmatrix}\right) + \mathbf{m_i}\right) \tag{5}$$

As advantage with respect to the *cartesian* parametrization, the observation model for an inverse depth parameterization is near linear. Aditionally, landmarks at infinity ($\rho_i = 0$) or uncertainties that extend to infinity can be represented. The main drawback of the inverse-depth parameterization, is that features are over-parameterized, and therefore the Covariance matrix size is greater.

4 FEATURE INITIALIZATION

The feature initialization in SLAM consists in the creation of a representation of the landmark's position and its introduction into the stochastic map through its *mean* and its *covariance matrix*. The feature initialization problem of bearing-only is due to the fact that features are only partially observable.

As told, a measurement only gives information about the direction towards the landmark and two or more observations must be combined in order to recover the depth of the landmark. However, there are some situations where the depth cannot be recovered given the observations of the landmark. Next we give a formal description of these situations.

Theorem 1.- Let us name $\mathbf{x_{r_1}}$ a robot position and $\mathbf{x_{r_2}}$ a second position translated but not rotated with respect to $\mathbf{x_{r_1}}$. Let us name z_{1i} the observation of a feature $\mathbf{x_i}$ taken from $\mathbf{x_{r_1}}$ and z_{2i} the observation of the same feature taken from $\mathbf{x_{r_2}}$. Let us name d_p the translation from $\mathbf{x_{r_1}}$ to $\mathbf{x_{r_2}}$ on a perpendicular direction to z_{1i} and d_t the translation on a parallel direction to z_{1i}. Without loss of generality, let d_t be equal to zero. The landmark depth (distance between $\mathbf{x_{r_1}}$ and the landmark) can be totally determined from $\alpha = z_{1i} - z_{2i}$ as

$$depth = d_p / \tan\alpha \tag{6}$$

Corollary 1.1.- This is an undetermined problem $(0/0)$ when simultaneously $d_p = 0$ and $\alpha = 0 + k\pi$ for $k \in \mathbb{Z}$.

Corollary 1.2.- This problem remains undetermined independently of the magnitude of d_t.

Corollary 1.3.- The landmark is at infinity if simultaneously $\alpha = 0 + k\pi$ for $k \in \mathbb{Z}$ and d_p is different of zero .

Theorem 2.- Let us name $\mathbf{x_{r_1}}$ a robot position and $\mathbf{x_{r_2}}$ a second position rotated but not translated with respect to $\mathbf{x_{r_1}}$. Let us name z_{1i} the observation of a feature $\mathbf{x_i}$ taken from $\mathbf{x_{r_1}}$ and z_{2i} the observation of the same feature taken from $\mathbf{x_{r_2}}$. Robot rotation (θ_{r_2}) can be absolutely determined from $\theta_{r_2} = z_{1i} - z_{2i}$.

Corollary 2.1.- Given a pure rotation motion, feature depth cannot be recovered.

Corollary 2.2.- Given a translation and rotation motion with landmarks of infinite depth, the robot rotation can be computed from $z_{1i} - z_{2i}$ for any $d_p < \infty$ and robot translation cannot be recovered.

Based on these theorems, ill-conditioned situations are identified:

- Features aligned with robot trajectory: Feature depth cannot be recovered. This situations is formalized in Corolaries 1.1 and 1.2.
- Pure rotation motion: Depth cannot be recovered, as shown in Corollary 2.1.
- Landmarks at infinity: When all landmarks are at infinity, robot orientation can be recovered but no translation information can be obtained. This is based on Corollary 2.2.

Feature estimates calculated when the depth computation problem is ill-conditioned present high covariances and great estimation errors which may cause linealization problems. Once a feature has been wrongly initialized, new observations taken from robot poses not aligned with the feature will not be able to correct its position. If a *cartesian* parameterization is used, an additional feature initialization problem is that features with infinite depth cannot be represented and their initialization must be deferred. This situation is formalized in Corollary 1.3.

4.1 Undelayed Initialization

The undelayed initialization consists in the introduction of landmarks into the system the first time the landmark is observed. This technique presents many benefits since the information attached to a landmark can be used earlier and it allows the use of landmarks which may never been initialized if a delayed strategy is used instead. Since the first time a landmark is observed only bearing information is available, undelayed techniques must deal with the problem of creating a representation for the depth and its associated uncertainty.

If an *inverse-depth* parameterization is used, landmarks are introduced using a fixed initial depth and an uncertainty representation is created which covers all depths from some d_{min} to infinity. This initial depth must be adjusted depending on the workspace.

Since *cartesian* parameterization requires low covariances, an *undelayed* initialization is only possible if multiple hypothesis in depth are created (Kwok and Dissanayake, 2004), (Kwok et al., 2007), (Sola et al., 2005). All these approaches present a high complexity and size of the map. Due to this complexity, approaches using *undelayed* initialization together with *cartesian* parameterization are no longer analyzed in this paper.

4.2 Delayed with Two Observations

This delayed technique consists in the combination of the first two observations of a landmark to recover its position using a triangulation algorithm. This is a not purely delayed technique, since there are no conditions which must be satisfied by the observations in order for the landmark to be initialized, and all landmarks are introduced in the map provided that they are observed from at least two robots poses. The main benefit of this initialization strategy is that the solution is independent on the workspace. However, triangulation algorithms used to recover the landmark position are highly non-linear and, depending on the arrangement of robot poses and features, the problem may be *ill-conditioned*.

If a *cartesian* parameterization is used, the recovered feature position must be near-Gaussian and covariances must be small. For this reason, additional tests are used to check that features satisfy these conditions. If features are parameterized using *inverse-depth*, this strategy may suppose a benefit in the sense that it is independent on the workspace while higher covariances in the estimates are admisible and recovered features are near-Gaussian even for low parallaxes.

4.3 Delayed until Condition

In a pure delayed initialization technique, observations to landmarks are accumulated and its initialization is deferred until a condition of Gaussianity is satisfied; then observations are used to create a representation for the feature (Bailey, 2003), (Klippenstein et al., 2007).

If a delayed initialization is used, some landmarks may never been initialized. Since the information provided by landmarks cannot been used until the landmark is initialized, a delayed technique decreases the amount of information available to improve robot the pose. Many delayed techniques present a high computational cost to calculate the condition, and have their own problems and limitations. The main benefit is that the representation for the landmark is more accurate and reliable than the obtained by an undelayed strategy.

5 DISCUSSION

As told, the aim of this work is the comparison of *cartesian* and *inverse-depth* parameterizations combined with *delayed* and *undelayed* initialization techniques. These have been selected because are the most commonly used, being also simple and of low computational complexity. Next a description of the initialization techniques together with the feature parameterization is given.

5.1 Inverse-Depth Undelayed

This technique is an adaptation to the 2D situation of the technique described in (Montiel et al., 2006). A feature $\mathbf{x_i}$ is introduced into the map using a single observation. The current robot pose $\mathbf{x_r} = (x_r, y_r, \theta_r)$ is used together with the observation z_i and an initial depth ρ_0 parameterized in inverse-depth to get the feature representation $\mathbf{x_i}$. This depth is worked out using a minimal distance d_{min} which must be selected depending on the workspace:

$$\rho_{min} \quad = \quad \frac{1}{d_{min}} \quad \rho_0 \quad = \quad \frac{\rho_{min}}{2} \quad \sigma_\rho \quad = \quad \frac{\rho_{min}}{4} \tag{7}$$

where ρ_{min} is the inverse of depth, ρ_0 is the initial inverse-depth, which is the middle value of the interval $[0, \rho_{min}]$, and σ_ρ is the standard deviation used to initialize ρ_0 (95% of ρ is in the interval $\left[\rho_0 - 2\sigma_\rho, \rho_0 + 2\sigma_\rho\right] = [0, \rho_{min}]$.) The initial value of the feature is calculated as:

$$\mathbf{x_i} = \mathbf{g}(\mathbf{x_r}, z_i, \rho_0) = (x_r, y_r, \theta_r + z_i, \rho_0) \tag{8}$$

5.2 Inverse-Depth Delayed with Two Observations

As a proposal, an *inverse-depth* parameterization (Montiel et al., 2006) is combined with a delayed initialization technique where the second observation is used to calculate the initial depth for the feature. The position for the feature $\mathbf{x_i}$ which has been observed from $\mathbf{x}_{r_1} = (x_{r_1}, y_{r_1}, \theta_{r_1})$ and $\mathbf{x}_{r_2} = (x_{r_2}, y_{r_2}, \theta_{r_2})$ producing measurements z_{1i} and z_{2i} is calculated as follows:

$$\begin{array}{rcl} \mathbf{x_i} & = & \mathbf{g}(\mathbf{x_{r_1}}, \mathbf{x_{r_2}} z_{1i}, z_{2i}) = (x_{r_2}, y_{r_2}, \theta_{r_2} + z_{2i}, \rho_0) \\ \rho_0 & = & \frac{s_2 * c_1 - c_2 * s_1}{(y_{r_1} - y_{r_2}) * c_1 - (x_{r_1} - x_{r_2}) * s_1} \end{array} \tag{9}$$

where $c_j = \cos(\theta_j + z_{ji})$ and $s_j = \sin(\theta_j + z_{ji})$, for $j = 1, 2$.

An additional check is used in order to detect situations where inverse-depth cannot be recovered and intersections take place in the opposite direction of the observation. In these situations, the initialization is deferred.

5.3 Cartesian Delayed with Two Observations

Given the first two observations z_{1i}, z_{2i} of a landmark $\mathbf{x_i}$ taken from robot poses $\mathbf{x_{r_1}}, \mathbf{x_{r_2}}$, the landmark position $\mathbf{x_i} = (x_i, y_i)$ is calculated as follows (Bailey, 2003):

$$\begin{array}{l} x_i = \mathbf{g_1}(\mathbf{x_{r_1}}, \mathbf{x_{r_2}}, z_{1i}, z_{2i}) = \frac{x_{r_1} s_1 c_2 - x_{r_2} s_2 c_1 - (y_{r_1} - y_{r_2}) c_1 c_2}{s_1 c_2 - s_2 c_1} \\ y_i = \mathbf{g_2}(\mathbf{x_{r_1}}, \mathbf{x_{r_2}}, z_{1i}, z_{2i}) = \frac{y_{r_2} s_1 c_2 - y_{r_1} s_2 c_1 + (x_{r_1} - x_{r_2}) s_1 s_2}{s_1 c_2 - s_2 c_1} \end{array} \tag{10}$$

where $c_j = \cos(\theta_j + z_{ji})$ and $s_j = \sin(\theta_j + z_{ji})$, for $j = 1, 2$.

Similarly a test is used to check that features can be recovered and intersections of bearings are not in the opposite direction of the observations.

5.4 Cartesian/Inverse-Depth Delayed until Finite Depth

A *delayed* technique is proposed where feature initialization is deferred until finite uncertainty in depth can be estimated. This is achieved by a simple test which compares two observation rays and checks if they are parallel. This situation is characterized by Corollaries 1.1 and 1.3. When observation rays are parallel, the uncertainty in depth of the recovered landmark extends to infinity and the initialization is deferred. This test is especially useful when a *cartesian* parameterization is used, since infinite depths cannot been modeled.

Let $\mathbf{x_{r_j}} = (x_{r_j}, y_{r_j}, \theta_{r_j})$, for $j = 1, 2$ be the two robot poses where observations z_{ji}, for $j = 1, 2$ to a landmark $\mathbf{x_i}$ were taken. Global bearings α_{ji}, for $j = 1, 2$ to the landmark are calculated as:

$$\alpha_{ji} = \theta_{r_j} + z_{ji} \tag{11}$$

If we name $S_{\alpha_{ji}}$ the linearized propagated covariance for bearing α_{ji} then the Chi-squared test for Finite Depth is expressed as:

$$\frac{(\alpha_{1i} - \alpha_{2i})^2}{S_{\alpha 1i} + S_{\alpha_{2i}}} > \chi^2_{0.99, 1d.o.f} \tag{12}$$

5.5 Cartesian/Inverse-Depth Delayed until Feature Not Aligned with Robot Poses

The initialization of features aligned with the robot trajectory is a problematic issue when working with bearing-only data. When a feature is observed from two robot poses which are in line with the feature, it is not possible to make a right depth initialization. Corollary 1.1. gives a formal explanation of this situation: feature is aligned with robot trajectory when the observation rays are parallel and the robot translation takes place in a direction which is parallel to the observation.

Let $\mathbf{x_{r_j}} = (x_{r_j}, y_{r_j}, \theta_{r_j})$, for $j = 1, 2$ be the two robot poses where observations to a landmark $\mathbf{x_i}$ were taken. From here α_{1i}, α_{2i}, for $j = 1, 2$ can be computed with equation 11. Let $S_{\alpha_{ji}}$, for $j = 1, 2$ be their linearized propagated covariances. Observation rays are parallel when:

$$\frac{(\alpha_{1i} - \alpha_{2i})^2}{S_{\alpha 1i} + S_{\alpha_{2i}}} \leq \chi^2_{0.99, 1d.o.f} \tag{13}$$

The robot trajectory from $\mathbf{x_{r_1}}$ to $\mathbf{x_{r_2}}$ has a global inclination which can be calculated as:

$$\theta_t = \arctan\left(\frac{y_{r_2} - y_{r_1}}{x_{r_2} - x_{r_1}}\right) \tag{14}$$

Let S_{θ_t} be the linearized propagated covariance for bearing θ_t and $S_{\alpha_{ji}}$, for $j = 1, 2$ the covariances of the ray inclinations α_{1i} and α_{2i}. The trajectory is parallel to the observation rays when:

$$\frac{(\theta_t - \alpha_{ji})^2}{S_{\theta t} + S_{\alpha_{ji}}} \leq \chi^2_{0.99, 1d.o.f} \tag{15}$$

for $j = 1, 2$.

The initialization of features is deferred until a pair of observations is available where the feature is not aligned with the trajectory. This delayed technique is less restrictive than the explained in section 5.4 and is specially useful for an inverse-depth parameterization since it allows the initialization and the use of features of inifinite depth.

6 EXPERIMENTS

In order to analyze the performance of the different parameterizations and initialization techniques, some experiments have been designed so that the performance and robustness of the algorithms to deal with bearing-only data can be analyzed.

The experimentation and result analysis is carried out using a simulator which presents many benefits. First of all, **exactly the same** experiment can be solved by several algorithms so that results are fully comparable. Besides, ground truth information is available and therefore obtained results can be compared with the truth situation.

Some preliminary experiments have been carried out using omnidirectional images which can be seen in Figure 1. The matches have been obtained using SURF descriptors (Murillo et al., 2007).

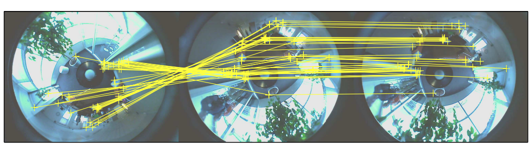

Figure 1: Omnidirectional image: feature extraction and matching.

In the simulated experiments, an observation noise with an standard deviation of 0.125 degrees is used. Features are placed on the walls of a squared room. An initialization to the system is introduced from three robot poses and the first 5 observed landmarks. It is based on SFM techniques with the Trifocal Tensor (Sagüés et al., 2006). The data association problem is not discussed in this paper and data association is supposed to be perfect.

Algorithms have been tested in different scenarios and under different conditions of *visibility*, *trajectory* and *step sizes*. The **Sensor Visibility** affects to the number of visible landmarks. Two possibilities are evaluated: **Total**, where all features are visible from all robot poses and **Section**, where the workspace is divided into four sections and, in every step, robot observes the features within its section and a few from the neighborhood in order to connect the sections. When the visibility is *Total*, no loop closing takes place and distant features are used.

As stated in section 4 the **Robot Trajectory** has a big influence on depth computation in such a way that if landmark is on the direction of robot translation, depth computation is an undetermined problem. Two trajectories have been evaluated. The first is an **Squared** trajectory composed by several pure translation motions and four 90° pure rotations. In this trajectory some features are aligned with the robot movement for many steps. The odometry noise is introduced as a function of the step size (st) and it can be seen in Table 1, columns *Pure translation* and *Pure rotation*. The second trajectory is **Circular**. Robot describes a circumference when moving along the environment which supposes mixed rotations and translations. No feature in the map is observed in line with the trajectory although, for small step sizes, features may seem to be in the direction of the robot translation. The standard deviations of the odometry noise are shown in Table 1, column *Mixed motion*.

Table 1: Odometry noise relative to the step size (st).

Standard deviation	Pure translation	Pure rotation	Mixed motion
x_r	$0.01 * st$	$0.03 * st$	$0.03 * st$
y_r	$0.01 * st$	$0.03 * st$	$0.03 * st$
θ_r	2°	2.5°	2.5°

The **Step Size** determines the distance (in meters) between two consecutive robot poses. This is the parameter which affects the most the behavior of algorithms. Step sizes of 0.125 m, 0.250 m, 0.5 m and 1 m are tested.

6.1 Analyzed Information

The variables used in order to analyze the performance of an algorithm are listed below.

Final divergence: Percent of results where the final robot pose diverges from its estimation. The condition which is tested for each component (x_r, y_r, θ_r) independently can be written as

$$\frac{(a-\widehat{a})^2}{P} \leq \chi^2_{0.99,1} \tag{16}$$

a being (x_r, y_r, θ_r) the ground-truth, $\widehat{a}$ the estimated value for variable a and P its estimated covariance.

Map consistency: Percent of features in the final map whose estimation is *consistent* with the ground truth. A feature is considered *i*nconsistent if a great estimation error takes place in its x_i or y_i coordinate:

$$\frac{|a-\widehat{a}|}{+\sqrt{P\,\chi^2_{0.99,1}}} \leq 1.5 \tag{17}$$

where the variable a represents the x_i or y_i coordinates.

Trajectory divergence: Percent of steps in the trajectory where the estimation of the robot pose (x_r, y_r, θ_r) diverges.

Feature initialization step: Average of the number of steps needed to initialize a feature, calculated as the difference between the step when a feature is first observed and the one when the feature is introduced into the map.

Feature usage: Average of the feature usage per step: percent of features used in the filter update versus the number of features observed.

Map consistency per step: Average of the percent of consistent features in the map in every step.

Additionaly, information related to the precision and error of the final robot pose, the trajectory and the final map has been also studied.

6.2 Results

A total of 160 experiments have been designed, and all of them have been solved using the available algorithms discussed in section 5.

For the Inverse-depth undelayed, a minimal depth $d_{min} = 0.5m$ is used.

The results are analyzed in three different blocks. In the first we compare the *cartesian delayed* algorithms. In the second, we compare all *inverse-depth delayed* approaches and in the third block, a general comparison is carried out where the best of the *cartesian delayed* algorithms and the *inverse-detph delayed* method which performs better are compared to the *inverse-depth undelayed* algorithm.

6.2.1 Cartesian Delayed Comparison

The results obtained by the *cartesian delayed* algorithms can be found in Figure 2. The cartesian delayed until finite depth (**xy-f**) algorithm performs better than the delayed with two observations (**xy-d**) and the delayed until features not aligned (**xy-l**) methods: the final divergence (Figure 2.a) and trajectory divergence (Figure 2.c) are the lowest for all step sizes, the map consistency (Figure 2.b, Figure 2.f) are the highest, and it the number of features used to update (Figure 2.e) is higher than the used by the other cartesian algorithms for all step sizes even though this algorithm needs more steps to initialize a feature (Figure 2.d).

6.2.2 Inverse-depth Delayed Comparison

From the study of the results obtained by the inverse-depth delayed algorithms, we can observe that all algorithms performed in a very similar way (Figure 3). The final divergence (Figure 3.a), map consistency (Figure 3.b), trajectory divergence (Figure 3.c), feature usage (Figure 3.e) and map consistency per step (Figure 3.f) results are similar for all inverse-depth delayed algorithms for the different step sizes. Only the feature initialization step (Figure 3.d) differs, due to the use of the different delayed strategies.

An especial study is carried out in order to compare the capability of the inverse-depth algorithms to deal with features which are observed during many steps aligned with the trajectory. The most critical situation is when the robot moves following an squared trajectory and only observes landmarks within its section. In this situation the problematic features are F12, F23, and F34 (Figure 4). In this figure, the ground-truth robot trajectory and landmark positions are displayed in red, while the estimates and uncertainties calculated by the algorithms are drawn in blue. As can be observed, both the trajectory and the landmark positions have been correctly estimated in all cases. However, features F12, F23 and F34 present high uncertainty Figure 4.a) when the algorithm used is the *inverse-depth with two observations* (**id-d**).

Paying attention to the problematic features (F12, F23, F34) in Figure 4 we can observe the results of an earlier initialization of features which are in line with the trajectory. Even though their initial estimate and covariance are correctly represent the feature position, posterior observations are not able to correct its position due to the huge innovation. The *Inverse-depth delayed until finite depth* (**id-f**) and *Inverse-depth delayed until feature not aligned with robot poses* (**id-l**) performed in a similar way. However, the second is prefered because of its capability to initialize and use features of infinite depth.

6.2.3 Global Comparison

The three algorithms *inverse-depth undelayed* with $d_{min} = 0.5m$, *inverse-depth delayed until feature not aligned with robot poses*, and *cartesian delayed until finite depth* are globally compared and their results are analyzed in order to find the one which performs better.

As can be observed in Figure 5, the behavior of the inverse-depth undelayed algorithm (**id-u**) is seriously affected by the step size. For the smallest step size (0.125m), almost all experiments converged in the last robot pose (Figure 5.a) while for the other step sizes, many experiments diverged. The number of consistent features in the final map (Figure 5.b) is lower than for the other algorithms. This behavior is also observed for the number of consistent features per step (Figure 5.f).

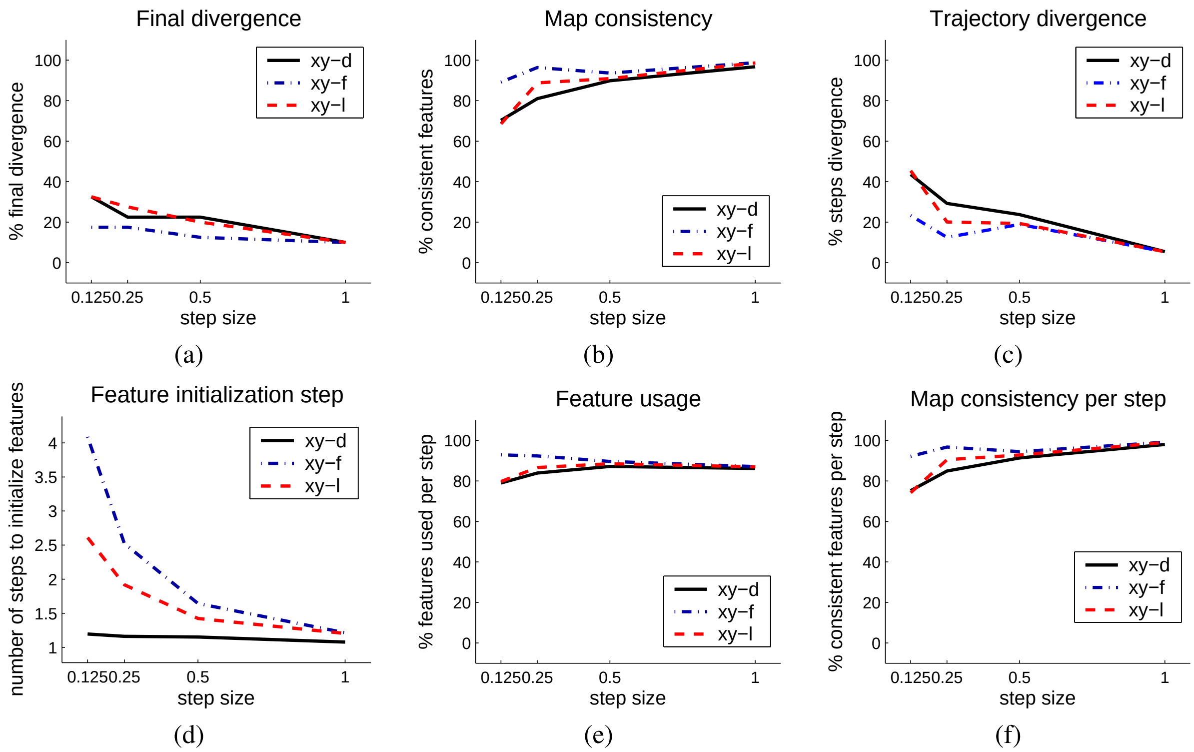

Figure 2: Cartesian delayed techniques comparison. Analysis of the results for different step sizes (x-axis). The algorithms used are *cartesian delayed*. **xy-d**: *with two observations*. **xy-f**: *until finite depth*. **xy-l**: *until feature not aligned with robot poses*.

The cartesian delayed until finite depth algorithm (**xy-f**), its behavior is not so much affected by the step size but we can observe a better performance when the step size increases: the final divergence (Figure 5.a) is slightly higher for smaller step sizes. The number of consistent features in the final map (Figure 5.b) and along the steps (Figure 5.f) slightly decreases for smaller step sizes. The feature usage (Figure 5.e) remains high for all step sizes.

The inverse-depth delayed until features not aligned algorithm (**id-l**) produced the best results, exhibiting an stable behavior for all step sizes: almost all experiments converged (Figure 5.a) and also along the trajectory (Figure 5.c), almost all features are consistent in the final map (Figure 5.b) and along the steps (Figure 5.f), and the feature usage is the highest (Figure 5.e).

An interesting information about the features usage can be extracted from Figure 5.d and Figure 5.e: it can be observed that when an undelayed strategy is selected, the percent of features used to update the map in every step (Figure 5.e) is much lower than the used by the delayed algorithms even though features initialization requires a lower number of steps 5.d. Therefore, delayed techniques provide important benefits due to the fact that the initial estimates introduced into the map are better with lower covariance.

7 Conclusions

In this paper we have discussed feature parameterization and initialization using bearing-only measurements. Both considerably affect the results of the algorithms. However, this paper shows that even with a perfect feature parameterization, if the initialization problem is ill-conditioned the results are inconsistent. As conclusion we can state that in general situations the delayed inverse depth until features not aligned performs competitively.

An interesting result of this study is the related to the cartesian parameterization when it is combined with a finite depth test. It was expected that cartesian algorithm based in triangulation techniques were to suffer a great degradation of their performance for small step sizes. However, results show that the algorithm delayed until finite depth with cartesian parameterization is not very sensitive to the step size and

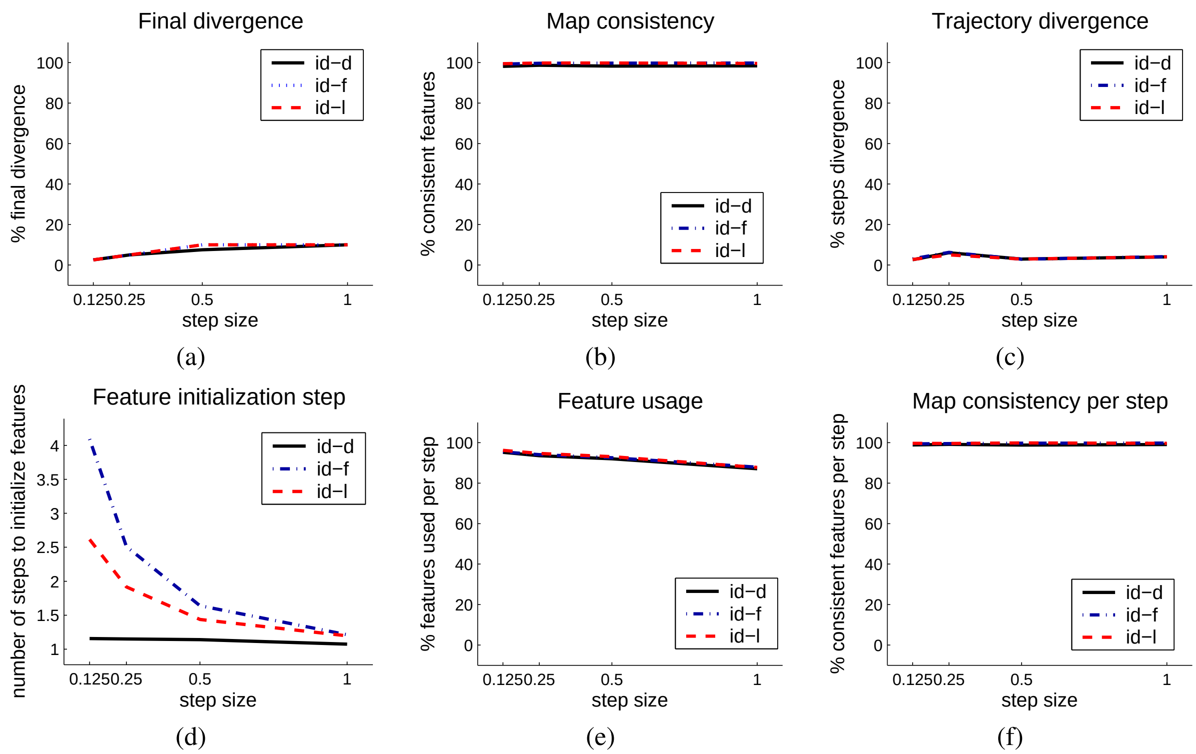

Figure 3: Inverse-depth delayed comparison. Analysis of the results for different step sizes (x-axis). The algorithms are: **id-d**: *with two observations*. **id-f**: *until finite depth*. **id-l**: *until feature not aligned with robot poses*.

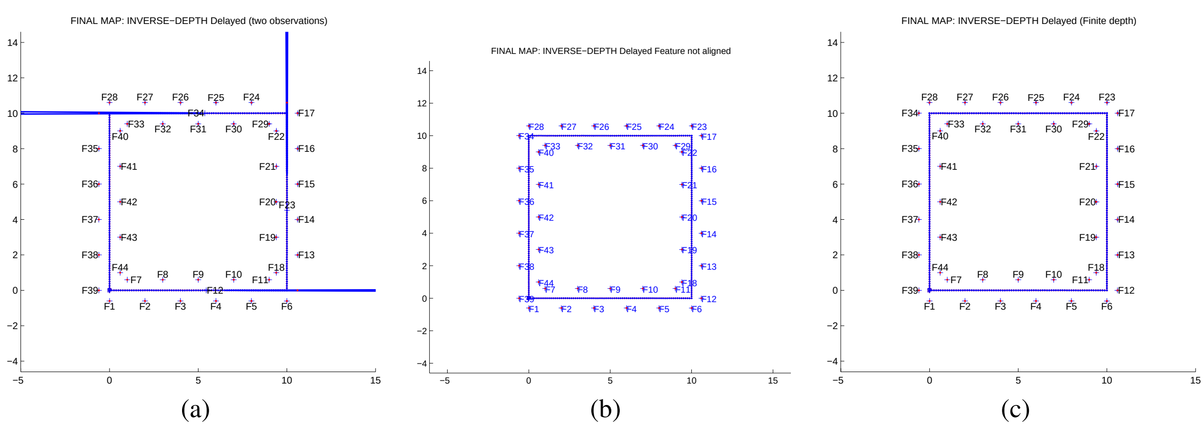

Figure 4: (a) Inverse-depth delayed with two observations. (b) Inverse-depth delayed until feature not aligned with robot poses. (c) Inverse-depth delayed until finite depth.

exhibits very competitive results. Other interesting conclusion is that introducing features earlier in the EKF does not mean that more/better information will be available to update the state.

In this paper we have also stated ill-conditioned situations: a pure rotation motion and features aligned with the trajectory. None of them can be managed in any case. Some ideas have been presented to detect these situations which will allow the algorithms to decide which data can be used in each instant.

REFERENCES

Bailey, T. (2003). Constrained initialisation for bearing-only slam. In *Proc. IEEE Intl. Conf. on Robotics and*

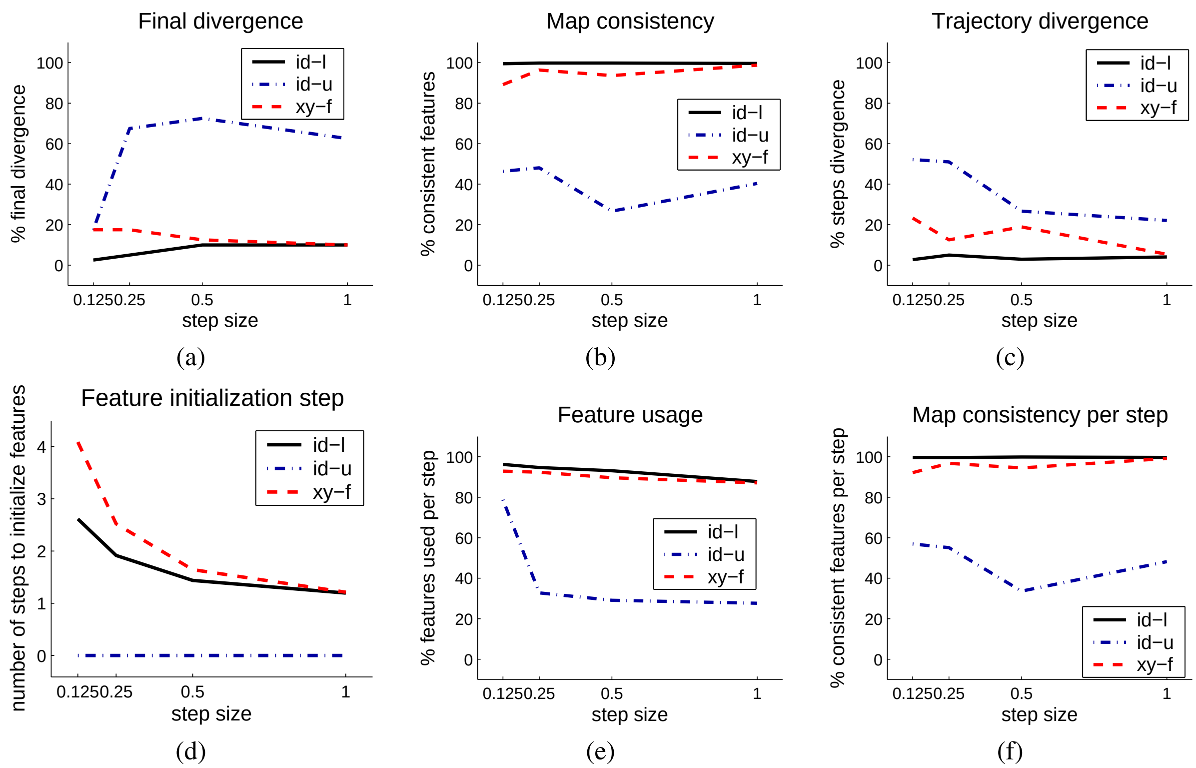

Figure 5: Global comparison. Analysis of the results for different step sizes (x-axis). The algorithms used are: **id-u**: *inverse depth undelayed*, $d_{min} = 0.5m$. **xy-f**: *cartesian delayed until finite depth*. **id-l**: *inverse depth delayed until feature not aligned with robot poses.*

Automation, pages 1996–1971, Taipei, Taiwan.

Costa, A., Kantor, G., and Choset, H. (2004). Bearing-only landmark initialization with unknown data association. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 1164–1770.

Davison, A. (2003). Real-time simultaneous localisation and mapping with a single camera. In *Proc. Ninth IEEE Intl.Conf. on Computer Vision*.

Klippenstein, J., Zhang, H., and Wang, X. (2007). Feature initialization for bearing-only visual slam using triangulation and the unscented transform. In *Proc. IEEE Intl Conf. on Mechatronics and Automation*, pages 157–164, Harbin, China.

Kwok, N. and Dissanayake, G. (2004). An efficient multiple hypothesis filter for bearing-only slam. In *Proc. IEEE/RSJ Intl. Conf. on Robotics and Systems*, volume 1, pages 736–741, Sendai, Japan.

Kwok, N. M., Ha, Q. P., Huang, S., Dissanayake, G., and Fang, G. (2007). Mobile robot localization and mapping using a gaussian sum filter. *Intl.Journal of Control, Automation, and Systems*, 5(3):251–268.

Montiel, J. M. M., Civera, J., and Davison, J. (2006). Unified inverse depth parametrization for monocular slam. In *Robotics Science and Systems, RSS*, Philadelphia, Pennsylvania.

Murillo, A. C., Guerrero, J. J., and Sagüés, C. (2007). Surf features for efficient robot localization with omnidirectional images. In *IEEE/RSJ Int. Conf. on Robotics and Automation*, pages 3901–3907.

Neira, J. and Tardós, J. (2001). Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890897.

Sagüés, C., Murillo, A. C., Guerrero, J. J., Goedemé, T., Tuytelaars, T., and Gool, L. V. (2006). Localization with omnidirectional images using the 1d radial trifocal tensor. In *Proc of the IEEE Int. Conf. on Robotics and Automation*, pages 551–556.

Sola, J., Monin, A., Devy, M., and Lemaire, T. (2005). Undelayed initialization in bearing only slam. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)*.

Trawny, N. and Roumeliotis, S. (2006). A unified framework for nearby and distant landmarks in bearing-only slam. In *Proc. IEEE Intl. Conf. on Robotics and Automation*.