The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

EKF-SLAM for AUV navigation under Probabilistic Sonar Scan-Matching

Angelos Mallios, Pere Ridao, David Ribas, Francesco Maurelli and Yvan Petillot

Abstract— **This paper proposes a pose-based algorithm to solve the full Simultaneous Localization And Mapping (SLAM) problem for an Autonomous Underwater Vehicle (AUV), navigating in an unknown and possibly unstructured environment. A probabilistic scan matching technique using range scans gathered from a Mechanical Scanning Imaging Sonar (MSIS) is used together with the robot dead-reckoning displacements. The proposed method utilizes two Extended Kalman Filters (EKFs). The first, estimates the local path traveled by the robot while forming the scan as well as its uncertainty, providing position estimates for correcting the distortions that the vehicle motion produces in the acoustic images. The second is an augmented state EKF that estimates and keeps the registered scans poses. The raw data from the sensors are processed and fused in-line. No priory structural information or initial pose are considered. Also, a method of estimating the uncertainty of the scan matching estimation is provided. The algorithm has been tested on an AUV guided along a 600 m path within a marina environment, showing the viability of the proposed approach.**

I. INTRODUCTION

During a long term mission with an Autonomous Underwater Vehicle (AUV) it is necessary to keep the track of the vehicle's position. The last decade, a number of studies in mobile robotics had developed techniques to address the localization problem with very promising results. In particular, the so-called Simultaneous Localization And Mapping (SLAM) techniques have been broadly and successfully applied to indoor and outdoor environments. Hence, it is of interest to study how to adapt these techniques in the hostile underwater environments.

This paper is a contribution in this area, proposing a pose-based algorithm to solve the full SLAM problem of an AUV navigating in an unknown and possibly unstructured environment. The technique incorporates probabilistic scan matching with range scans gathered from a Mechanical Scanning Imaging Sonar (MSIS), taking into account the robot dead-reckoning displacements estimated from a Doppler Velocity Logger (DVL) and a Motion Reference Unit (MRU). Scan matching is a technique that estimates the robot relative displacement between two configurations, by maximizing the overlap between the range scans normally gathered with a laser or a sonar sensor [1]. Due to the rapidly attenuation of the high frequencies in water, the use of high resolution devices like laser scanners it becomes impractical, so normally low frequency sonars are utilized. Acoustic sonar frequencies can penetrate deep the water (e.g. 10 - 150 m for a forward looking sonar), they are not subject of the water visibility but provide limited information and medium to low resolution and refresh rate.

Although a large literature exists reporting successful applications of scan matching and SLAM to mobile robots, very few attempts have been done to use sonar scan matching in underwater applications and even fewer putting them in a SLAM framework. In [2] a non-probabilistic variation of Iterative Closest Point (ICP) is proposed to achieve on-line performance for registering multiple views captured with a 3D acoustic camera. Silver et al. [3], proposed to use a particle filter to deal with the sonar noisy data but only simulated results are reported whilst in [4], sonar scans are registered with FastSLAM and occupancy grids in order to map flooded ancient cisterns. An application combining SLAM and sonar scan matching underwater is reported in [5] were an ICP variant is used for registering bathymetric sub-maps gathered with a multibeam sonar profiler. With the same type of sensor, [6] modeled the uncertainty in the vehicle state using a particle filter and an Extended Kalman Filter (EKF). In [7], a real time SLAM is used with an AUV the explores underwater caves and tunnels, in a 3D environment. Their method consists of a Rao-Blackwellized particle filter with a 3D evidence grid map representation. The Mechanical Scanning Imaging Sonar Probabilistic Iterative Correspondence (MSISpIC) algorithm proposed in [8], is dealing with data gathered by an AUV utilizing MSIS. Is based on the Probabilistic Iterative Correspondence (pIC) algorithm [9] but taking into account the distortions in the acoustic image due to vehicles' motion. In order to deal with them, an EKF using a constant velocity model with acceleration noise updated with velocity measurements from a DVL and attitude measurements from MRU, is used to estimate the trajectory followed by the robot along the scan. This trajectory is used to remove the motion induced distortion on the acoustic image as well as to predict the uncertainty of the range scans prior to register them through the modified pIC algorithm.

In this paper we extend the MSISpIC algorithm in the pose-based SLAM framework. Now, each new pose of a scan is maintained in a second Augmented State EKF (ASEKF) and is compared with previous scans that are in the nearby area. If there is enough data overlapping, a new scan match will put a constraint between the poses updating the ASEKF.

This research was sponsored by the Marie Curie Research Training Network FREEsubNET (MRTN-CT-2006-036186)

A. Mallios, P. Ridao and D. Ribas are with the Institute of Informatics and Applications, Universitat de Girona, Spain {amallios; pere; dribas}@eia.udg.edu

F. Maurelli and Y. Petillot are with the Ocean Systems Laboratory, Heriot-Watt University, Edinburgh, Scotland, UK, {f.maurelli; Y.R.Petillot}@hw.ac.uk

These constraints help to identify and close the loops which correct the entire previous trajectory, bounding the drift. The proposed method has been tested with a real world dataset, including Differential Global Positioning System (DGPS) for ground truth, acquired during an experimental survey in an abandoned marina located in the Girona coast. The results show substantial improvements in trajectory correction and map reconstruction.

The main principles of pose-based SLAM can be rooted back to [10] where the authors proposed a nonlinear optimization technique to estimate the robot path, in consistency with relative pose constrains established through odometry and scan matching. In [11] observed that the information form formulation of the feature-based SLAM posterior was almost diagonal and proposed the Sparse Extended Information Filter (SEIF) to force the sparseness of the Information matrix through an approximation called sparsification. More over, [12] showed that the information form of the pose-based SLAM (referred by them as delayed state filter) is exactly sparse. With the aim of reducing as much as possible the size of information vector, as well as to delay inconsistency [13] have recently proposed methods to keep only high informative loop-closure links as well as non redundant poses within the information vector. Because the focus of our work consists in applying probabilistic scan matching techniques on sparse sonar data to perform underwater SLAM, having access to the uncertainties of the robot poses plays an important role. Hence, even though that there exist more computationally efficient formulations of the pose-based SLAM [14], [15], [16], [17], the EKF form is preferred here given the size of the validating experiment.

The paper is structured as follows. In section II the probabilistic scan matching algorithm is described. Section III details the MSISpIC following by its uncertainty estimation in section IV. Both are essential parts of the proposed SLAM algorithm which is described in section V. Section VI reports the experimental results before conclusions.

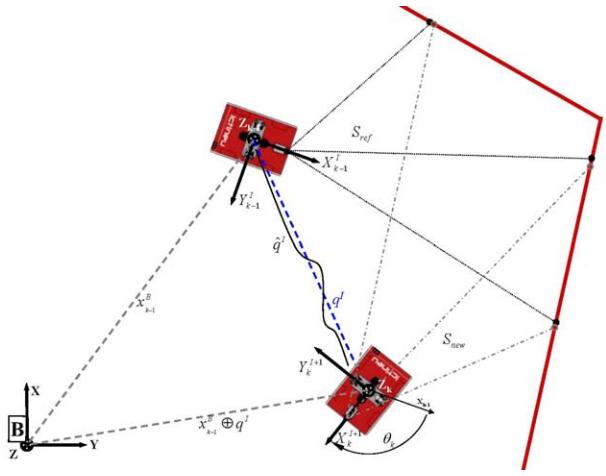


Fig. 1. Scan matching problem description.

II. PROBABILISTIC SCAN MATCHING

The goal of scan matching is to compute the relative displacement of a vehicle between two consecutive configurations by maximizing the overlap between the range measurements obtained from a laser or a sonar sensor. That means, that given a reference scan S_{ref}, a new scan S_{new} and an rough displacement estimation $\mathbf{q}_0$ between them, the objective of scan matching methods is to obtain a better estimation of the real displacement $\mathbf{q} = (x, y, \theta)$ (Fig. 1).

Several scan matching algorithms exists with most of them being variations of the ICP algorithm. The geometric representation of a scan in the conventional ICP algorithm does not model the uncertainty of the sensor measurements. Correspondences between two scans are chosen based on the closest-point rule normally using the Euclidean distance. As pointed out in [9], this distance does not take into account that the points in the new scan, which are far from the sensor, could be far from their correspondents in the previous scan. On the other hand, if the scan data are very noisy, two statistically compatible points could appear far enough, in terms of the Euclidean distance. Both situations might prevent a possible association or even generate a wrong one. Probabilistic scan matching algorithms are statistical extensions of the ICP algorithm where the relative displacement $\mathbf{q}_0$ as well as the observed points in both scans $\mathbf{r_i}$ and $\mathbf{n_i}$, are modeled as Random Gaussian Variables (RGV).

III. MSISPIC ALGORITHM

Scan matching techniques are conceived to accept as input parameters two range scans with a rough displacement estimation between them. Most of the approaches utilize laser technology sensors, which beside their high accuracy, they also have two more major advantages: they gather each scan almost instantaneously and the beams angle can be consider almost perfect. However, when using ultrasonic range finders these advantages are no longer valid because of their lower angular resolution and the sparsity of the readings. To overcome this problem the Sonar Probabilistic Iterative Correspondence (spIC) is proposed in [18], as an extension of the pIC. Further than mobile robotics, as the light propagation in the water is very poor, AUVs utilized acoustic sonars instead of laser sensors. In [8] the MSISpIC algorithm is proposed, modifying the pIC in order to deal with the strong deformations that, by its nature, are introduced in the scan whilst is formed in the underwater domain.

Commercially available underwater scan sensors are based on acoustics with a mechanical head that rotates at fixed angular steps. At each step, a beam is emitted and received, measuring ranges and intensities to the obstacles found across its trajectory. Acquiring a complete scan can take several seconds, therefor it can be distorted from the vehicle's motion. Thus, getting a complete scan that lasts few seconds while the vehicle is moving, generating deformed acoustic images. For this reason, it is necessary taking into account the robot pose when the beam was grabbed.

One part of the MSISpIC algorithm forms a scan, corrected from the vehicle's motion distortion. MSISpIC uses an

EKF with a constant velocity model and acceleration noise for the prediction step. Then, it updates with velocity and attitude measurements obtained from a DVL and a MRU respectively, in order to estimate the trajectory followed by the robot along the scan. This trajectory is used to remove the motion induced distortion of the acoustic image as well as to predict the uncertainty of the range scans prior to register them. After the corrected scan has formed, MSISpIC grabs two scans and registers them using a modified pIC algorithm. MSISpIC algorithm, consist of three major parts: Beam segmentation, Relative vehicle localization and Scan forming.

A. Beam Segmentation and Range Detection

The MSIS returns a polar acoustic image composed of beams. Each beam has a particular bearing angle value and a set of intensity measurements. The angle corresponds to the orientation of the sensor head when the beam was emitted. The acoustic linear image corresponding to one beam is returned as an array of acoustic intensities detected at a certain distance. The beam is then segmented using a predefined threshold to compute the intensity peaks. Due to the noisy nature of the acoustic data, a minimum distance between peaks criteria is also applied. Hence, positions finally considered are those corresponding to high intensity values above the threshold with a minimum distance between each other.

B. Relative Vehicle Localization

To maximize the probability for data overlapping, we collect a complete 360^o scan sector and register it with the previous one in order to estimate the robots' displacement. From now on we will refer this sector as a *scan*. Since MSIS needs a considerable period of time to obtain a complete scan, if the robot does not remain static (which is very common in water), the robot's motion induces a distortion in the acoustic image (Fig. 2). To deal with this problem it is necessary to know the robot's pose at the beam reception time. Hence, it is possible to define an initial coordinate system I to reference all the range measurements belonging to the same scan. In order to reduce the influence of the motion uncertainties to the scan, as [18] suggested, we set this initial frame at the robot pose where the center beam of the current scan was read. The localization system used in this work is a slight modification of the navigation system described in [19]. In this system, a MRU provides heading measurements and a DVL unit is used to update the robot's velocity during the scan. MSIS beams are read at 30 Hz while DVL and MRU readings arrive asynchronously at a frequency of 1.5 Hz and 10 Hz respectively. An EKF is used to estimate the robot's 6 Degrees Of Freedom (DOF) pose whenever a sonar beam is read. DVL and MRU readings are used asynchronously to update the filter. To reduce the noise inherent to the DVL measurements, a simple 6 DOF constant velocity kinematics model is used. The model prediction is updated by the standard Kalman filter equations each time a new DVL or MRU measurement arrives.

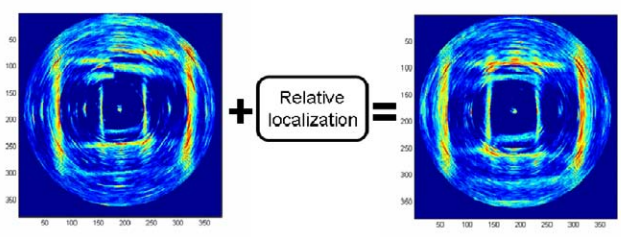


Fig. 2. The distortion produced by the displacement of the robot while acquiring data can be corrected with the relative displacement.

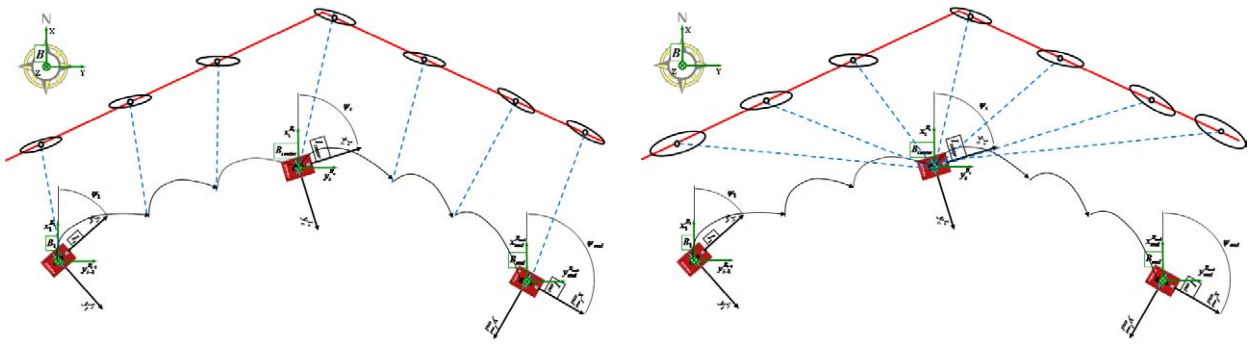
C. Scan Forming

The navigation system presented above is able to estimate the robot's pose, but the uncertainty will grow without limit due to its dead-reckoning nature. Moreover, we are only interested in the robot's relative position (and uncertainty) with respect to the center of the scan (I_{center}-frame, Fig. 3). Hence a slight modification to the filter is introduced making a reset in position (setting x, y, z to 0 in the vector state) whenever a new scan is started. modified filter provides the robot's relative position where the beams where gathered including its uncertainty accumulated during the scan. Hence, it is possible to reference all the ranges computed from the beams to the frame I, removing the distortion induced by the robot's motion.

IV. SCAN MATCHING COVARIANCE

Although the ICP-style algorithms lead to very good estimations of relative displacements, they all missing the part of the uncertainty of their estimation. Calculating the covariance of a measurement is essential when it has to be fused with other measurements in a stochastic localization framework, like the SLAM. To the authors knowledge, very few works exists which try to address in depth this problem. Covariance estimation based on the environment and the uncertainties of the sensor model has been introduced in [20] and [21] but as has been shown in [22], that kind of estimation in a number of cases can be very optimistic. The authors conclude that the Hessian method is suitable for online estimation being able to capture the shape but not the size of the covariance matrix. On the other hand, the second method while capturing the size and shape of the covariance matrix can only be applied offline due to the computational cost.

In a SLAM context, an approach similar to the offline method is used in [23], although in this case only the initial guess of the displacement is sampled. In [24], the authors suggested that the method for estimating the covariance should be independent from the algorithm used for the minimization. They observed the paradox that different optimization methods led to seemingly different uncertainties, but none of them indicate which uncertainty is to be adopted. The Hessian method can be considered a method in this line since it is based on the error function being minimized. Nevertheless, [25] has shown that for square environments the Hessian



a) b)

Fig. 3. The scan forming process, simplified for clarity. a) Each beam is grabbed in different vehicle poses. b) The beams of a full scan are referenced in the center I-frame of the vehicle trajectory covering that scan. The uncertainty of the motion has been propagated to the scan points.

method provides pessimistic estimations for the covariance matrix. In the same work the author proposed an approach formerly proposed by the computer vision community [26], based on: 1) the Hessian of the error function with respect to the scan displacement, and 2) the derivative (with respect to sonar scan points) of the Jacobian (with respect to the scan displacement) of the error function. We will refer this method as Haralicks method, which allows propagating the uncertainty of the sonar points to the uncertainty of the scan matching solution. By means of Monte Carlo simulations, equivalent results for corridor and circle shaped environments and further better results for square type environments are reported in [25]. Also the author has made widely available in his webpage a numerical based implementation of Haralicks method algorithm. Hereafter, the method is adapted to the MSISpIC for which a close form solution is proposed.

A. Closed-form equation

Let $\mathbf{a}_i = [a_{x_i}, a_{y_i}]^T \in S_{ref}$ be the corresponding point of $\mathbf{c}_i = [c_{x_i}, c_{y_i}]^T \in S_{new}$ and $\mathbf{q}_k = [x, y, \theta]^T$ the predicted displacement, then the real robot displacement $\hat{\mathbf{q}}_{min} = [x, y, \theta]^T$ can be estimated by the Non-Linear Least Squares (NLLS) of the MSISpIC, through the minimization of the error function $\mathbf{e}_i = \mathbf{a}_i - \mathbf{q}_k \oplus \mathbf{c}_i$:

$$\mathbf{E}\left(S_{ref}, S_{new}, \hat{\mathbf{q}}_{min}\right) = \frac{1}{2}\sum_{i=1}^{n}\left(\mathbf{e}_i^T \cdot \mathbf{P}_{e_i}^{-1} \cdot \mathbf{e}_i\right) \quad (1)$$

The estimation of the covariance of (1) in a closed form is not trivial. Depending on the error function and the parameters, the construction of the necessary matrices can be tricky. Bellow, we unfold analytically the closed-form expressions for approximating the covariance of (1).

B. Closed-form implementation

Let,

- $\hat{\mathbf{z}}$, be the vector of the measured scan points assumed to be perturbated with a zero mean Gaussian random noise. In our case, the pIC does point-to-point association, therefore the measurements vector $\hat{\mathbf{z}}$ is a dimension of $4n \times 1$:

$$\hat{\mathbf{z}} = [\underbrace{a_{x_1}, a_{y_1}, c_{x_1}, c_{y_1}}_{\hat{\mathbf{z}}_1} \cdots \underbrace{a_{x_n}, a_{y_n}, c_{x_n}, c_{y_n}}_{\hat{\mathbf{z}}_n}]^T \quad (2)$$

  and its covariance, given by $\mathbf{\Sigma_z}$, is a $(4n \times 4n)$ matrix consisted of the uncertainties of the scan points:

$$\mathbf{\Sigma_z} = blockdiag\left(\mathbf{P}_{a_1}, \mathbf{P}_{c_1} \cdots \mathbf{P}_{a_n}, \mathbf{P}_{c_n}\right) \quad (3)$$

  being block diagonal since the scan points are assumed to be uncorrelated. As a difference with laser scanners or multi-beam sonar profilers, when using a rotating mono-beam sonar head, the scan points become correlated when represented in the scan I-frame. Nevertheless, for the sake of simplicity, those correlations have been neglected in this work and their impact will be part of our future research.
- $\hat{\mathbf{x}}$, be the unknown parameters vector corresponding to the $\hat{\mathbf{q}}_{min}$ estimated by the the pIC.

$$\hat{\mathbf{x}} = [x, y, \theta]^T \quad (4)$$

- $f(\mathbf{z}, \mathbf{x})$, be an scalar, continuous, non-negative function, then (1) can be defined as:

$$f(\mathbf{z}, \mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n}\left(\mathbf{e}_i^T \cdot \mathbf{P}_{e_i}^{-1} \cdot \mathbf{e}_i\right) \quad (5)$$

then, we can apply [26] in order to estimate $\mathbf{\Sigma_x}$:

$$\mathbf{\Sigma_x} = \left(\frac{\partial g}{\partial \mathbf{x}}\right)^{-1} \cdot \frac{\partial g}{\partial \mathbf{z}} \cdot \mathbf{\Sigma_z} \cdot \left(\frac{\partial g}{\partial \mathbf{z}}\right)^{T} \cdot \left(\frac{\partial g}{\partial \mathbf{x}}\right)^{-1} \quad (6)$$

where $g(\mathbf{z}, \mathbf{x}) = \left[\frac{\partial f(\mathbf{z},\mathbf{x})}{\partial \mathbf{x}}\right]^T$. To do this, it is necessary to compute $g(\mathbf{z}, \mathbf{x})$, $\frac{\partial g(\mathbf{z},\mathbf{x})}{\partial \mathbf{x}}$ and $\frac{\partial g(\mathbf{z},\mathbf{x})}{\partial \mathbf{z}}$. Let us begin rewriting the function (1) as:

$$f(\mathbf{z}, \mathbf{x}) = \frac{1}{2} \cdot \mathbf{R}^T \cdot \mathbf{W} \cdot \mathbf{R} \quad (7)$$

where $\mathbf{R}$ is the stacked vector of the measurement errors (of dimension $2n \times 1$):

$$\mathbf{R} = \left[\mathbf{e}_1^T \cdots \mathbf{e}_n^T\right]^T \quad (8)$$

and $\mathbf{W}$ is the inverted block diagonal matrix of the measurement errors covariances (of dimension $2n \times 2n$):

$$\mathbf{W} = blockdiag\left(\mathbf{P}_{e_1} \cdots \mathbf{P}_{e_n}\right)^{-1} \tag{9}$$

Because $\mathbf{W}$ is the inverse of a covariance matrix, it is positive-definite and hence a Cholesky decomposition $\mathbf{W} = \mathbf{L}^T \cdot \mathbf{L}$ exists. Now, for simplicity and without lost of generality, let us define:

$$\hat{\mathbf{R}} = \mathbf{L} \cdot \mathbf{R} \tag{10}$$

$\mathbf{W}$ is block diagonal because the scan points are assumed to be uncorrelated, so as is $\mathbf{L}$:

$$\mathbf{L} = blockdiag\left(\mathbf{L}_{e_1} \cdots \mathbf{L}_{e_n}\right) \tag{11}$$

Now, equation (7) can be rewritten as:

$$f\left(\mathbf{z}, \mathbf{x}\right) = \frac{1}{2} \cdot \hat{\mathbf{R}}^T \cdot \hat{\mathbf{R}} \tag{12}$$

and $g(\mathbf{z}, \mathbf{x})$ can be defined as the Jacobian of the cost function, being a (1×3) matrix;

$$g\left(\mathbf{z}, \mathbf{x}\right) = \frac{\partial f}{\partial x} = \hat{\mathbf{R}}^T \cdot \hat{\mathbf{J}}_x \tag{13}$$

where $\hat{\mathbf{J}}_x$ is the $(2n \times 3)$ Jacobian matrix of error vector $\hat{\mathbf{R}}$:

$$\hat{\mathbf{J}}_x = \frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial (\mathbf{L}_{e_1} \cdot \mathbf{e}_1)}{\partial x} & \frac{\partial (\mathbf{L}_{e_1} \cdot \mathbf{e}_1)}{\partial y} & \frac{\partial (\mathbf{L}_{e_1} \cdot \mathbf{e}_1)}{\partial \theta} \\ \vdots & \vdots & \vdots \\ \frac{\partial (\mathbf{L}_{e_n} \cdot \mathbf{e}_n)}{\partial x} & \frac{\partial (\mathbf{L}_{e_n} \cdot \mathbf{e}_n)}{\partial y} & \frac{\partial (\mathbf{L}_{e_n} \cdot \mathbf{e}_n)}{\partial \theta} \end{bmatrix} \tag{14}$$

The $\frac{\partial g(\mathbf{z},\mathbf{x})}{\partial \mathbf{x}}$ is the (3×3) Hessian of $f\left(\mathbf{z}, \mathbf{x}\right)$ and is calculated as follows:

$$\frac{\partial g}{\partial \mathbf{x}} = 2 \cdot \hat{\mathbf{J}}_{\mathbf{x}}^T \cdot \hat{\mathbf{J}}_{\mathbf{x}} + 2 \cdot \hat{\mathbf{R}}^T \frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{x}} \tag{15}$$

where $\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{x}}$ is the $(6n \times 3)$ Hessian of $\hat{\mathbf{R}}$ which can be computed in the following way:

$$\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}} \left(\frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{x}} \right) = \sum_{i=1}^{3} \left(vec \left(\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial x_i} \right) \right) \cdot \mathbf{r}_i^T \tag{16}$$

being $\mathbf{r}_i$ a (3×1) vector, with all zeros except its i^{th} row which is equal to 1. To compute the second part of (15), the (16) is multiplied by $\hat{\mathbf{R}}^T$ as follows:

$$\hat{\mathbf{R}}^T \frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{x}} = \left(\hat{\mathbf{R}}^T \otimes I_3 \right) \cdot \frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{x}} \tag{17}$$

where $\otimes$ denotes Kronecker product of two matrices. Similarly, the $\frac{\partial g}{\partial \mathbf{z}}$ is a $3 \times 4n$ matrix, computed as:

$$\frac{\partial g}{\partial \mathbf{z}} = 2 \cdot \hat{\mathbf{J}}_{\mathbf{x}}^T \cdot \hat{\mathbf{J}}_{\mathbf{z}} + 2 \cdot \hat{\mathbf{R}}^T \frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{z}} \tag{18}$$

where $\hat{\mathbf{J}}_z$ is the Jacobian of the error vector $\hat{\mathbf{R}}$, being a $(2n \times 4n)$ matrix:

$$\hat{\mathbf{J}}_z = \frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{z}} = blockdiag \left(\frac{\partial \hat{\mathbf{R}}_1}{\partial \mathbf{z}} \cdots \frac{\partial \hat{\mathbf{R}}_n}{\partial \mathbf{z}} \right) \tag{19}$$

where each $\frac{\partial \hat{\mathbf{R}}(i)}{\partial \mathbf{z}}$ is a (2×4) matrix, and $\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{z}}$ is the following $(6n \times 4n)$ matrix:

$$\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{z}} = \sum_{i=1}^{4n} \left(vec \left(\frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial z_i} \right) \right) \cdot \mathbf{r}_i^T \tag{20}$$

being this time r_i a $(4n \times 1)$ vector of all zeros except its i^{th} row which is equal to 1. As previously, the second part of (18) is given by:

$$\hat{\mathbf{R}}^T \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial \hat{\mathbf{R}}}{\partial \mathbf{x}} \right) = \left(\hat{\mathbf{R}}^T \otimes \mathbf{I}_3 \right) \cdot \frac{\partial \hat{\mathbf{J}}_{\mathbf{x}}}{\partial \mathbf{z}} \tag{21}$$

The verification of the above formulation has been carried out in [27] with extensive Monte Carlo testing and it also compared against the numerical based implementation reported in [25], being in agreement up to the fifth decimal digit.

V. SLAM ALGORITHM

The proposed pose-based SLAM algorithm uses an ASEKF for the scan poses estimation. In this implementation of the stochastic map [28], the estimate of the positions of the vehicle at the center of each full scan $\{\mathbf{x}_1 \ldots \mathbf{x}_n^B\}$ at the time step (k), are stored in the state vector $\hat{\mathbf{x}}$:

$$\hat{\mathbf{x}}_k^B = \left[\hat{\mathbf{x}}_{n_k}^B \ \ldots \ \hat{\mathbf{x}}_{i_k}^B \ \ldots \ \hat{\mathbf{x}}_{1_k}^B \right]^T \tag{22}$$

and the covariance matrix for this state is defined as:

$$\mathbf{P}_k^B = E([\mathbf{x}_k^B - \hat{\mathbf{x}}_k^B][\mathbf{x}_k^B - \hat{\mathbf{x}}_k^B]^T) \tag{23}$$

Note that, a full scan is defined as the final 360° polar range image obtained after compounding, along the path, the robot pose with the range and bearing data, which is the output from the MSISpIC algorithm.

A. Map Initialization

All the elements on the state vector are represented in the map reference frame B. Although this reference frame can be defined arbitrarily, we have chosen to place its origin on the initial position of the vehicle and orient it to the north, so compass measurements can be easily integrated.

The pose state $\mathbf{x}_i$ is represented as:

$$\mathbf{x}_i^B = [x \ y \ \psi]^T \tag{24}$$

where, x, y and ψ is the position and orientation vector of the vehicle in the global frame B. The state and the map are initialized from the first full scan obtained by the MSISpIC algorithm.

B. Prediction

Let

- $\mathbf{x}_{n_k}^B \equiv N(\hat{\mathbf{x}}_{n_k}^B, \ \mathbf{P}_{q_k}^B)$ be the last robot pose and
- $\hat{\mathbf{q}}_n^{B_n} \equiv N(\hat{\mathbf{q}}_n^{B_n}, \ \mathbf{P}_{q_n}^B)$ be the robot displacement during the last scan, estimated through dead reckoning

then the prediction / state augmentation equation is given by:

$$\hat{\mathbf{x}}_{k+1}^B = \hat{\mathbf{x}}_k^B \odot \hat{\mathbf{q}}_{n_k}^{B_n} = \tag{25}$$

$$\left[\ \hat{\mathbf{x}}^B_{n-1_k} \odot \hat{\mathbf{q}}^{B_n}_{n_k} |\ \hat{\mathbf{x}}^B_{n-1_k} \dots \hat{\mathbf{x}}^B_{i_k} \dots \hat{\mathbf{x}}^B_{1_k}\ \right]^T \tag{26}$$

where, given that B and B_n frames are both north aligned, the operator $\odot$ is defined as:

$$\mathbf{x} \odot \mathbf{q} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \odot \begin{bmatrix} d \\ e \\ f \end{bmatrix} = \begin{bmatrix} a+d \\ b+e \\ f \end{bmatrix} \tag{27}$$

being $\mathbf{J}_{1\odot}$ and $\mathbf{J}_{2\odot}$ the corresponding linear transformation matrices:

$$\mathbf{J}_{1\odot} = \begin{bmatrix} \mathbf{I}_{2x2} & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{J}_{2\odot} = \mathbf{I}_{3x3}$$

and being the predicted pose uncertainty $\mathbf{P}^{B+}_k$ computed as:

$$\mathbf{P}^B_{k+1} = \mathbf{F}_k \mathbf{P}^B_k \mathbf{F}^T_k + \mathbf{G}_k \mathbf{P}^B_{q_i} \mathbf{G}^T_k \tag{28}$$

where,

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{J}_{1\odot} & \mathbf{0}_{3x3} & \dots & \mathbf{0}_{3x3} \\ \mathbf{0}_{3x3} & \mathbf{I}_{3x3} & \dots & \mathbf{0}_{3x3} \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{0}_{3x3} & \mathbf{0}_{3x3} & \dots & \mathbf{I}_{3x3} \end{bmatrix} \quad \mathbf{G}_k = \begin{bmatrix} \mathbf{J}_{2\odot} \\ \mathbf{0}_{3x3} \\ \vdots \\ \mathbf{0}_{3x3} \end{bmatrix}$$

C. Loop Closing Candidates

Each new pose of a scan is compared against the previous scan poses that are in the nearby area defined by a threshold. Whenever enough points are overlapping, a new scan match puts a constraint between the poses updating the ASEKF. These constraints close the loops correcting the whole trajectory and bounding the drift.

Let

- $\mathbf{x}^B_{n_k}$ be the last scan pose and S_{n_k} the corresponding scan,
- $Overlap_k = \left\{ S_{i_k} \ / \ \left\| \hat{\mathbf{x}}^B_{n_k} - \hat{\mathbf{x}}^B_{i_k} \right\| < threshold \right\}$ the set of overlapping scans and $\mathbf{O}_k = [S_{1_k},\ S_{2_k}\ \dots\ S_{m_k}]$ the sequence of overlapping scans belonging to the $Overlap_k$ set

then $\forall\ [S_{i_k}, \mathbf{x}^B_i] \in \mathbf{O}_k$, perform a new scan matching between the scan poses $(\mathbf{x}^B_{n_k}, \mathbf{x}^B_{i_k})$ with the corresponding scans (S_{n_k}, S_{i_k}) obtaining $[\hat{\mathbf{q}}^{I_i}_i, \mathbf{R}^{I_i}_{q_i}]$ the result of the scan matching. $\mathbf{R}^{I_i}_{q_i}$ is the corresponding uncertainty computed as described in Section IV. Finally the scan matching result is used to update the filter.

D. Scan Matching

In order to execute the modified pIC algorithm, given two overlapping scans (S_i, S_n) with related poses $(\mathbf{x}^B_i, \mathbf{x}^B_n)$, an initial guess of their relative displacement is necessary. This initial guess $[\hat{\mathbf{q}}^{I_i}_i, \mathbf{P}^{I_i}_i]$ can be easily extracted from the state vector using the tail-to-tail transformation [28]:

$$\hat{\mathbf{q}}^{I_i}_i = \ominus \hat{\mathbf{x}}^B_i \oplus \hat{\mathbf{x}}^B_n \tag{29}$$

Since the tail-to-tail transformation is actually a nonlinear function of the state vector $\hat{\mathbf{x}}^B_k$, the uncertainty of the initial guess can be computed by means of the Jacobian of the non linear function:

$$\mathbf{P}^{I_i}_{q_i} = \mathbf{H}_k \mathbf{P}^B_k \mathbf{H}^T_k \tag{30}$$

where

$$\mathbf{H}_k = \left. \frac{\partial\ \ominus \hat{\mathbf{x}}^B_i \oplus \hat{\mathbf{x}}^B_n}{\partial\ \mathbf{x}^B_k} \right|_{(\mathbf{x}^B_k = \hat{\mathbf{x}}^B_k)} \tag{31}$$

Moreover, as shown in [28], the Jacobian for the tail-to-tail transformation $\mathbf{x}_{a_c} = \ominus \mathbf{x}_{b_a} \oplus \mathbf{x}_{b_c}$, is:

$$\frac{\partial\ \ominus \mathbf{x}_{b_a} \oplus \mathbf{x}_{b_c}}{\partial\ (\mathbf{x}_{b_a} \mathbf{x}_{b_c})} = [\mathbf{J}_{1\oplus} \mathbf{J}_{\ominus}\ \ \mathbf{J}_{2\oplus}] \tag{32}$$

where the $\mathbf{J}_{1\oplus}, \mathbf{J}_{2\oplus}$ and $\mathbf{J}_{\ominus}$ are the Jacobian matrices of the compounding and inverse transformations respectively.

Being in our case $\hat{\mathbf{x}}^B_{n_k}$ and $\hat{\mathbf{x}}^B_{i_k}$ components of the full state vector, the Jacobian of the measurement equation becomes:

$$\mathbf{H}_k = \frac{\partial\ \ominus \hat{\mathbf{x}}^B_{i_k} \oplus \hat{\mathbf{x}}^B_{n_k}}{\partial\ \mathbf{x}_k} =$$
$$[\mathbf{J}_{2\oplus_{3x3}}\ \ \mathbf{0}_{3x3(n-i-1)}\ \ \mathbf{J}_{1\oplus} \mathbf{J}_{\ominus_{3x3}}\ \ \mathbf{0}_{3x3(i-1)}]$$

Once the initial displacement guess is available, the pIC algorithm can be used to produce an updated measurement of this displacement.

E. State Update

When two overlapping scans (S_i, S_n) with the corresponding poses $(\mathbf{x}^B_i, \mathbf{x}^B_n)$ are registered, their relative displacement defines a constraint between both poses. This constraint can be expressed by means of the measurement equation, which again in our case becomes:

$$\mathbf{z}_k = \ominus \hat{\mathbf{x}}^B_{i_k} \oplus \hat{\mathbf{x}}^B_{n_k} \tag{33}$$

where $\hat{\mathbf{x}}^B_{i_k}$ is the scan pose which overlaps with the last scan pose $\hat{\mathbf{x}}^B_{n_k}$. Now, an update of the stochastic map can be performed with the standard extended Kalman filter equations.

VI. EXPERIMENTAL RESULTS

The method described in this paper has been tested with a dataset obtained in an abandoned marina located in the Catalan coast [29]. This dataset corresponds to structured environment but our algorithm does not take into account any structural information neither features and with the current sensor suite, it can been used wherever there is enough vertical information. The survey mission was carried out using ICTINEU^{AUV} [30] traveling along a 600 m path. The MSIS was configured to scan a full 360° sector and it was set to maximum range of 50 m with a 0.1 m resolution and a 1.8° angular step. With those settings, the MSIS needed around 14 secs to complete a full scan. Dead-reckoning was computed using the velocity reading coming from the DVL and the heading data obtained from the MRU sensor, both merged using the described EKF. The whole dataset was acquired in 53 mins and the off-line execution of the algorithm implemented in MATLAB took around 14 mins in a simple Pentium M @2,00 GHz laptop, which gives good possibilities for real time implementation of the proposed algorithm.

Fig. 4 shows the trajectory and the map estimated with the proposed SLAM algorithm and in Fig. 6, the results are projected on an ortophotomap of the real environment. As

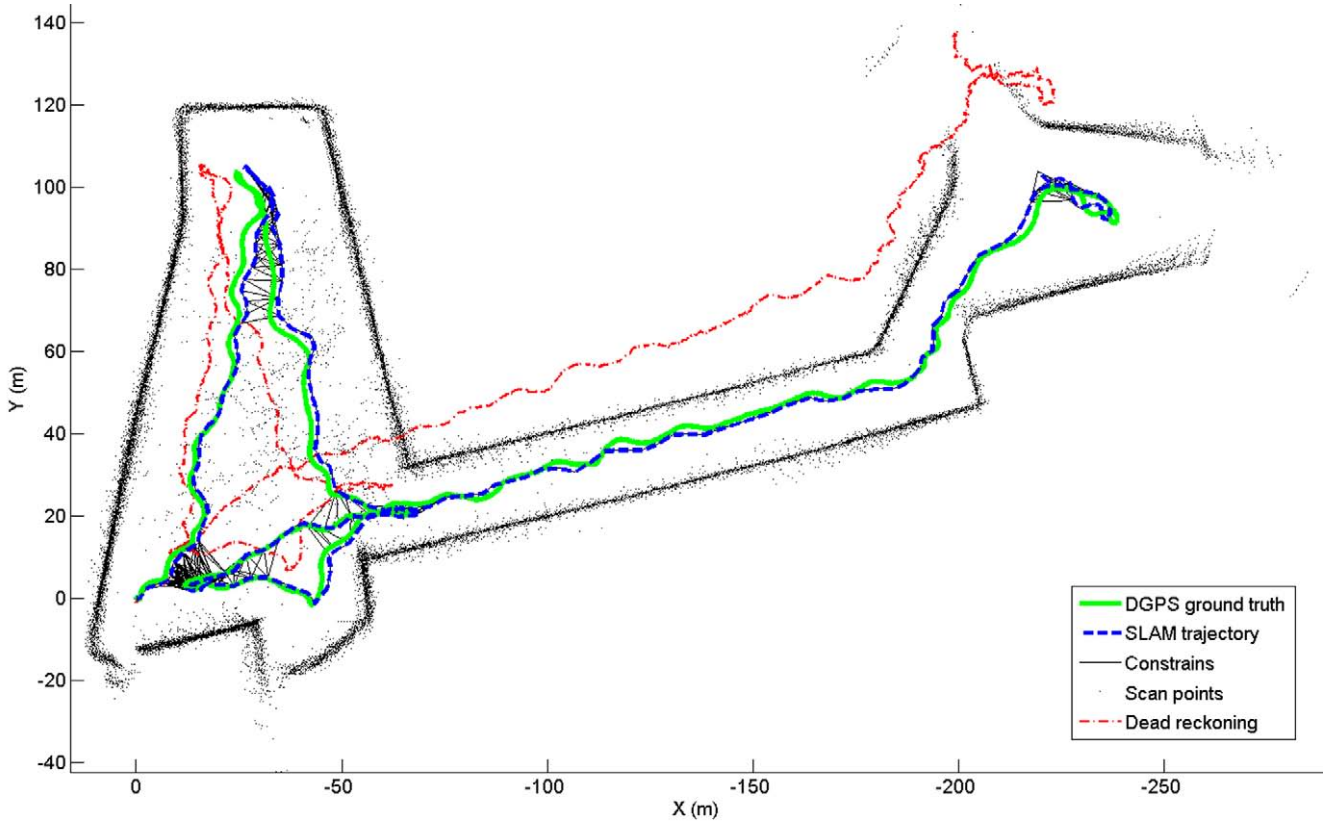
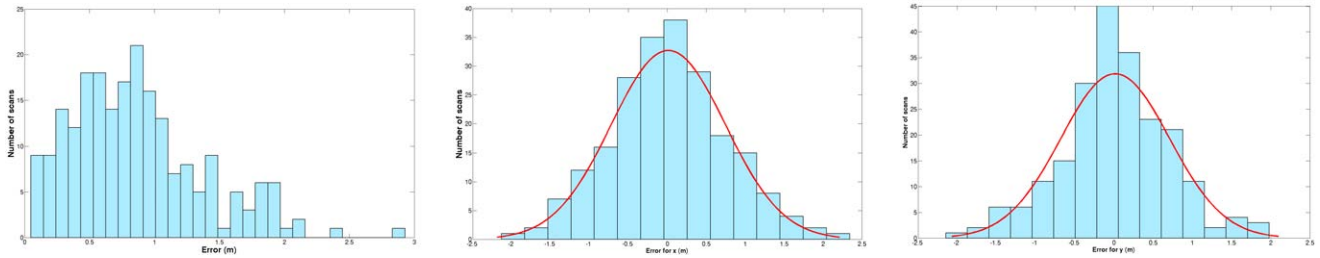


Fig. 4. SLAM Trajectory and map. In red is the dead reckoning (dash-dot), in solid green line the DGPS trajectory used as a ground truth and in blue (dash) the trajectory estimated with the SLAM algorithm.



a) b) c)

Fig. 5. Error analysis for all the scan poses as the displacement estimated through the SLAM and the corresponding displacement estimated with the DGPS. a) absolute error. b) Error in X vectors. c) Error in Y vectors. The solid line is the Gaussian fit to the histogram.

was expected, the dead-reckoning estimated trajectory suffers from a significant drift which is drastically limited by the SLAM algorithm. The maximum absolute error is 2.9 m as shown in fig. 5 (a). The absolute error is computed as the difference between the displacement estimated through the SLAM algorithm and the corresponding displacement estimated with the DGPS, assuming zero DGPS error. In the (b) and (c) figures are the histograms of the errors for the X and Y vectors. The solid line is the Gaussian fit to the histogram.

The nominal accuracy of a DGPS is around 1 m and it degrades at an approximate rate of 1 m for each 150 km distance from the broadcast site [31]. In our experiment, we were receiving differential corrections from a nearby base station (<40 km), thus is reasonable to assume that the DGPS drift was not more than 2 m.

VII. CONCLUSIONS

This paper proposes an extension to the MSISpIC algorithm in the pose-based SLAM framework. To deal with the motion induced distortion of the acoustic image, an EKF is used to estimate the robot motion during the scan. The filter uses a constant velocity model with acceleration noise for motion prediction and velocity (DVL) and attitude measurements (MRU) for updating the state. Through the compounding of the relative robot position within the scan, with the range and bearing measurements of the beams gathered with the sonar, the acoustic image gets undistorted. Assuming Gaussian noise, the algorithm is able to predict the uncertainty of the sonar measurements with respect to

Fig. 6. Trajectory and map estimated with the SLAM algorithm projected on the orthophoto map.In red are the sonar map, in solid green line the DGPS trajectory and in blue (dash) the trajectory estimated with the SLAM algorithm.

a frame located at the position occupied by the robot at the center of the scan. Each full scan pose is maintained in a second filter, an augmented EKF, and is cross registered with all the previous scan poses that are in a certain range applying a modified pIC algorithm. A closed form method of estimating the uncertainty of the minimization algorithm is presented. The proposed method has been tested with a real world dataset including DGPS for ground truth acquired during a survey mission in an abandoned marina located in the Girona coast. The results show substantial improvements in trajectory correction and map reconstruction over the MSISpIC results [8].

REFERENCES

[1] D. Hahnel, W. Burgard, D. Fox, and S. Thrun, "An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. IEEE/RSJ International Conference on*, vol. 1, 27-31 Oct. 2003, pp. 206–211.

[2] U. Castellani, A. Fusiello, V. Murino, L. Papaleo, E. Puppo, S. Repetto, and M. Pittore, "Efficient on-line mosaicing from 3D acoustical images," *OCEANS'04. MTTS/IEEE TECHNO-OCEAN'04*, vol. 2, pp. 670–677, 2004.

[3] D. Silver, D. Bradley, and S. Thayer, "Scan matching for flooded subterranean voids," in *IEEE conference on Robotics Automation and Mechatronics (RAM)*, vol. 1, December 2004, pp. 422 – 427.

[4] C. Clark, C. Olstad, K. Buhagiar, T. Gambin, A. Special, and P. Trust, "Archaeology via Underwater Robots: Mapping and Localization within Maltese Cistern Systems," in *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, 2008, pp. 662–667.

[5] C. Roman and H. Singh, "Improved vehicle based multibeam bathymetry using sub-maps and SLAM," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005.(IROS 2005)*, 2005, pp. 3662–3669.

[6] S. Barkby, S. Williams, O. Pizarro, and M. Jakuba, "An efficient approach to bathymetric slam," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, Oct. 2009, pp. 219–224.

[7] N. Fairfield, G. Kantor, and D. Wettergreen, "Real-time slam with octree evidence grids for exploration in underwater tunnels," *Journal of Field Robotics*, vol. 24, no. 1, p. 2, 2007.

[8] E. Hernandez, P. Ridao, D. Ribas, and J. Batlle, "Msispic: A probabilistic scan matching algorithm using a mechanical scanned imaging sonar," *Journal on Physical Agents (JoPhA)*, vol. 3, no. 1, pp. 3–12, 2009.

[9] L. Montesano, J. Minguez, and L. Montano, "Probabilistic scan matching for motion estimation in unstructured environments," in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, Aug. 2005, pp. 3499–3504.

[10] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, pp. 333–349, 1997.

[11] S. Thrun, Y. Liu, D. Koller, A. Y. Ng, Z. Ghahramani, and H. F. Durrant-Whyte, "Simultaneous localization and mapping with sparse extended information filters," *International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 693–716, 2004.

[12] R. Eustice, H. Singh, and J. Leonard, "Exactly sparse delayed-state filters," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, 2005, pp. 2417–2424.

[13] P.-J. M. A.-C. J. Ila, V., "Information-based compact pose slam," *Robotics, IEEE Transactions on*, vol. 26, no. 1, pp. 78 –93, Feb. 2010.

[14] U. Frese, "Treemap: An O (log n) algorithm for simultaneous localization and mapping," *Spatial Cognition IV. Reasoning, Action, and Interaction*, pp. 455–477, 2005.

[15] E. Olson, J. Leonard, and S. Teller, "Spatially-adaptive learning rates for online incremental SLAM," in *Robotics: Science and Systems (RSS)*, vol. 1. Citeseer, 2007.

[16] M. Kaess, "Incremental smoothing and mapping," 2008.

[17] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard, "A tree parameterization for efficiently computing maximum likelihood maps using gradient descent," in *Proc. of Robotics: Science and Systems (RSS)*. Citeseer, 2007.

[18] A. Burguera, Y. González, and G. Oliver, "A Probabilistic Framework for Sonar Scan Matching Localization," *Advanced Robotics*, vol. 22, no. 11, pp. 1223–1241, 2008.

[19] D. Ribas, P. Ridao, J. Tardós, and J. Neira, "Underwater SLAM in man made structured environments," *Journal of Field Robotics*, vol. 25, no. 11-12, pp. 898–921, July 2008.

[20] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2d range scans," *Journal of Intelligent and Robotic Systems*, vol. 18, no. 3, pp. 249–275, 1997.

[21] S. Pfister, K. Kriechbaum, S. Roumeliotis, and J. Burdick, "Weighted range sensor matching algorithms for mobile robot displacement estimation," in *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, vol. 2, 2002, pp. 1667–1674.

[22] O. Bengtsson and A.-J. Baerveldt, "Robot localization based on scan-matching - estimating the covariance matrix for the idc algorithm," *Robotics and Autonomous Systems*, vol. 44, no. 1, pp. 29 – 40, Jun 2003, best Papers of the Eurobot '01 Workshop.

[23] J. Nieto, T. Bailey, and E. Nebot, "Recursive scan-matching SLAM," *Robotics and Autonomous Systems*, vol. 55, no. 1, pp. 39–49, Jan 2007.

[24] A. Balsamo, G. Mana, and F. Pennecchi, "The expression of uncertainty in non-linear parameter estimation," *Metrologia*, vol. 43, no. 5, p. 396, 2006. [Online]. Available: http://stacks.iop.org/0026-1394/43/i=5/a=009

[25] A. Censi, "An accurate closed-form estimate of ICP's covariance," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Rome, Italy, apr 2007, pp. 3167–3172. [Online]. Available: http://purl.org/censi/2006/icpcov

[26] R. Haralick, "Propagating covariance in computer vision," *Performance Characterization in Computer Vision*, vol. 1, pp. 95 – 115, 1998.

[27] A. Mallios, P. Ridao, A. Elibol, R. Garcia, F. Maurelli, and Y. Petillot, "On the closed-form covariance estimation for the pic algorithm," Universitat de Girona (UDG), Tech. Rep. IIiA 10-02-RR, 2010. [Online]. Available: http://eia.udg.edu/∼pere/IIiA.pdf

[28] R. Smith, M. Self, and P. Cheeseman, *Estimating uncertain spatial relationships in robotics*. New York, NY, USA: Springer-Verlag New York, Inc., 1990, pp. 167–193.

[29] D. Ribas. (2009, March) David Ribas homepage. [Online]. Available: http://eia.udg.edu/∼dribas

[30] D. Ribas, N. Palomer, P. Ridao, M. Carreras, and E. Hernandez, "Ictineu AUV wins the first SAUC-E competition," in *IEEE International Conference on Robotics and Automation*, Roma, Italy, apr 2007.

[31] L. Monteiro, T. Moore, and C. Hill, "What is the accuracy of DGPS?" *The Journal of Navigation*, vol. 58, no. 02, pp. 207–225, 2005.