A Trajectory Based Framework to Perform Underwater SLAM using Imaging Sonar Scans

Antoni Burguera, Gabriel Oliver and Yolanda González
Dept. Matemàtiques i Informàtica - Universitat de les Illes Balears
Ctra. Valldemossa Km. 7,5 - 07122 Palma de Mallorca (SPAIN)
{antoni.burguera,goliver,y.gonzalez}@uib.es

Abstract

This paper proposes a framework to perform Simultaneous Localization and Mapping *(SLAM) using the scans gathered by a* Mechanically Scanned Imaging Sonar *(MSIS). To this end, the acoustic profiles provided by the MSIS are processed to obtain range data. Also, dead reckoning is used to compensate the robot motion during the sonar mechanical scanning and build range scans.*

When a new scan is constructed, its estimated position with respect to the previously gathered one is used to augment the SLAM state vector. Also, each new scan is matched against the previously detected ones by means of scan matching techniques. As the state vector contains relative positions between consecutively gathered scans, the measurement update explicitly takes into account the robot trajectory involved in each loop closure.

1. Introduction

A crucial issue for a mobile robot to execute useful long term missions is to determine and keep track of its position. The *Simultaneous Localization and Mapping* (SLAM) [5, 1] has shown to be one of the most successful approaches to estimate the robot pose. A wide variety of SLAM algorithms exist in the literature. However, the most common SLAM approach is based on feature maps composed of straight lines and corners. Although this approach has proved its accuracy and robustness in structured terrestrial [14] and underwater [15, 12] environments, the use of feature maps reduces the scenarios where the robot can be deployed. This problem is especially important in underwater scenarios where man made, structured, environments are uncommon.

The use of raw range data in SLAM not assuming any type of feature in the environment has been proposed as an alternative to feature based SLAM [11]. Moreover, the feasibility of SLAM in underwater environments using the raw readings of a *Mechanically Scanned Imaging Sonar* (MSIS) has been demonstrated recently [10].

When used to perform SLAM based on raw range readings, a MSIS has two important problems. Firstly, this sensor does not provide range measurements but echo intensity profiles. Accordingly, the sensor information has to be processed before being used in this context. A simple yet effective method to perform such process has been described in [12]. Secondly, the scan time of a MSIS is not negligible. For example, in our particular configuration, the sensor needs more than 13 seconds to gather a 360^o scan. In consequence, it can not be assumed that the robot remains static while the scan is being obtained. Some considerations regarding this issue using terrestrial Polaroid sensors are provided in [3]. Moreover, a recent study by [8] shows the feasibility of underwater localization based on matching MSIS scans.

This paper proposes a framework to perform SLAM using the scans gathered by a MSIS. The problem is summarized section 2. Section 3 states our proposal to deal with the aforementioned MSIS problems. The proposed SLAM framework is detailed in section 4. This approach has two key points. Firstly, the measurement model is based on scan matching techniques. Two different measurement models are stated. Secondly, the relationship between consecutively gathered scans is exploited to correct the robot trajectory instead of only the current robot pose. In order to deal with the trajectory correction, two approaches are proposed. The first one is based on an *Extended Kalman Filter* (EKF) and the second one relies on an *Iterated EKF* (IEKF). The experimental results evaluating the different proposals are provided in section 5. Finally, section 6 concludes the paper.

2. Problem statement

The experiments conducted in this paper have been performed using the sensor data gathered by the *Ictineu AUV*. This *Autonomous Underwater Vehicle* (AUV) was designed and developed at the University of Girona (see [12] for more details). Among other sensors, the AUV is endowed with a *Doppler Velocity Log* (DVL) which measures the velocities of the unit with respect to bottom and water, a *Motion Reference Unit* (MRU) that provides absolute attitude data by means of compass and inclinometers, and a MSIS.

The MSIS obtains 360^o scans of the environment by

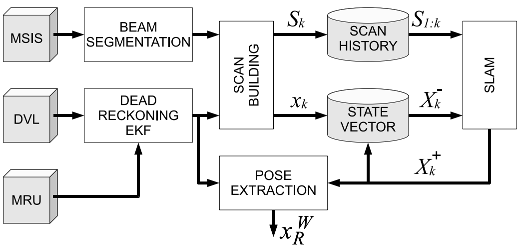
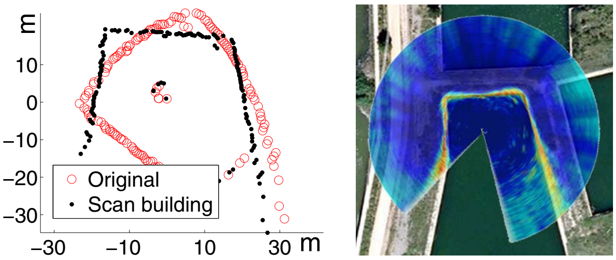


Figure 1. The framework. The notation is explained throughout the paper.



(a) (b)

Figure 2. (a) Range data before and after the scan building. (b) Acoustic image of one MSIS scan after the scan building.

rotating a sonar beam through 200 angular steps in about 13.8 seconds. At each angular position, a set of 500 values, named *bins*, is obtained representing a 50 m long echo intensity profile with a resolution of 10cm. Each of these sets of 500 bins will be referred to as *beam*. By accumulating this information, an *acoustic image* of the environment can be obtained.

As stated previously, different problems arise when using a MSIS to perform scan-based localization. In order to solve them several processes are necessary. Our proposal is summarized in Figure 1. First, range information is extracted from each MSIS measurement by means of the *beam segmentation*. Also, DVL and MRU readings are fused by means of an EKF to obtain *dead reckoning* estimates, as described in [12]. Both the obtained range information and the dead reckoning estimates are stored in two buffers, called *readings history* and *transformations history* respectively. When the MSIS has rotated 360^o, the information in these buffers is used by the *scan building* to compensate the robot motion and build the scan S_k. Also, the scan building computes the motion x_k between the previously gathered scan and the current one. The scans are stored in the so called *scan history* and the motions are used to augment the SLAM state vector. Afterwards, the new scan S_k is matched against the scans in the scan history to perform the SLAM state update. This state update exploits the relations between consecutively gathered scans, similarly to the top level in the *Hierarchical SLAM* approach [6]. Finally, the robot pose with respect to a fixed coordinate frame, x_R^W, is computed by the *pose extraction* process. The rest of the paper is devoted to describe the abovementioned processes.

3. Building the scans

Our proposal to build the scans consists of two steps. First, the beam segmentation extracts a single range from each acoustic profile. Second, after a 360^o rotation of the MSIS, the scan building combines the obtained ranges with the dead reckoning estimates in order to reduce the distortions induced by the robot motion. A detailed description of both processes is provided in [4]. Next, both processes are summarized.

The beam segmentation process is as follows. Firstly, a dynamic thresholding method is used to remove the low echo intensities from each beam. Secondly, the remaining bins are eroded to remove spurious echoes and peaks due to backscatter. Finally, among the remaining bins, the one with the highest echo intensity is selected and used to compute the range data.

In order to build the scan, the ranges obtained during the last 360^o MSIS rotation have to be represented with respect to a common coordinate frame. As the robot motions during the MSIS data gathering are stored in the transformations history, the ranges can be represented with respect to any robot pose estimate during that period. Our proposal is to represent the ranges with respect to the central position of the robot trajectory during the scan data gathering. The central position has been chosen for two main reasons. On the one hand, because of the similarity to the scans generated by a laser range finder. On the other hand, in order to reduce the maximum uncertainty of each reading with respect to the reference frame. This process is referred to as the scan building, and it outputs both the scan itself, S_k, and the estimated robot motion x_k between the coordinate frame of the previously built scan and the current one. This robot motion is computed from the transformations history and is modeled by a normal distribution $N(\hat{x}_k, P_{xk})$ representing both the estimated motion and its uncertainty. The parameters of this distribution are provided by the dead reckoning EKF. For the particular case of the first scan, x_1 represents the robot motion from the starting pose to the first scan coordinate frame.

Figure 2-a illustrates the process by showing the set of ranges extracted by the beam segmentation before and after the scan building. The effects of the scan building can be clearly appreciated in Figure 2-b, where the acoustic image corresponding to the corrected scan is overlayed to a satellite view of the environment.

4. Trajectory-based SLAM

Our scan-based approach is based on the EKF concepts. The state vector contains the relative positions between consecutively gathered scans. That is, the state vector contains explicit information about the robot trajectory. The mean $\hat{X}_k$ of the state vector X_k after gathering the k-th scan is

$$\hat{X}_k = [\hat{x}_{1,k}, \hat{x}_{2,k}, ..., \hat{x}_{k,k}]^T \quad (1)$$

and the state vector covariance is denoted by P_k.

The robot pose with respect to an earth-fixed coordinate frame could be included in the state vector. However, for the sake of simplicity, we have decided to build a state vector containing only the map. As the map stores relative poses between consecutively obtained scans, the pose of the most recent scan with respect to the robot's starting pose is $\hat{x}_{1,k} \oplus \hat{x}_{2,k} \oplus ... \oplus \hat{x}_{k,k}$, where $\oplus$ denotes the compounding operator, commonly used in stochastic mapping. Additionally, all the information needed to compute the current robot pose with respect to the last scan is available in the transformations history. Using this data the current robot pose with respect to the first robot pose, x_R^W, can be easily computed by the pose extraction process (see Figure 1).

4.1. Prediction and state augmentation

Under the assumption of a static environment, the EKF prediction step does not change the state vector. That is, $X_k^- = X_{k-1}^+$.

After the execution of the scan building process, both the scan S_k and the relative motion x_k are available. The scan is stored in the scans history SH_k so that

$$SH_k = \{S_1, S_2, ..., S_k\} \quad (2)$$

The relative motion $x_k = N(\hat{x}_k, P_{xk})$ is used to augment the state. The mean of the augmented state vector is $\hat{X}_k = [\hat{X}_{k-1}, \hat{x}_k]^T$. At this point, the relative motion is independent to the previous ones. Thus, the covariance is constructed as follows:

$$P_k = \begin{bmatrix} P_{k-1} & 0_{3(k-1)\times 3} \\ 0_{3\times 3(k-1)} & P_{xk} \end{bmatrix} \quad (3)$$

4.2. Data association

The measurement update step is performed at the scan level. Thus, the first step is to decide which ones of the stored scans sufficiently overlap with the most recent one. Our proposal is to take this decision based on a proximity, euclidean, criteria, although other distances could be used.

The displacement and rotation from a scan $S_i \in SH_k$, with $i < k$, to S_k can be estimated from the state vector as follows:

$$\delta(i, X_k) = \hat{x}_{i+1,k} \oplus \hat{x}_{i+2,k} \oplus ... \oplus \hat{x}_{k,k} \quad (4)$$

Thus, the euclidean distance $d(i,k)$ from the coordinate frame of S_i to the one of S_k is the norm of the $[x,y]$ components of $\delta(i, X_k)$. The proposal of this paper is, similarly to [10], to select as possible associations those scans in SH_k that are at an euclidean distance below a certain threshold γ. Let the *associated scans* set AS_k be defined as the set of possible associations as follows:

$$AS_k = \{i | d(i,k) < \gamma\} \equiv \{a1, a2, ..., aM\} \quad (5)$$

4.3. The measurement models

In order to measure the displacement and rotation between each of the associated scans, two scan matching techniques are proposed. On the one hand, the *Iterative Closest Point* (ICP) [9] as it is a well known and widely used technique. On the other hand, the *sonar probabilistic Iterative Correspondence* (spIC) because it has shown to be a reliable, stand-alone, scan matching algorithm both using terrestrial ultrasonic range finders [3] and underwater MSIS [4]. The explanation of the spIC and ICP is out of the scope of this paper. The reader is directed to the aforementioned studies to have a full description of the algorithms.

The scan matching (either ICP or spIC) is executed for each S_i with $i \in AS_k$ to estimate the displacement and rotation between S_i and S_k. These scan matching estimates constitute the measurements and will be denoted by z_i.

The observation function h_i is in charge of predicting the spIC measurement corresponding to S_i from the state vector X_k^-. In other words, the observation function estimates the displacement and rotation from S_i to S_k using the state vector. This displacement has been previously computed when performing the data association by means of Equation 4. Thus, $h_i = \delta(i, X_k)$. As this observation function explicitly takes into account the whole chain of motions between each couple of matched scans, the proposed approach is able to correct the whole robot trajectories involved in loops.

The observation matrix H_i is

$$H_i = \left.\frac{\partial h_i}{\partial X_k}\right|_{X_k^-} = \left[\left.\frac{\partial h_i}{\partial x_{1,k}}\right|_{X_k^-} \left.\frac{\partial h_i}{\partial x_{2,k}}\right|_{X_k^-} \cdots \left.\frac{\partial h_i}{\partial x_{k,k}}\right|_{X_k^-} \right] \quad (6)$$

where $x_{i,k}$ denote the state vector components. It is straightforward to see that

$$H_i = \left[0_{3\times 3i} \left.\frac{\partial h_i}{\partial x_{i+1,k}}\right|_{X_k^-} \left.\frac{\partial h_i}{\partial x_{i+2,k}}\right|_{X_k^-} \cdots \left.\frac{\partial h_i}{\partial x_{k,k}}\right|_{X_k^-} \right] \quad (7)$$

By applying the chain rule, the non-zero terms of this Equation are as follows:

$$\begin{aligned} \left.\frac{\partial h_i}{\partial x_{j,k}}\right|_{X_k^-} &= \left.\frac{\partial h_i}{\partial x_{i+1,k} \oplus x_{i+2,k} \oplus ... \oplus x_{j,k}}\right|_{X_k^-} \cdot \\ &\cdot \left.\frac{\partial x_{i+1,k} \oplus x_{i+2,k} \oplus ... \oplus x_{j,k}}{\partial x_{j,k}}\right|_{X_k^-} \quad (8) \end{aligned}$$

According to [6] this can be computed as follows:

$$\left.\frac{\partial h_i}{\partial x_{j,k}}\right|_{X_k^-} = J_{1\oplus}\{g_j, \ominus g_j \oplus h_i\}|_{X_k^-} \cdot \cdot J_{2\oplus}\{g_j \ominus x_{j,k}, x_{j,k}\}|_{X_k^-} \quad (9)$$

where $J_{1\oplus}$ and $J_{2\oplus}$ are the Jacobians of the composition of transformations [13] and $g_j = x_{i+1,k} \oplus x_{i+2,k} \oplus ... \oplus x_{j,k}$. The operator $\ominus$ denotes the inversion of a transformation.

At this point, the measurements z_i coming from the scan matching and the observation function h_i, as well as the observation matrix H_i are available for all $i \in AS_k$. Let us build the measurement vector z considering all the associated scans as follows:

$$z = \begin{bmatrix} z_{a1} \\ z_{a2} \\ ... \\ z_{aM} \end{bmatrix} \quad (10)$$

where $a_1, a_2, ..., a_M$ denote the items in AS_k (see Equation 5). The observation function $h(X_k^-)$ and the observation matrix $H(X_k^-)$ considering all the associated scans are

$$h(X_k^-) = \begin{bmatrix} h_{a1} \\ h_{a2} \\ ... \\ h_{aM} \end{bmatrix} \quad H = \begin{bmatrix} H_{a1} \\ H_{a2} \\ ... \\ H_{aM} \end{bmatrix} \quad (11)$$

where H is used instead of $H(X_k^-)$ to ease notation.

4.4. The EKF approach

By means of z, h and H the SLAM update step can be performed by applying an EKF as follows:

$$K = P_k^- H^T (H P_{k+1}^- H^T + R)^{-1} \quad (12)$$
$$\hat{X}_k^+ = \hat{X}_k^- + K(z - h(X_k^-)) \quad (13)$$
$$P_k^+ = (I - KH)P_k^- \quad (14)$$

where R is a block diagonal matrix containing the scan matching covariances corresponding to the items in z.

4.5. The IEKF approach

The effects of the linearizations in the observation model may be problematic especially when closing large loops. In order to alleviate this problem, our second proposal is not to use an EKF but an IEKF [2]. Roughly speaking, the IEKF consists on iterating an EKF and relinearizing the system at each iteration until convergence is achieved.

Let $\mathbf{X}_j = N(\hat{\mathbf{X}}_j, \mathbf{P}_j)$ denote the state vector estimate provided by the $j-th$ IEKF iteration. At iteration $j+1$ it is updated as follows:

$$\hat{\mathbf{X}}_{j+1} = \hat{\mathbf{X}}_j + \mathbf{P}_j \mathbf{H}_j^T R^{-1}(z - h(\mathbf{X}_0)) - -\mathbf{P}_j \mathbf{P}_0^{-1}(\hat{\mathbf{X}}_j - \hat{\mathbf{X}}_0) \quad (15)$$

where $\mathbf{H}_j$ denotes the observation matrix H evaluated at $\hat{\mathbf{X}}_j$ (i.e. each H_i in Equation 6 is evaluated at the value of the state vector in the previous IEKF iteration). The terms $\mathbf{X}_0$ and $\mathbf{P}_0$ denote the state vector and its covariance before starting the IEKF. The state vector covariance is updated by the IEKF by the following expression:

$$\mathbf{P}_j = \mathbf{P}_0 - \mathbf{P}_0 \mathbf{H}_j^T (\mathbf{H}_j \mathbf{P}_0 \mathbf{H}_j^T + R)^{-1} \mathbf{H}_j \mathbf{P}_0 \quad (16)$$

When the IEKF achieves convergence, the obtained state vector constitutes the X_k^+.

4.6. Some considerations

It is important to emphasize that both the EKF and the IEKF only update the items in the state vector involved in the detected loops. Thus, the matrix H in Equation 11 could be reduced by removing all the zero valued columns on the left side of the matrix and then updating only the part of the state vector involved in all the detected loops.

Moreover, the presented update step makes it possible to store different loops when they are detected and close them later simultaneously, not necessarily at each SLAM step. Thanks to this, the loop closure can be delayed if the computational resources are not available at a certain time step. Also, the overall computational cost is reduced because, prior to the loop closing, the newly gathered scans are independent and those parts of the covariance matrix related to the new scans are block diagonal, thus making it possible the use of algorithms operating on sparse matrices.

5. Experimental results

The experimental data used to validate our underwater SLAM approach was obtained by [12] in an abandoned marina situated near St. Pere Pescador in the Costa Brava (Spain). A satellite view of this environment is available in [7]. The Ictineu AUV was teleoperated along a 600m trajectory at an average speed of 0.2m/s. The trajectory includes a small loop as well as a 200m long straight path. The gathered data included measurements from the DVL, the MRU and the MSIS. Additionally, a buoy with a GPS was attached to the robot in order to obtain the ground truth.

Figure 3-a shows the trajectories provided by dead reckoning (DVL+MRU) and the GPS. Also, the sonar readings are plotted according to the dead reckoning trajectory for visual inspection. The problems of dead reckoning can be appreciated. For example, the entrance to the canal is misaligned (i.e. the loop is not closed) due to the drift error. The absolute errors corresponding to dead reckoning are shown in Figure 3-b. The represented error corresponds to the distance from each dead reckoning estimate to the corresponding GPS ground truth.

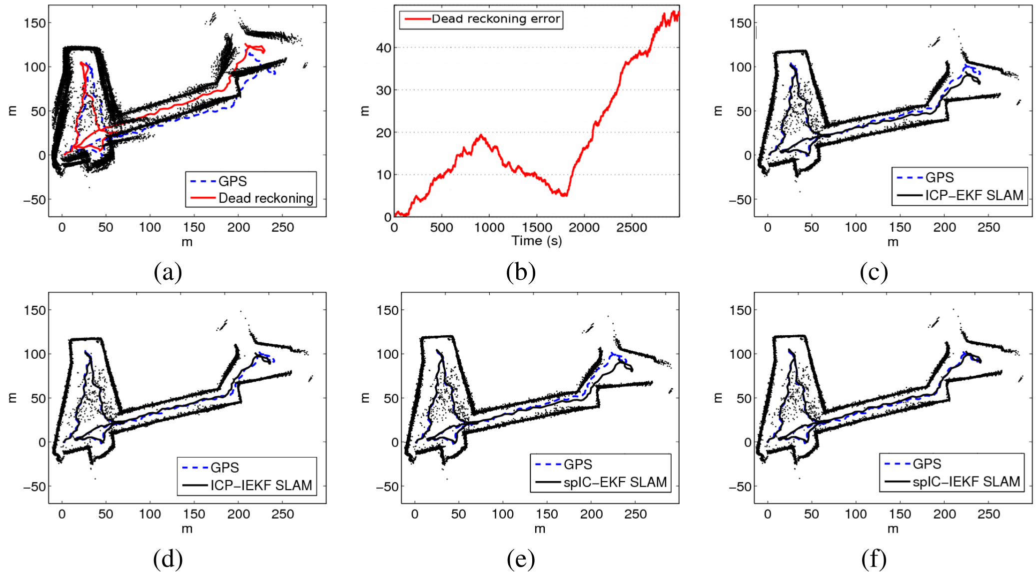


Figure 3. (a) Trajectories according to dead reckoning and GPS. Sonar readings positioned according to dead reckoning. (b) The dead reckoning error. (c) to (f) The different SLAM approaches discussed in the paper.

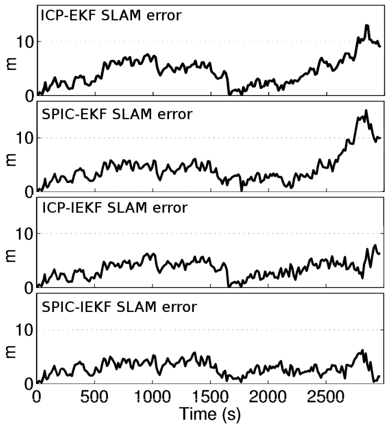


Figure 4. The SLAM error

In this paper, two measurement models based on ICP and spIC have been proposed. Also, two approaches to perform the update have been presented. One of them is based on an EKF while the other relies on a IEKF. The aim of this section is to experimentally evaluate the four resulting proposals (ICP-EKF, spIC-EKF, ICP-IEKF and spIC-IEKF). The resulting trajectories and maps are shown from Figure 3-c to Figure 3-f. It can be observed that the four approaches provide an important improvement with respect to the original data, especially at the entrance of the canal, where the loop has been closed in all cases. It can also be appreciated that the IEKF approaches (Figures 3-d and 3-f) provide better results than the EKF ones (Figures 3-c and 3-e) at the end of the trajectory, after more than 200m of an almost straight trajectory with no loops.

The SLAM absolute errors are shown in Figure 4. It can be observed that the IEKF approaches, overall, produces lower errors than their EKF counterparts. Regarding the differences between the ICP and the spIC measurement models, although the spIC with EKF provides large error than the ICP with EKF at the very end of the trajectory, the overall error when using spIC is below the one produced by ICP.

	Mean	Std. dev	Max
Dead reckoning	18.32m	13.64m	49.03m
ICP-EKF	4.48m	2.58m	13.01m
spIC-EKF	4.34m	2.82m	15.10m
ICP-IEKF	3.44m	1.55m	7.84m
spIC-IEKF	2.94m	1.27m	6.26m

Table 1. Mean, standard deviation and maximum errors corresponding to dead reckoning and the four proposed methods.

Table 1 summarizes the previous results. The first thing to be noticed is the important error reduction of all the methods with respect to dead reckoning. It can also be observed that, although spIC-EKF has a larger maximum error, the mean error is below the one of ICP-EKF. It is also clear that the use of an IEKF update produces an important error reduction. Thus, if one takes into account the resulting error, the use of an IEKF is the best choice and, regarding the measurement model, the spIC provides important benefits with respect to ICP.

Nevertheless, IEKF and spIC demand more computations than EKF and ICP. Thus, the computation time is also a factor to be considered. Table 2 summarizes the computation time for each method measured for a Matlab

	Time	CPU usage
ICP-EKF	572.32s	17.87%
spIC-EKF	772.91s	24.13%
ICP-IEKF	1113.1s	34.76%
spIC-IEKF	1310s	40.9%

Table 2. Execution times (Matlab) including beam segmentation and scan building.

7 R14 implementation running on a Intel Core 2 Duo at 2.4GHz. These times include the SLAM process, as well as the beam segmentation and scan building processes, which are also time consuming (474.07 s in all cases). As the whole mission took more than 53 minutes, the column CPU usage reports the fraction of the mission time actually used in SLAM-related computations.

It can be observed that the use of IEKF is significantly CPU demanding with respect to the standard EKF. This difference is more important than the additional time required for spIC with respect to ICP. However, even the non-optimized Matlab implementation is able to run on real time, and the overall CPU usage for a 600m long trajectory is, in the worst case, below the 50%.

6. Conclusion

This paper presents a novel approach to perform SLAM in underwater environments using a MSIS. To achieve this goal, the sonar data has to be segmented in order to obtain range information from the acoustic profiles. Also, the obtained ranges have to be grouped to build scans while taking into account the robot motion. The rough motion estimates used to compensate the motion induced distortions comes from an EKF that fuses the information produced by a DVL and a MRU.

The SLAM process is performed at the scan level. The estimated position of each newly gathered scan is used to augment the state vector. Moreover, this estimated position is stored relative to the previously gathered scan. In this way, the state vector holds information about the robot trajectory. The measurement model is based on matching each new scan against the previously obtained ones. If a reliable matching is detected, the relative information stored in the state vector makes it possible to explicitly correct the robot trajectory involved in the loop closure.

Two matching techniques (ICP and spIC) have been proposed to build the measurement model. Also, two possible ways to perform the SLAM update (EKF and IEKF) have been presented. The experimental results show that major improvements appear when using the IEKF approach. Also, the use of spIC as a measurement model provides significant benefits with respect to ICP. The computation times have also been measured and they show that, even a non optimized Matlab implementation can be executed in real time for the presented 600m long trajectory.

Acknowledgements

This work is partially supported by DPI 2008-06548-C03-02, FEDER funding and FP7-TRIDENT Project SCP8-GA-2010-248497. The authors are grateful to the members of the Underwater Robotics Lab, in Universitat de Girona, for the data sets and the useful advice.

References

[1] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): part II. *IEEE Robotics and Automation Magazine*, 13(3):108–117, September 2006.

[2] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley and Sons, Inc., 2001.

[3] A. Burguera, Y. González, and G. Oliver. A probabilistic framework for sonar scan matching localization. *Advanced Robotics*, 22(11):1223–1241, 2008.

[4] A. Burguera, Y. González, and G. Oliver. Underwater scan matching using a mechanical scanned imaging sonar. In *Submitted to Proceedings of Intelligent Autonomous Vehicles*, 2010.

[5] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping (SLAM): part I. *IEEE Robotics and Automation Magazine*, 13(2):99–110, June 2006.

[6] C. Estrada, J. Neira, and J. D. Tardós. Hierarchical SLAM: real-time accurate mapping of large environments. *IEEE Transactions on Robotics*, 21(4):588–596, August 2005.

[7] Google Maps. St. Pere Pescador, Abandoned Marina. `http://maps.google.es/maps?hl=es&ie=UTF8&ll=42.202645,3.10662&spn=0.005182,0.008862&t=h&z=17`. Accessed 09-March-2010.

[8] E. Hernández, P. Ridao, D. Ribas, and J. Batlle. MSISpIC: A probabilistic scan matching algorithm using a mechanical scanned imaging sonar. *Journal of Physical Agents*, 3(1):3–11, 2009.

[9] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *Journal of Intelligent and Robotic Systems*, 18(3):249–275, March 1997.

[10] A. Mallios, P. Ridao, E. Hernandez, D. Ribas, F. Maurelli, and Y. Petillot. Pose-based slam with probabilistic scan matching algorithm using a mechanical scanned imaging sonar. In *Proceedings of Oceans 2009-Europe*, pages 1–6, May 2009.

[11] J. Nieto and T. Bailey. Recursive scan matching SLAM. *Robotics and Autonomous Systems*, 55(1):39–49, Jannuary 2007.

[12] D. Ribas, P. Ridao, J. Tardós, and J. Neira. Underwater SLAM in man made structured environments. *Journal of Field Robotics*, 11-12(25):898–921, 2008.

[13] R. Smith, P. Cheeseman, and M. Self. A stochastic map for uncertain spatial relationships. In *Proceedings of International Symposium on Robotic Research, MIT Press*, pages 467–474, 1987.

[14] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard. Robust mapping and localization in indoor environments using sonar data. *International Journal of Robotics Research*, 21(4):311–330, April 2002.

[15] S. Williams, P. Newman, G. Dissanayake, and H. Durrant-Whyte. Autonomous underwater simultaneous localisation and map building. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2000.