TOWARDS A SLAM SOLUTION FOR A ROBOTIC AIRSHIP

Cesar Castro, Samuel Bueno
Computer Vision and Robotics Division, CenPRA
Rod. SP 65, km 143.6, 13069-901, Campinas-SP, Brazil

Alessandro Victorino
INRIA-Sophia Antipolis, ICARE project
2004, Route des Lucioles, 06902, Sophia Antipolis , France

Keywords: SLAM, localization, mapping, Kalman filtering, UAVs.

Abstract: This article presents the authors ongoing work towards a six degrees of freedom simultaneous localization and mapping (SLAM) solution for the Project AURORA autonomous robotic airship. While the vehicle's mission is being executed in an unknown environment, where neither predefined maps nor satellite help are available, the airship has to use nothing but its own onboard sensors to capture information from its surroundings and from itself, locating itself and building a map of the environment it navigates. To achieve this goal, the airship sensorial input is provided by an inertial measurement unit (IMU), whereas a single onboard camera detects features of interest in the environment, such as landmark information. The data from both sensors are then fused using an architecture based on an extended Kalman filter, which acts as an estimator of the robot pose and the map. The proposed methodology is validated in a simulation environment, composed of virtual sensors and the aerial platform simulator of the AURORA project based on a realistic dynamic model. The results are hereby reported.

1 INTRODUCTION

Beside their use as military surveillance platforms, Unmanned Aerial Vehicles (UAVs) have a wide spectrum of potential civilian observation and data acquisition applications. Usually, unmanned aerial platforms are airplanes, helicopters or, more recently, airships - also known as lighter-than-air (LTA) vehicles.

Unmanned airships have extended airborne capabilities and large payload to weight ratio. They are therefore best suited for low altitude, low speed or hovering airborne data gathering missions, where the vehicle should also take-off and land vertically (Elfes et al., 1998). In this context, Project AURORA – Autonomous Unmanned Remote mOnitoring Robotic Airship – was proposed, aiming the gradual establishment of the technologies required for autonomous operation of unmanned robotic airships (Elfes et al., 1998; Bueno et al., 2002). Presently, AURORA LTA robotic platform consists of (de Paiva et al., 2006): i) a 10.5 m long, 34 m^3 of volume and 10 Kg payload capacity nonrigid airship; ii) an onboard system containing a PC/104 CPU with RT Linux, sensors and actuators; iii) a portable PC with RT Linux ground station for system operation; iv) a communication system comprising radio modems, analogical and short range wireless Ethernet video links. The onboard sensor package mainly includes: a GPS, an Inertial Measurement Unit (IMU), a single firewire IEEE 1394 digital camera and a Wind sensor. Experimental results achieved so far include path following flight through a set of pre-defined way points under lateral and longitudinal control; experimental validation of visual servoing line following strategies are under way.

Other important research efforts on robotic airships are described in (Kantor et al., 2001; Wimmer et al., 2002; Hygounenc et al., 2004) and, considering Planetary Exploration missions, in (Elfes et al., 2003). Unmanned LTA vehicles are also suited for stratospheric HALE (High Altitude Long Endurance) operation, and efforts in this sense have been carried out in Japan, Korea, Europe and USA, involving industry and academia.

For UAVs in general, and for AURORA in particular, an important aspect of the aircraft embedded system is to determine the aerial platform position and orientation in space. To do so, the most common approach is to use an *inertial navigation system* (INS) or external navigational aids (such as a GPS). However,

Castro C., Bueno S. and Victorino A. (2006).
TOWARDS A SLAM SOLUTION FOR A ROBOTIC AIRSHIP.
In *Proceedings of the Third International Conference on Informatics in Control, Automation and Robotics*, pages 241-248
DOI: 10.5220/0001213202410248

Figure 1: The AURORA project robotic aerial platform.

inertial systems diverge over time and GPS information might not always be available. Besides, neither inertial navigation nor satellite systems provide information about the surroundings of the airship, which means localization with respect to the environment is not possible. So, more important than finding the pose of the vehicle is to do so autonomously, using nothing but its embedded sensors, with no external sensors, no predefined maps or satellite help. Furthermore, localization in an unknown environment implies not only pose determination but also incremental map building. The process in which a mobile robot constructs a map and, while doing it, locates itself in it, is called *Simultaneous Localization And Mapping* (SLAM).

Robotic mapping started with approaches based on *occupancy grids* (Elfes, 1987; Elfes, 1989), but it was only with the work of Smith, Self and Cheeseman (Smith et al., 1987; Smith et al., 1988) *simultaneous localization and mapping* was first formulated as a probabilistic estimation problem. Since then, interest in SLAM increased and several techniques were developed, taking the extended Kalman filter (EKF) approach further (Castellanos et al., 1999; Castellanos et al., 2001). From the indoors robotics research community, these solutions were then taken to open spaces, where the size of the environment and of the map became a concern (Sukkarieh et al., 1999; Guivant and Nebot, 2001; Bailey, 2002; Guivant and Nebot, 2003). Finally, it reached aerial robots. Some interesting solutions to the SLAM problem have been posed at the international aerial robotics community, consisting of directly correcting the IMU readings with the aid of a GPS (Panzieri et al., 2002) and/or updating the prediction by integrating the inertial navigation data with observations of the environment (made by an exteroceptive sensor such as a range finder or a camera). The aerial SLAM problem has received relevant attention from the robotic community over the last years (Ruiz et al., 2001; Kim, 2004; Langelaan and Rock, 2005; Wu et al., 2005) but, even though many advances and partial solutions have been proposed, it remains an open problem when related to autonomous aerial navigation in unknown and unstructured environments.

Considering the present status of Project AURORA, in this article we present a first effort towards the establishment of a 3D SLAM solution, with six degrees of freedom (DoF), specifically aiming to enhance the autonomous capability of the robotic airship. An IMU is used in an inertial navigation model to calculate the displacement of the airship during a time interval, while a vision system, in which a single onboard camera is used to capture landmark information from the environment, allows the computation of the map and of the airship pose with respect to it. An EKF is used to fuse all the information and calculate the uncertainty of the estimation.

After this introductory section, the remaining parts of this article are organized as follows. Section 2 describes the methodology used to achieve SLAM through an extended Kalman filter. Section 3 shows the simulation results of the methodology. Finally, Section 4 stresses the concluding remarks and the future work concerning the implementation of such methodology on the real airship and its evolution.

2 SYSTEM DESCRIPTION

In this section, the mathematical formulation of the problem is derived. The Kalman filtering approach to sensor fusion and autonomous localization and mapping is presented, along with the equations that define the airship movement and the observation model, which will lead to feature matching and state augmentation.

2.1 SLAM Architecture

Suppose that the airship is placed in an unknown location, with no prior information about its surroundings. It is equipped with an IMU, which provides roll, pitch and yaw velocities as well as linear accelerations in the body reference frame, and a single camera, which can be used to take relative measurements between features in the environment (their absolute locations are not known) and the camera itself. Usually, the IMU readings are fed to the *inertial navigation system* (INS) of the airship that computes the relative displacement of the vehicle in a geodesic reference frame. In the present work, as the vehicle moves around, features are observed by the camera and fused to the IMU data in order to build a map and minimize localization errors of both vehicle and map features.

The INS equations are inserted in the vehicle kinematic model.

An extended Kalman filter (EKF) is used as a state estimator for both the vehicle location and the map, propagating first-order uncertainty, in an approach known as Stochastic Mapping(Smith et al., 1987; Smith et al., 1988). For a map with N features, the system state vector $\hat{\mathbf{x}}$ is defined as the estimate of the vehicle state vector $\hat{\mathbf{x}}_v$ augmented with the map state vector $\hat{\mathbf{x}}_m$, so that

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{\mathbf{x}}_v \\ \hat{\mathbf{x}}_{m1} \\ \vdots \\ \hat{\mathbf{x}}_{mN} \end{bmatrix}. \tag{1}$$

The state vector is accompanied by a covariance matrix $\mathbf{P}$, which represents the uncertainty in the state vector, defined as

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{v,v} & \mathbf{P}_{v,m1} & \dots & \mathbf{P}_{v,mN} \\ \mathbf{P}_{m1,v} & \mathbf{P}_{m1,m1} & \dots & \mathbf{P}_{m1,mN} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{mN,v} & \mathbf{P}_{mN,m1} & \dots & \mathbf{P}_{mN,mN} \end{bmatrix}. \tag{2}$$

In the following sections, the kinematic model of the airship using the IMU inputs is derived, along with the observation model. In the estimation process, the kinematic model is used to predict the airship pose and estimate the increase in the pose uncertainty. The observation model is used to update the previous prediction and reduce the uncertainty of the estimation based in the observations.

2.2 Kinematic Model

The vehicle state vector is defined as

$$\hat{\mathbf{x}}_v = \begin{bmatrix} \hat{\mathbf{p}} \\ \hat{\mathbf{v}} \\ \hat{\mathbf{q}} \end{bmatrix} \tag{3}$$

where vectors $\hat{\mathbf{p}}$ and $\hat{\mathbf{v}}$ are, respectively, the position and the velocity of the airship in earth reference. The orientation of the robot is represented by the quaternion vector $\hat{\mathbf{q}}$. Although it's a non-mimimal representation of orientation, a quaternion is used in order to avoid linearization problems and singularities of other representations.

The prediction of the robot state $\hat{\mathbf{x}}(k|k-1)$ is obtained from its last pose and the signals provided by the IMU that constitute the input $\mathbf{u}(k)$ of the inertial navigation model, composed of: $\mathbf{f}$, the linear acceleration, and ω, the angular velocity, both in frame of reference fixed in the airship:

$$\begin{aligned} \hat{\mathbf{x}}_v(k|k-1) &= f(\hat{\mathbf{x}}_v(k), \mathbf{u}(k), \mathbf{w}_v(k)) \\ &= \begin{bmatrix} \hat{\mathbf{p}}(k-1) \\ \hat{\mathbf{v}}(k-1) \\ \hat{\mathbf{q}}(k-1) \end{bmatrix} + \begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{v} \\ \Delta\mathbf{q} \end{bmatrix} \\ &\quad + \begin{bmatrix} \mathbf{w_p}(k) \\ \mathbf{w_v}(k) \\ \mathbf{w_q}(k) \end{bmatrix}, \end{aligned} \tag{4}$$

where $\mathbf{w}(k)$ represents uncorrelated, zero-mean process noise errors with covariance $\mathbf{Q}(k)$ and

$$\begin{bmatrix} \Delta\mathbf{p} \\ \Delta\mathbf{v} \\ \Delta\mathbf{q} \end{bmatrix} = \begin{bmatrix} \mathbf{v}(k-1)t + \mathbf{C}_B^W\,\mathbf{f}\,\frac{t^2}{2} \\ \mathbf{C}_B^W\,\mathbf{f}\,t \\ (\frac{1}{2}\,\mathbf{D}\,\omega + \varepsilon\,\mathbf{q}(k-1))\,t \end{bmatrix}, \tag{5}$$

$$\mathbf{D} = \begin{bmatrix} q_i & q_j & q_k \\ -q_r & q_k & -q_j \\ -q_k & -q_r & q_i \\ q_j & -q_i & -q_r \end{bmatrix}, \tag{6}$$

$$\varepsilon = 1 - \|\mathbf{q}(k-1)\|^2. \tag{7}$$

It is important to notice that $\mathbf{C}_B^W$ is the direction cosine matrix obtained from $\hat{\mathbf{q}}(k-1)$, which takes from the airship reference frame (B) orientation to the earth reference frame (W) orientation, $\mathbf{D}$ is also computed from $\hat{\mathbf{q}}(k-1)$, and ε represents an integration correction gain, which comes into place if the norm of the quaternion is different than one.

2.3 Observation Model

It is important to investigate how the observations of the landmarks relate to its actual position. In this work, it is considered that the visual sensor returns to the system the relative position between camera and landmark, as shown in figure 2.

Each map feature is represented by its cartesian threedimensional coordinates in the earth-fixed frame W, while the observations consist of the cartesian threedimensional coordinates of the features in the camera-fixed frame C. Considering the camera frame to be a known rotation of the airship body frame B, it is possible to specify each a map feature as

$$\begin{aligned} \mathbf{x}_{mi} &= g(\mathbf{x}_v, \mathbf{z}_i) \\ &= \mathbf{p}(k) + R_B^W(R_C^B(\mathbf{z}_i)), \end{aligned} \tag{8}$$

where $R_C^B(\cdot)$ is a function that uses a quaternion to rotate from the camera orientation to the body orientation (the airship orientation) and $R_B^W(\cdot)$ rotates a vector from the body frame to the world frame.

Taking the inverse of the previous equation, the observations at any instant can be predicted from the state vector with the resulting observation model:

$$\begin{aligned} \hat{\mathbf{z}}_i &= h(\mathbf{x}_v, \mathbf{x}_{mi}) + \mathbf{v}_i(k) \\ &= g^{-1}(\mathbf{x}_v, \mathbf{x}_{mi}) + \mathbf{v}_i(k) \\ &= R_B^C(R_W^B(\mathbf{x}_{mi} - \mathbf{p})) + \mathbf{v}_i(k), \end{aligned} \tag{9}$$

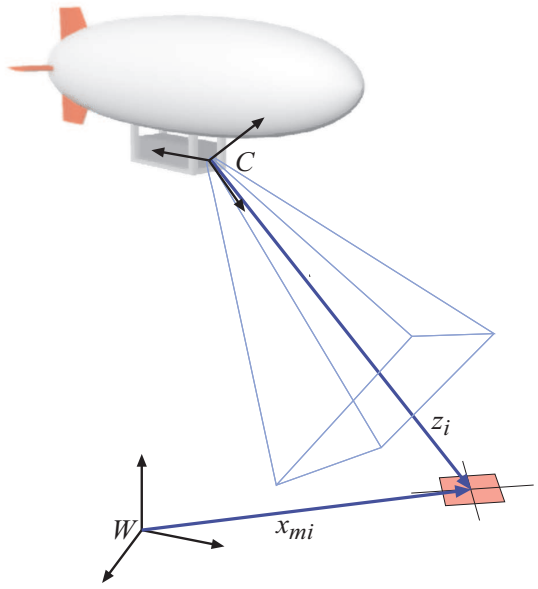

Figure 2: The airship observing a landmark.

with the rotation functions following the same notation already specified and $\mathbf{v}_i(k)$ being uncorrelated, zero mean observation noise with covariance $\mathbf{R}(k)$.

2.4 Estimation Process

The general state estimation process using the extended Kalman filter is represented in Fig. 3. At the kth step, the kinematic model of the robot is used, along with the inputs from the IMU, to compute a prediction $\hat{\mathbf{x}}(k|k-1)$ and the covariance $\mathbf{P}(k|k-1)$. Because the IMU readings do not contain any map information, the uncertainty in the inputs is reflected in the vehicle state uncertainty, which should increase. An observation prediction $h(\hat{\mathbf{x}}(k|k-1))$ is then calculated and compared to the real observation $\mathbf{z}(k)$, obtained from the vision system. From that comparison, the observations are divided into matched and unmatched observations. The first are used to generate the estimate $\hat{\mathbf{x}}(k|k)$ and $\mathbf{P}(k|k)$ (the uncertainty should decrease, since the pose with respect to the map can be corrected), while the second are used to augment the state vector and the associated covariance matrix with new map entrances.

2.4.1 Prediction

The prediction of the state vector is obtained directly from (4), derived in section 2.2. Because the map is considered static with respect to the earth frame, the map transition equation is

$$\hat{\mathbf{x}}_m(k|k-1) = \hat{\mathbf{x}}_m(k-1), \qquad (10)$$

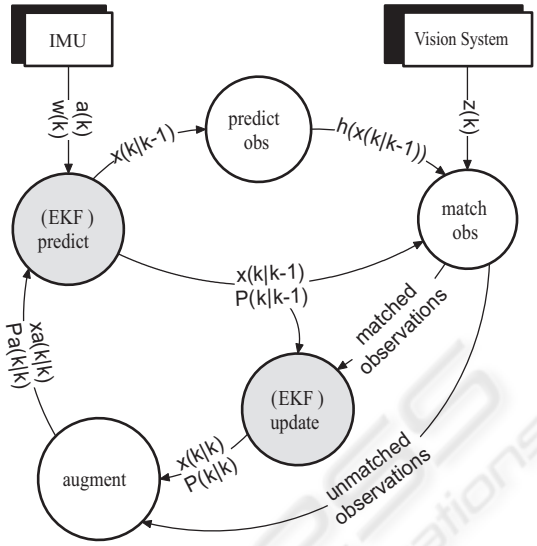

Figure 3: Structure of the SLAM system.

which leads to the full state transition equation:

$$\begin{bmatrix} \hat{\mathbf{x}}_v(k|k-1) \\ \hat{\mathbf{x}}_m(k|k-1) \end{bmatrix} = \begin{bmatrix} f(\hat{\mathbf{x}}_v(k-1), \mathbf{u}(k), \mathbf{0}) \\ \hat{\mathbf{x}}_m(k-1) \end{bmatrix} \qquad (11)$$

However, as it was already mentioned, the prediction implies an increase of the airship pose uncertainty, since the IMU readings contain errors. Therefore, the covariance matrix must be recalculated to reflect that change. This is done through the following equation:

$$\begin{aligned} \mathbf{P}(k|k-1) \quad = \quad & \nabla f_x \, \mathbf{P}(k-1) \, \nabla f_x^T \\ & + \nabla f_u \, \mathbf{Q}(k) \, \nabla f_u^T, \end{aligned} \qquad (12)$$

in which ∇f_x and ∇f_u are the jacobians of f with respect to the state vector and the input, respectively. The diagonal matrix $\mathbf{Q}(k)$ represents a model of the error covariance in the sensor readings.

2.4.2 Update

In order to update the prediction $\mathbf{x}(k|k-1)$, using the landmarks perceived by the vision system, the equations derived in section 2.3 are used. From (9), it is possible to calculate the uncertainty in the observation prediction at instant k, $\mathbf{S}(k)$, which is related to the system state and the observation errors:

$$\mathbf{S}(k) = \nabla h_{\mathbf{x}}(k) \mathbf{P}(k|k-1) \nabla h_{\mathbf{x}}^T(k) + \mathbf{R}(k), \qquad (13)$$

where $\nabla h_{\mathbf{x}}(k)$ is the Jacobian of $h(\mathbf{x})$ with respect to $\hat{\mathbf{x}}(k|k-1)$ and $\mathbf{R}(k)$ is the sensor uncertainty, modeled as a diagonal matrix.

From (9) and (13), the observations are associated to their predictions in observation space. A statistical

validation gate, the *Mahalanobis distance*, is used to match observations. This distance γ is computed by

$$\gamma_{i,j} = \nu_{i,j}^T \, \mathbf{S}_j^{-1}(k) \, \nu_{i,j}, \tag{14}$$

where $\mathbf{S}_j(k)$ is the covariance related to the jth prediction and $\nu_{i,j}$ is the innovation vector, defined as

$$\nu_{i,j} = \mathbf{z}_i - h(\hat{\mathbf{x}}_v, \hat{\mathbf{x}}_{mj}). \tag{15}$$

If any prediction or observation is matched twice, it is necessary to resolve the ambiguity. The method used in the present work is the *nearest neighbors* data association, in which the observation is matched against the prediction with the smallest value $N_{i,j}$,

$$N_{i,j} = \gamma_{i,j} + \ln |\mathbf{S}_j(k)|. \tag{16}$$

If any match is found, not only the state is updated but the uncertainty in both airship position and map can be reduced, according to the following:

$$\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{W}(k)\nu(k), \tag{17}$$

$$\mathbf{P}(k) = \mathbf{P}(k|k-1) - \mathbf{W}(k)\mathbf{S}(k)\mathbf{W}(k)^T, \tag{18}$$

where $\nu(k)$ is calculated from the union of the matched predictions and observations, the gain matrix $\mathbf{W}(k)$ is obtained from

$$\mathbf{W}(k) = \mathbf{P}(k|k-1)\nabla h_{\mathbf{x}}^T(k)\mathbf{S}^{-1}(k) \tag{19}$$

and $\mathbf{S}(k)$ is computed as in (13).

2.4.3 State Augmentation

The unmatched features must be included in the map, in order for them to be used in the update stage the next time they are observed. The process of incrementally building the map is known as state augmentation and is accomplished by the following equation:

$$\mathbf{x}_{aug}(k) = f_a(\mathbf{x}(k)) = \begin{bmatrix} \mathbf{x}_v(k) \\ \mathbf{x}_m(k) \\ g(\mathbf{x}_v(k), \mathbf{z}(k)) \end{bmatrix}, \tag{20}$$

with $g(\cdot,\cdot)$ defined in (8). The newly added map feature is highly correlated to the vehicle state, due to the fact that position and orientation uncertainty directly affect the new landmark uncertainty. To accommodate that situation, the matrix $\mathbf{P}(k)$ must change accordingly:

$$\mathbf{P}_{aug}(k) = \nabla f_a \begin{bmatrix} \mathbf{P}(k) & \mathbf{0} \\ \mathbf{0} & \mathbf{R}(k) \end{bmatrix} \nabla f_a^T \tag{21}$$

With these equations, the cycle shown in figure 3 was fully described.

3 RESULTS

In this section, the described methodology is validated in simulation. The simulation setup and the performance measurements are presented, as well as an analysis of the results.

3.1 Simulation Setup

The SLAM implementation and studies reported here were done in a simulation environment for the AS800 airship (Gomes and Ramos, 1998; de Paiva et al., 1999). In the simulator, the airship 6DoF dynamics are modeled according to

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{F}_d(\mathbf{v}) + \mathbf{F}_a(\mathbf{v}) + \mathbf{F}_p + \mathbf{F}_g, \tag{22}$$

where $\mathbf{M}$ is the mass matrix which includes both the actual inertia of the airship and the virtual inertia elements associated with the dynamics of buoyant vehicles; $\mathbf{v} = [\,u,\, v,\, w,\, p,\, q,\, r\,]^T$ is the vector of the airship linear and angular velocities; $\mathbf{F}_d$ is the dynamics vector containing the Coriolis and centrifugal force terms, and also the wind-induced forces (Azinheira et al., 2001); $\mathbf{F}_a$ is the vector of aerodynamic forces and moments; $\mathbf{F}_p$ is the vector of propulsion forces and moments, and $\mathbf{F}_g$ represents the gravity forces and moments.

In order to test the SLAM scheme in different conditions during the simulated flight, the airship is exposed to mild wind and turbulence, also experiencing different accelerations and turns. Lateral and longitudinal control schemes are used to make the airship follow a trajectory path defined as a sequence of straight lines between a set of pre-defined points in latitude, longitude and altitude. The lateral heading/guidance control uses a PID controller as described in (Azinheira et al., 2000). The longitudinal control involves an altitude state feedback controller with a proportional plus derivative (PD) action, and a simple proportional velocity controller (Bueno et al., 2002).

Virtual implementations of an IMU and a vision system were developed so that the SLAM algorithms could be effectively tested. The virtual IMU consists simply of a module that collects the IMU perfect readings, which are used to describe the error free path of the airship, and adds a noise with the selected distribution. The virtual vision system includes the simulation of a camera, comprising focus, aspect and the direction axes (which can be modified to reflect mechanical imperfections). With these parameters, images of the environment can be rendered in a visualization frontend, as shown in Fig. 4. Also, it calculates the observations based in the environment mathematical representation, emulating an algorithm that can return the relative position of the map landmarks with respect to the camera. To this relative position, it adds an error with known distribution to reflect other noise sources.

Before starting the simulation, the airship is placed at an arbitrary position in the virtual 3D world and the initial estimate is set at the same pose. Because the estimate and the map are made relative to the initial setup of the system, the divergence between the real

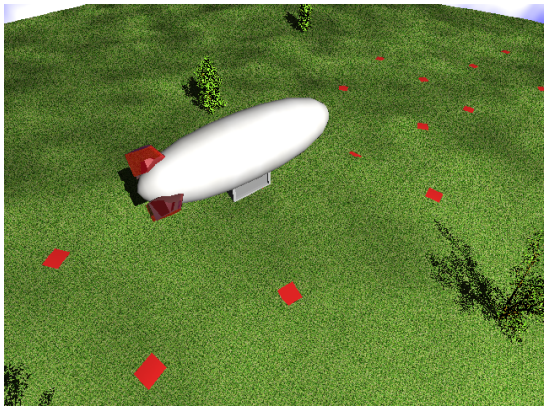

Figure 4: Renderization of the virtual environment, displaying the airship in flight and the landmarks on the field.

state of the system and its estimate will directly show the estimation error.

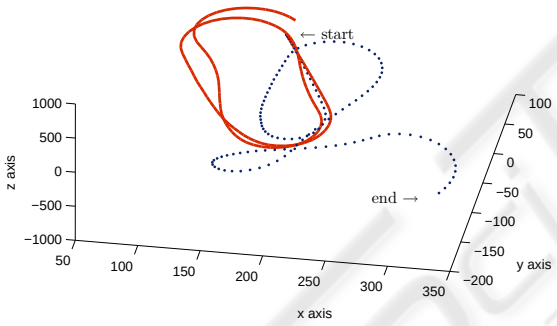

Figure 5: Comparison of the real trajectory (solid line) with the one obtained from the IMU alone (dotted line). The estimation diverges unconstrained.

As in the real onboard airship infrastructure (Bueno et al., 2002; de Paiva et al., 2006), in the simulation environment, new information from the virtual IMU and vision system are provided every $0.05\,seconds$ ($20\,Hz$). The IMU readings contain an error with distribution $N(-0.03,\ 0.04)$ in m/s^2 in the linear acceleration $\mathbf{f}$ and $N(0.01,\ 0.0025)$ rad/s in the angular velocity ω, which means that both of them lead to a serious drift that, if not corrected, accumulates over time, generating ever increasing errors, as it is seen in Fig. 5. The vision system makes observations of landmarks (which are unknown to the vehicle system) and detects their relative position with an additional error with zero mean and a variance of $2.25\,m$ in each of the coordinates $(x_i^b,\ y_i^b,\ z_i^b)$.

3.2 SLAM System Performance

In what follows, localization and mapping results of the proposed methodology are presented and discussed.

In terms of localization, Fig. 6 compares the path followed by the airship during the experiment and the trajectory computed by using the SLAM scheme. The solid line represents the real position of the airship, while the dotted line represents the estimate of the airship position using the SLAM-EKF methodology. In contrast to the IMU-only estimate shown in Fig. 5, the estimation error does not grow indefinitely and is always small, considering the errors in both the IMU and the vision system.

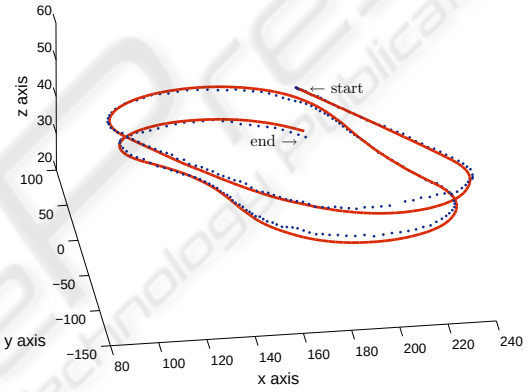

Figure 6: Comparison of the real trajectory (solid line) with its estimate, using the SLAM methodology (dotted line).

The generated map associated to the previous trajectory is shown in Fig. 7. The circles represent the position of the real landmarks, while the crosses show the map estimate. As in the trajectory, the errors are small. The mean distance from the estimates to the real landmarks is 2.26 meters and the maximum distance is 4.59 m.

The error norm of the vehicle state, separated in position, velocity and orientation, is shown in Fig. 8, and they demonstrate the overall SLAM convergence. To understand the behavior of these errors during some points of the flight, it is necessary to analyze the number of feature matches at each observation and the covariance of the map features. With less than three matched features, the update stage will not respond properly, as a unique position of the airship with respect to the matched features cannot be calculated. Also, when there are too many recently discovered features being observed, their covariance is high and the gain matrix $\mathbf{W}$ will not favor the update, which means very quick turns and high speeds will stop the system from working properly.

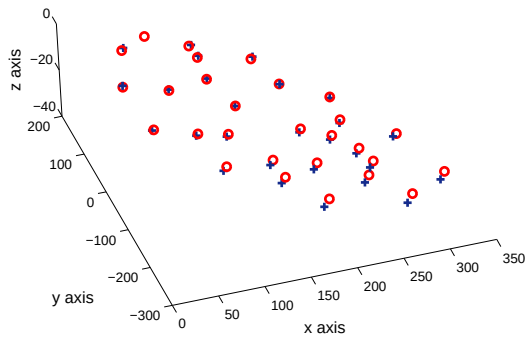

Figure 7: Real map, represented by circles, and the estimated map, represented by crosses.

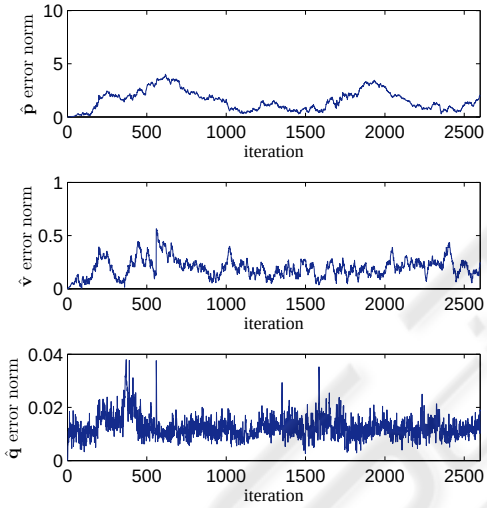

Figure 8: On top is the position error norm. In the middle, it is shown the velocity error norm. Bellow, is the orientation quaternion error norm.

It is also important to point out that, even in ideal conditions, the estimation system is always underdetermined. If N features are matched, the dimension of the 6Dof SLAM system is $9+3N$, while the observations can provide rank $6+3N$. This, however, does not pose a problem to relative positioning and mapping. As the system converges, the errors related to the landmarks will tend to zero, the airship pose with respect to the map will be updated to its best and the position of the landmarks will stop being corrected.

4 CONCLUSION

In this article, a methodology to estimate the AURORA aerial platform position and attitude while incrementally building a map of the environment was derived. A strategy to fuse the data provided by the airship IMU and the vision system – which detects the relative position between the landmarks and the camera – using an extended Kalman filter was presented. The validation of the proposed methodology was performed using a realistic simulation environment that includes the airship dynamic model and automatic trajectory control used in real flights, a set of virtual sensors and a visualization frontend.

The obtained results demonstrate the correctness of the SLAM approach and constitute an important step in the Project AURORA effort towards the incremental evolution of robotic airships. Considering the experimental validation objectives, the SLAM methodology hereby presented will be improved and integrated to the airship embedded system for subsequent experimental validation in real flights. Considering this objective, further work comprises: i) concluding the SLAM experiments with the real airship, detecting simple artificial landmarks placed in the environment; ii) computational performance optimizations, aiming for larger maps and the simultaneous observation of many landmarks; iii) a bearing-only vision system, comprising a landmark tracking algorithm and an efficient feature initialization approach, which will allow faster computation of the vehicle and map state update (Davison, 2003); iv) a visual odometry model where the relative motion of the airship is recovered by decomposing the Euclidean homography, established from projective geometry relationships (Silveira et al., 2006).

ACKNOWLEDGEMENTS

The present work was sponsored by the Brazilian agency FAPESP, under grants 04/13467-5 and 03/13845-7. The authors thank Dr. Ely Paiva, for his support concerning the airship dynamic model and control strategies, and Dr. Paulo Valente for his assistance regarding Mr. Castro's M.Sc. work.

REFERENCES

Azinheira, J. R., de Paiva, E. C., Ramos, J. G., Bueno, S. S., and Bergerman, M. (2001). Extended dynamic model for aurora robotic airship. In *14th AIAA Lighter-Than-Air Systems Technology Conference*.

Azinheira, J. R., de Paiva, E. C., Ramos, J. J. G., and Bueno, S. S. (2000). Mission path following for an

autonomous unmanned airship. In *International Conference on Robotics and Automation*, San Francisco, USA.

Bailey, T. (2002). *Mobile Robot Localisation and Mapping in Extensive Outdoor Environments*. PhD thesis, Australian Centre for Field Robotics, University of Sydney.

Bueno, S. S., Azinheira, J. R., Ramos, J. G., de Paiva, E. C., Rives, P., Elfes, A., Carvalho, J. R. H., and Silveira, G. F. (2002). Project aurora - towards an autonomous robotic airship. In *Workshop on Aerial Robotics. IEEE International Conference on Intelligent Robot and Systems*, Lausanne, Switzerland.

Castellanos, J. A., Montiel, J. M. M., Neira, J., and Tards, J. D. (1999). The spmap: A probabilistic framework for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation*.

Castellanos, J. A., Neira, J., and Tards, J. D. (2001). Multisensor fusion for simultaneous localization and map building. *IEEE Transactions on Robotics and Automation*.

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

de Paiva, E. C., Azinheira, J. R., Ramos, J. G., Moutinho, A., and Bueno, S. S. (2006). Project aurora: infrastructure and flight control experiments for a robotic airship. *Journal of Field Robotics*. In press.

de Paiva, E. C., Bueno, S. S., Gomes, S. B. V., Ramos, J. J. G., and Bergerman, M. (1999). A control system development environment for aurora's semi-autonomous robotic airship. In *Int. Conference on Robotics and Automation*.

Elfes, A. (1987). Sonal-based real-world mapping and navigation. *IEEE Journal of Robotics and Automation*.

Elfes, A. (1989). *Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation*. PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University.

Elfes, A., Bueno, S. S., Bergerman, M., and Ramos, J. J. G. (1998). A semi-autonomous robotic airship for environmental monitoring missions. In *Int. Conf. on Robotics and Automation*.

Elfes, A., Bueno, S. S., Bergman, M., de Paiva, E. C., Ramos, G. J., and Azinheira, J. R. (2003). Robotic airships for exploration of planetary bodies with an atmosphere: Autonomy challenges. *Autonomous Robots*.

Gomes, S. B. V. and Ramos, J. J. G. (1998). Airship dynamic modeling for autonomous operation. In *Int. Conference on Robotics and Automation*, Leuven, Belgium.

Guivant, J. and Nebot, E. (2001). Optimization of the simultaneous localization and map-building for real-time implementation. *IEEE Transactions on Robotics and Automation*.

Guivant, J. and Nebot, E. (2003). Solving computational and memory requirements of feature-based simultaneous localization and mapping algorithms. *IEEE Transactions on Robotics and Automation*.

Hygounenc, E., Jung, I., Soures, P., and Lacroix, S. (2004). The autonomous blimp project of laas-cnrs: achievements in flight control and terrain mapping. *Intl. Journal of Robotics Research*.

Kantor, G., Wettergreen, D., Ostrowski, J., and Singh, S. (2001). Collection of environmental data from and airship platform. In *SPIE Conference on Sensor Fusion and Decentralized Control in Robotic Systems IV*.

Kim, J. (2004). *Autonomous Navigation for Airborne Applications*. PhD thesis, Australian Centre for Field Robotics, University of Sydney.

Langelaan, J. and Rock, S. (2005). Towards autonomous uav flight in forests. In *AIAA Guidance, Navigation and Control Conference*.

Panzieri, S., Pascucci, F., and Ulivi, G. (2002). An outdoor navigation system using gps and inertial platform. *IEEE/ASME Transactions on Mechatronics*.

Ruiz, I. T., Pelillot, Y., Lane, D. M., and Salson, C. (2001). Feature extraction and data association for auv concurrent mapping and localisation. In *IEEE International Conference on Robotics and Automation*.

Silveira, G., Malis, E., and Rives, P. (2006). Visual servoing over unknown, unstructured, large-scale scenes. In *IEEE International Conference on Robotics and Automation*, Orlando, USA.

Smith, R., Self, M., and Cheeseman, P. (1987). A stochastic map for uncertain spatial relationships. In *Fourth International Symposium on Robotics Research*.

Smith, R., Self, M., and Cheeseman, P. (1988). Estimating uncertain spatial relationships in robotics. In *Autonomous Robot Vehicles*. Springer Verlag.

Sukkarieh, S., Nebot, E., and Durrant-Whyte, H. (1999). A high integrity imu/gps navigation loop for autonomous land vehicle applications. *IEEE Transactions on Robotics and Automation*.

Wimmer, D., Bildstein, M., Well, K. H., Schlenker, M., Kingl, P., and Krplin, B. (2002). Research airship "lotte": development and operation of controllers for autonomous flight phases. In *IEEE/RSJ Int. Conference on Intelligent Robots and Systems*.

Wu, A., Johnson, E., and Proctor, A. (2005). Vision-aided inertial navigation for flight control. In *AIAA Guidance, Navigation and Control Conference*.