

# Analysis, Design, and Implementation of an Integrated Simultaneous Localization and Mapping Algorithm

Pedro Lourenço

Instituto Superior Técnico, Technical University of Lisbon

plourenco@isr.ist.utl.pt

**Abstract**—This paper presents the design, analysis, performance evaluation and preliminary experimental validation of an integrated simultaneous localization and mapping algorithm (SLAM) with application to unmanned aerial vehicles (UAV). The SLAM problem is first formulated in a sensor-based framework and modified in such a way that the system structure may be regarded as time-varying for observability analysis purposes, from which a Kalman filter with globally asymptotically stable error dynamics follows naturally. The proposed solution includes the estimation of both body-fixed linear velocity and rate-gyro measurement biases. Furthermore, the formulation, solution, and validation of the problem of estimating the inertial map and trajectory with uncertainty using the sensor-based map provided by the SLAM filter are also addressed. An optimization problem with a closed-form solution is formulated, and its uncertainty description is derived resorting to perturbation theory. Both simulation results and preliminary experimental results, using an instrumented quadrotor equipped with a RGB-D camera, are included in this work to illustrate the performance of the algorithm under realistic conditions.

**Keywords**—Simultaneous localization and mapping, Filtering, Optimization, Sensor Fusion, Unmanned Aerial Vehicles

## I. INTRODUCTION

**A**UTONOMOUS robot missions, especially those taking place in environments where absolute positioning systems may not be used, either because of their absence or unreliability, raise the need for dependable navigation and positioning algorithms. One of the answers to this problem is probabilistic Simultaneous Localization and Mapping (SLAM), spanning solutions as diverse as EKF SLAM, graph-based SLAM, or particle filters [1]. SLAM is the problem of navigating a vehicle in an unknown environment, by building a map of the area and using this map to deduce its location, without the need for a priori knowledge of location. Despite significant advances, there are still no global convergence results for the most popular strategies, to the best of the author's knowledge. Among all the algorithms that implement SLAM, there are two which bear resemblance to the procedure proposed in this work: the robocentric map joining [2], in the sense that the filtering process takes place in the sensor space, and the RGB-D SLAM [3] for its use of a RGB-D camera for acquisition of images that are processed through a SURF/SIFT algorithm for landmark detection.

The main contributions of the first part of this work are the design, analysis, and validation of a novel sensor-based SLAM filter for tridimensional environments. This work extends that presented in [4], where a bidimensional (2-D) sensor-based SLAM filter was addressed. The assumption that the environment is structured vertically, needed in the 2-D case, is dropped in this work, where only the immobility of the landmarks is assumed. The second part of this work proposes a methodology for obtaining the inertial map and the pose of the vehicle, by building on the work in [5], using the formulation of the extended Procrustes problem there

presented, and providing the uncertainty characterization of the obtained transformation. The proposed integrated algorithm 1) has globally asymptotically stable (GAS) error dynamics; 2) resorts to the linear and angular motion kinematics that are exact; 3) uses the low-cost *Microsoft Kinect*<sup>TM</sup>, in opposition to the 2-D landmark approach, which demands the use of considerably more expensive laser range finders; 4) builds on the well-established linear time-varying Kalman filtering theory; 5) explicitly estimates the rate-gyro bias, fusing high bandwidth dynamic measurements with information on static landmarks; 6) finds the vehicle pose in the inertial frame by solving an optimization problem; and 7) has fully characterized uncertainty, both provided by a Kalman filter and by the proposed novel formulation.

The paper is organized as follows. Section II presents a short description of the problem, with the definition of the system dynamics. The observability analysis is performed in Section III. The filter design is described in Section IV, which includes not only the standard Kalman Filter prediction and update steps, but also landmark detection, data association and loop closing procedures. Section V presents the optimization problem to obtain the inertial map and trajectory and its uncertainty description. In Section VI the complete algorithm is described and, finally, simulation results are presented in Section VII and preliminary experimental results using an instrumented quadrotor are detailed in Section VIII.

*a) Problem Statement:* The problem addressed in this paper is that of developing an online SLAM algorithm for an unmanned aerial vehicle (UAV), such as a quadrotor, by designing 1) a SLAM filter in the space of sensors, providing a sensor-based map and the velocity of the vehicle, producing maps represented by tridimensional landmarks, which may include up to three directions for each position landmark; and 2) an Inertial Map and Trajectory Estimation algorithm resultant of an optimization process by using the sensor-based estimate of the SLAM filter.

*b) Notation:* The superscript  $I$  indicates a vector or matrix expressed in the inertial frame  $\{I\}$ . For the sake of clarity, when no superscript is present, the vector is expressed in the body-fixed frame  $\{B\}$ .  $\mathbf{I}_n$  is the identity matrix of dimension  $n$ , and  $\mathbf{0}_{n \times m}$  is a  $n$  by  $m$  matrix filled with zeros. If  $m$  is omitted, the matrix is square.  $\mathbf{S}[\mathbf{a}]$  is a special skew-symmetric matrix, henceforth called the cross-product matrix, as  $\mathbf{S}[\mathbf{a}]\mathbf{b} = \mathbf{a} \times \mathbf{b}$  with  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ .

## II. DESCRIPTION OF THE PROBLEM

This paper addresses the problem of designing a navigation system for a vehicle operating in an unknown environment. This problem is solved resorting to a novel Sensor-based SLAM filter, where no linearization or approximation is used whatsoever and an optimization-based Inertial Map and Trajectory Estimation algorithm fully characterized by an approximate uncertainty description. The only available sensors are a

triaxial rate-gyro, and a RGB-D camera, such as the *Microsoft Kinect*<sup>TM</sup>, which provide angular rate measurements and RGB-D images, from where 3-D landmarks may be extracted.

### A. Nonlinear System Dynamics

Let  $\mathbf{R}(t) \in \text{SO}(3)$  be the rotation matrix from the body-fixed frame  $\{B\}$  to the inertial frame  $\{I\}$ , with  $\dot{\mathbf{R}}(t) = \mathbf{R}(t) \mathbf{S}[\boldsymbol{\omega}(t)]$ , where  $\boldsymbol{\omega}(t) \in \mathbb{R}^3$  is the angular velocity expressed in body-fixed coordinates. Then, the position and velocity of a landmark expressed in the body-fixed frame,  $\mathbf{p}_i(t) \in \mathbb{R}^3$  and  $\dot{\mathbf{p}}_i(t) \in \mathbb{R}^3$ , satisfy

$$\dot{\mathbf{p}}_i(t) = \mathbf{R}^T(t) ({}^I\dot{\mathbf{p}}_i(t) - {}^I\mathbf{p}(t)) \quad (1)$$

and

$$\dot{\mathbf{p}}_i(t) = -\mathbf{v}(t) - \mathbf{S}[\boldsymbol{\omega}(t)] \mathbf{p}_i(t), \quad (2)$$

respectively, where  ${}^I\mathbf{p}(t) \in \mathbb{R}^3$  represents the vehicle position (as well as the origin of the body-fixed frame) in the inertial frame  $\{I\}$  at time  $t$ ,  ${}^I\mathbf{p}_i(t) \in \mathbb{R}^3$  is the position of landmark  $i$  expressed in the same frame, and  $\mathbf{v}(t) \in \mathbb{R}^3$  denotes the velocity of the vehicle expressed in the body-fixed frame. Note that landmarks are assumed to be static in the inertial frame. It is important to notice that  $\boldsymbol{\omega}(t)$  is available through noisy and biased rate-gyros measurements

$$\boldsymbol{\omega}_m(t) = \boldsymbol{\omega}(t) + \mathbf{b}_\omega(t) + \mathbf{n}_\omega(t), \quad (3)$$

where the bias  $\mathbf{b}_\omega(t) \in \mathbb{R}^3$  is assumed constant and  $\mathbf{n}_\omega(t) \in \mathbb{R}^3$  corresponds to the rate-gyro noise, which is assumed to be zero-mean white Gaussian noise with standard deviation  $\sigma_{n_\omega}$  in each component, i.e.,  $\mathbf{n}_\omega(t) \sim N(0, \sigma_{n_\omega}^2 \mathbf{I}_3)$ . Taking this into account, and using the property  $\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a}$ , it is possible to rewrite (2), in a deterministic setting, as

$$\dot{\mathbf{p}}_i(t) = -\mathbf{v}(t) - \mathbf{S}[\mathbf{p}_i(t)] \mathbf{b}_\omega(t) - \mathbf{S}[\boldsymbol{\omega}_m(t)] \mathbf{p}_i(t). \quad (4)$$

The vehicle-related variables, i.e., the linear velocity and the angular measurement bias will constitute the vehicle state,  $\mathbf{x}_V(t) := [\mathbf{v}^T(t) \ \mathbf{b}_\omega^T(t)]^T \in \mathbb{R}^6$ , with simple dynamics given by

$$\dot{\mathbf{x}}_V(t) = 0, \quad (5)$$

which means that both are assumed, in a deterministic setting, as constant. In the filtering framework, the inclusion of state disturbances allows to consider them as slowly time-varying.

It is now possible to derive the full state dynamics. For that purpose consider the position landmark dynamics (4), which may now be expressed as a function of the state vector, yielding

$$\dot{\mathbf{p}}_i(t) = \mathbf{A}_{MV_i}(\mathbf{p}_i(t)) \mathbf{x}_V(t) - \mathbf{S}[\boldsymbol{\omega}_m(t)] \mathbf{p}_i(t), \quad (6)$$

where  $\mathbf{A}_{MV_i}(\mathbf{p}_i(t)) = [-\mathbf{I}_3 \ -\mathbf{S}[\mathbf{p}_i(t)]]$ . Finally, the observed, also designated as visible, landmarks  $\mathbf{x}_O(t)$  and the unobserved or non-visible ones  $\mathbf{x}_U(t)$  are concatenated in the landmark-based state vector,  $\mathbf{x}_M(t) := [\mathbf{x}_O^T(t) \ \mathbf{x}_U^T(t)]^T$ . The two state vectors here defined constitute the full state vector  $\mathbf{x}_F(t) = [\mathbf{x}_V^T(t) \ \mathbf{x}_M^T(t)]^T$ , with the full system dynamics reading as

$$\begin{cases} \dot{\mathbf{x}}_F(t) = \mathbf{A}_F(t, \mathbf{x}_M(t)) \mathbf{x}_F(t) \\ \mathbf{y}(t) = \mathbf{x}_O(t) \end{cases}, \quad (7)$$

with

$$\mathbf{A}_F(t, \mathbf{x}_M(t)) = \begin{bmatrix} \mathbf{0}_{n_V} & \mathbf{0}_{n_V \times n_M} \\ \mathbf{A}_{MV}(\mathbf{x}_M(t)) & \mathbf{A}_M(t) \end{bmatrix},$$

where

$\mathbf{A}_{MV}(\mathbf{x}_M(t)) = [\mathbf{A}_{MV_1}^T(\mathbf{p}_1(t)) \ \cdots \ \mathbf{A}_{MV_{N_M}}^T(\mathbf{p}_{N_M}(t))]^T$  and  $\mathbf{A}_M(t) = \text{diag}(-\mathbf{S}[\boldsymbol{\omega}_m(t)], \dots, -\mathbf{S}[\boldsymbol{\omega}_m(t)])$ . From (7) it follows that the system may be expressed in a way similar to the usual linear dynamical system form. However, it is obvious to conclude that the system above is nonlinear, as the dynamics matrix depends on the landmarks that may be visible or not. Note that no linearization was performed.

## III. OBSERVABILITY ANALYSIS

Observability is of the utmost importance in any filtering problem, and the work presented in this section aims at analysing the observability of the dynamical system previously exposed. It is important to notice that, although system (7) is inherently nonlinear, discarding the non-visible landmarks  $\mathbf{x}_U(t)$  makes it possible to regard the resulting system as linear time-varying (LTV).

Consider the new state vector  $\mathbf{x}(t) = [\mathbf{x}_V^T(t) \ \mathbf{x}_O^T(t)]^T$ , which does not include the non-visible landmarks, for which the resulting system dynamics can be written as

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}(t, \mathbf{y}(t)) \mathbf{x}(t) \\ \mathbf{y}(t) = \mathbf{C} \mathbf{x}(t) \end{cases}, \quad (8)$$

where  $\mathbf{A}(t, \mathbf{y}(t)) = \begin{bmatrix} \mathbf{0}_{n_V} & \mathbf{0}_{n_V \times n_O} \\ \mathbf{A}_{MV_O}(\mathbf{y}(t)) & \mathbf{A}_{M_O}(t) \end{bmatrix}$  and  $\mathbf{C} = [\mathbf{0}_{n_O \times n_V} \ \mathbf{I}_{n_O}]$ . Note that the matrix  $\mathbf{A}(t, \mathbf{y}(t))$  depends not only on time but also on the system output. Nevertheless, the dependency on the system state is now absent and the system output is known, thus, the system can be seen as a linear time-varying system for observability analysis purposes. According to [6, Lemma 1, Section 3], if the observability Gramian associated with a system with a dynamics matrix depending on the system output is invertible, then the system is observable. This result will be used throughout this section. Before proceeding with this analysis the following assumption is needed.

*Assumption 1:* Any two detected position landmarks are assumed to be different and nonzero, i.e.,  $\mathbf{y}_i(t), \mathbf{y}_j(t) \neq \mathbf{0}$  and  $\mathbf{y}_i(t) \neq \mathbf{y}_j(t)$  for all  $t \geq t_0$  and  $i, j \in \mathcal{I}_O$ , where  $\mathcal{I}_O$  denotes the set of visible landmarks.

Notice that this is a very mild assumption, as it is physically impossible to have two collinear landmarks, let alone equal, visible at the same time, because of the intrinsic characteristics of the camera: its angle of view is always inferior to  $180^\circ$ . Also, as in the body-fixed frame, in which the output is expressed, the origin denotes the position of the vehicle, it is impossible to have it coincide with a landmark.

The following theorem states the analysis of the observability of system (8).

*Theorem 1:* Consider system (8) and let  $\mathcal{T} := [t_0, t_f]$  and  $\{t_1, t_2, t_3\} \in \mathcal{T}$ . The system is observable on  $\mathcal{T}$  in the sense that, given the system output, the initial condition is uniquely defined, if at least one of these conditions hold:

- (i) There are, at least, three visible position landmarks at the same time  $t_1$  that define a plane.
- (ii) There exist two visible position landmarks in the interval  $[t_1, t_2]$  such that at least one of the landmark sets  $\{\mathbf{p}_1(t_1), \mathbf{p}_2(t_1), \mathbf{p}_2(t_2)\}$  and  $\{\mathbf{p}_1(t_1), \mathbf{p}_2(t_1), \mathbf{p}_1(t_2)\}$  defines a plane.
- (iii) There is a visible time-varying position landmark whose coordinates,  $\{\mathbf{p}_1(t_1), \mathbf{p}_1(t_2), \mathbf{p}_1(t_3)\}$ , define a plane.

*Proof:* The proof follows by transforming the system in question by means of a Lyapunov transformation [7, Chapter 1, Section 8], and then proving that the observability Gramian of the transformed system is non-singular.

Let  $\mathbf{T}(t)$  be a Lyapunov transformation such that

$$\mathbf{z}(t) = \mathbf{T}(t) \mathbf{x}(t), \quad (9)$$

where  $\mathbf{T}(t) = \text{diag}(\mathbf{I}_{n_V}, \mathbf{R}_m(t), \dots, \mathbf{R}_m(t))$  and  $\mathbf{R}_m(t) \in \text{SO}(3)$  is a rotation matrix respecting  $\dot{\mathbf{R}}_m(t) = \mathbf{R}_m(t) \mathbf{S}[\boldsymbol{\omega}(t)]$ . A Lyapunov transformation preserves the observability properties of a system, hence it suffices to prove that the new, transformed system is observable. This approach has been used

successfully in the past, see [6] and [8]. The computation of the new system dynamics and output is simple, yielding

$$\begin{cases} \dot{\mathbf{z}}(t) = \mathcal{A}(t, \mathbf{y}(t))\mathbf{z}(t) \\ \mathbf{y}(t) = \mathcal{C}(t)\mathbf{z}(t), \end{cases} \quad (10)$$

with

$$\mathcal{A}(t, \mathbf{y}(t)) = \begin{bmatrix} \mathbf{0}_{n_V} & \mathbf{0}_{n_V \times n_O} \\ \mathcal{A}_{MV}(t, \mathbf{y}(t)) & \mathbf{0}_{n_O} \end{bmatrix},$$

$$\mathcal{C}(t) = [\mathbf{0}_{n_O \times n_V} \quad \text{diag}(\mathbf{R}_m^T(t), \dots, \mathbf{R}_m^T(t))],$$

where

$$\mathcal{A}_{MV}(\mathbf{y}(t)) = [\mathcal{A}_{MV_1}^T(\mathbf{y}_1(t)) \quad \dots \quad \mathcal{A}_{MV_{N_O}}^T(\mathbf{y}_{N_O}(t))]^T,$$

and

$$\mathcal{A}_{MV_i}(t) = [-\mathbf{R}_m(t) \quad -\mathbf{R}_m(t)\mathbf{S}[\mathbf{p}_i(t)]].$$

Before proceeding to computing the observability Gramian associated with (10), it is necessary to know its transition matrix. A simple computation of  $\mathbf{z}(t)$  as a function of  $\mathbf{z}(t_0)$  by solving  $\mathbf{z}(t) = \mathbf{z}(t_0) + \int_{t_0}^t \mathcal{A}(\tau, \mathbf{y}(\tau))\mathbf{z}(\tau) d\tau$  yields

$$\phi(t, t_0) = \begin{bmatrix} \mathbf{I}_{n_V} & \mathbf{0}_{n_V \times n_O} \\ \phi_{MV}(t, t_0) & \mathbf{I}_{n_O} \end{bmatrix}, \quad (11)$$

where

$$\phi_{MV}(t, t_0) = \begin{bmatrix} -\int_{t_0}^t \mathbf{R}_m(\sigma) d\sigma & -\int_{t_0}^t \mathbf{R}_m(\sigma)\mathbf{S}[\mathbf{p}_1(\sigma)] d\sigma \\ \vdots & \vdots \\ -\int_{t_0}^t \mathbf{R}_m(\sigma) d\sigma & -\int_{t_0}^t \mathbf{R}_m(\sigma)\mathbf{S}[\mathbf{p}_{N_O}(\sigma)] d\sigma \end{bmatrix}.$$

Finally, the observability Gramian is

$$\mathcal{W}(t_0, t_f) = \int_{t_0}^{t_f} \phi^T(\tau, t_0) \mathcal{C}^T(\tau) \mathcal{C}(\tau) \phi(\tau, t_0) d\tau, \quad (12)$$

and, if  $\mathcal{W}(t_0, t_f)$  is invertible, the system (10) is observable, in the sense that given the system input and output, the initial condition  $\mathbf{z}(t_0)$  is uniquely defined. The next step is to prove, by contradiction, that this is the case, i.e., by assuming that  $\mathcal{W}(t_0, t_f)$  is singular. In that case, there exists a unit vector  $\mathbf{c} = [\mathbf{c}_1^T \quad \mathbf{c}_2^T \quad \mathbf{c}_3^T \quad \dots \quad \mathbf{c}_{2+N_O}^T]^T \in \mathbb{R}^{(6+3N_O)}$ , such that,

$$\mathbf{c}^T \mathcal{W}(t_0, t_f) \mathbf{c} = 0. \quad (13)$$

Expanding (13) gives

$$\mathbf{c}^T \mathcal{W}(t_0, t_f) \mathbf{c} = \int_{t_0}^{t_f} \|\mathbf{g}(\tau, t_0)\|^2 d\tau, \quad (14)$$

where  $\mathbf{g}(\tau, t_0) = \text{diag}(\mathbf{R}_m(\tau), \dots, \mathbf{R}_m(\tau)) \mathcal{C}(\tau) \phi(\tau, t_0) \mathbf{c}$ . The evaluation of  $\mathbf{g}(\tau, t_0)$  and its derivative yields

$$\mathbf{g}(\tau, t_0) := [\phi_{MV}(\tau, t_0) \quad \mathbf{I}_{n_O}] \mathbf{c} = \begin{bmatrix} \mathbf{c}_3 - \int_{t_0}^{\tau} \mathbf{R}_m(\sigma) \mathbf{c}_1 d\sigma - \int_{t_0}^{\tau} \mathbf{R}_m(\sigma) \mathbf{S}[\mathbf{p}_1(\sigma)] \mathbf{c}_2 d\sigma \\ \vdots \\ \mathbf{c}_{2+N_O} - \int_{t_0}^{\tau} \mathbf{R}_m(\sigma) \mathbf{c}_1 d\sigma - \int_{t_0}^{\tau} \mathbf{R}_m(\sigma) \mathbf{S}[\mathbf{p}_{N_O}(\sigma)] \mathbf{c}_2 d\sigma \end{bmatrix}, \quad (15)$$

$$\frac{d\mathbf{g}(\tau, t_0)}{d\tau} = \begin{bmatrix} -\mathbf{R}_m(\tau) \mathbf{c}_1 - \mathbf{R}_m(\tau) \mathbf{S}[\mathbf{p}_1(\tau)] \mathbf{c}_2 \\ \vdots \\ -\mathbf{R}_m(\tau) \mathbf{c}_1 - \mathbf{R}_m(\tau) \mathbf{S}[\mathbf{p}_{N_O}(\tau)] \mathbf{c}_2 \end{bmatrix}. \quad (16)$$

In order for (13) to be true, both  $\mathbf{g}(\tau, t_0)$  and  $\frac{d\mathbf{g}(\tau, t_0)}{d\tau}$  must be zero for all  $\tau \in \mathcal{T}$ , which implies that

$$\begin{bmatrix} \mathbf{I}_3 & \mathbf{S}[\mathbf{p}_1(\tau)] \\ \vdots & \vdots \\ \mathbf{I}_3 & \mathbf{S}[\mathbf{p}_{N_O}(\tau)] \end{bmatrix} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix} = \mathbf{0}, \quad \forall \tau \in \mathcal{T}. \quad (17)$$

Thus, it remains to show that, if the observability Gramian is singular the conditions of Theorem 1 cannot hold. Consider then the situation where there are three visible landmarks  $\mathbf{p}_i(t_1)$ ,  $i \in \{1, 2, 3\}$ . In this case (17) can be rewritten as

$$\begin{bmatrix} \mathbf{I}_3 & \mathbf{S}[\mathbf{p}_1(t_1)] \\ \mathbf{0}_3 & \mathbf{S}[\mathbf{p}_2(t_1) - \mathbf{p}_1(t_1)] \\ \mathbf{0}_3 & \mathbf{S}[\mathbf{p}_3(t_1) - \mathbf{p}_1(t_1)] \end{bmatrix} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \end{bmatrix} = \mathbf{0}. \quad (18)$$

From this, it is simple to find that either  $\mathbf{c}_2 = \mathbf{c}_1 = \mathbf{0}$  thus contradicting the hypothesis of a unit vector  $\mathbf{c}$  or

$$\mathbf{c}_2 = \alpha (\mathbf{p}_2(t_1) - \mathbf{p}_1(t_1)) = \beta (\mathbf{p}_3(t_1) - \mathbf{p}_1(t_1)). \quad (19)$$

The second possibility implies that all three landmarks form a line, thus contradicting the hypothesis of the theorem. In the case where any of the remaining conditions applies, an equation similar to (18) may be constructed, this time with the sets  $\{\mathbf{p}_1(t_1), \mathbf{p}_2(t_1), \mathbf{p}_2(t_2)\}$  or  $\{\mathbf{p}_1(t_1), \mathbf{p}_2(t_1), \mathbf{p}_1(t_2)\}$ , for condition (ii) and  $\{\mathbf{p}_1(t_1), \mathbf{p}_1(t_2), \mathbf{p}_1(t_3)\}$  for condition (iii). Hence, if the observability Gramian is not invertible, none of the hypothesis of the theorem can hold, which means that, if at least one of the conditions of Theorem 1 holds, then  $\mathcal{W}(t_0, t_f)$  is invertible on  $\mathcal{T}$ , and, using [6, Lemma 1, Section 3], it follows that (10) is observable. Moreover, as the Lyapunov transformation (9) preserves observability, the system (8) is also observable, thus concluding the proof of the theorem. ■

Given the sufficient conditions for observability, a Kalman Filter for the nonlinear system (7), with globally asymptotically stable error dynamics, can be designed following the classical approach. The following result addresses the equivalence between the state of the nonlinear system (8), regarded as LTV, and that of the nominal nonlinear system (7), when the non-visible landmarks are not considered.

*Theorem 2:* Consider that the conditions of Theorem 1 hold. Then,

- (i) the initial state of the nonlinear system (7), discarding the non-visible landmarks, is uniquely determined, and is the same of the nonlinear system (8), regarded as LTV;
- (ii) a state observer with uniformly globally asymptotically stable error dynamics for the LTV system is also a state observer for the underlying nonlinear system, with uniformly globally asymptotically stable error dynamics.

*Proof:* Consider the transformed system (10), whose state, and therefore initial condition, is related with the state of the nonlinear system (8), regarded as LTV. The proof follows with the transformed system for simplicity of analysis. Let the initial condition for this system be given by  $\bar{\mathbf{z}}(t_0) = [\bar{\mathbf{v}}^T(t_0) \quad \bar{\mathbf{b}}_\omega^T(t_0) \quad \bar{\mathbf{z}}_{p_1}^T(t_0) \quad \dots \quad \bar{\mathbf{z}}_{p_N}^T(t_0)]^T$ , where  $\bar{\mathbf{z}}_{p_i}(t) = \mathbf{R}_m(t) \mathbf{y}_i(t)$ , which comes from  $\mathbf{y}(t) = \mathcal{C}(t) \mathbf{z}(t)$ . Recall that  $\mathbf{z}(t) = \phi(t, t_0) \bar{\mathbf{z}}(t_0)$ , which then yields  $\mathbf{y}(t) = \mathcal{C}(t) \phi(t, t_0) \bar{\mathbf{z}}(t_0)$ . The  $i$ -th component of the output of the system is given by

$$\begin{aligned} \mathbf{y}_i(t) &= \mathbf{R}_m^T(t) \mathbf{R}_m(t_0) \mathbf{y}_i(t_0) \\ &\quad - \mathbf{R}_m^T(t) \int_{t_0}^t \mathbf{R}_m(\sigma) (\bar{\mathbf{v}}(t_0) + \mathbf{S}[\mathbf{y}_i(\sigma)] \bar{\mathbf{b}}_\omega(t_0)) d\sigma, \end{aligned} \quad (20)$$

where the relation between  $\mathbf{y}_i(t)$  and  $\bar{\mathbf{z}}_i(t_0)$  was used. The next steps include left multiplying this expression by  $\mathbf{R}_m(t)$ , differentiating, and further simplifying by left-multiplying both sides by  $\mathbf{R}_m(t)^T$ , yielding the first derivative of  $\mathbf{y}_i(t)$ ,

$$\dot{\mathbf{y}}_i(t) = -\bar{\mathbf{v}}(t_0) - \mathbf{S}[\mathbf{y}_i(t)] (\bar{\mathbf{b}}_\omega(t_0) - \omega(t)). \quad (21)$$

Now consider the nonlinear system (7). The initial condition of this system is given by  $\mathbf{x}(t_0) = [\mathbf{v}^T(t_0) \quad \mathbf{b}_\omega^T(t_0) \quad \mathbf{p}_1^T(t_0) \quad \dots \quad \mathbf{p}_N^T(t_0)]^T$ , where  $\mathbf{p}_i(t) =$

$\mathbf{y}_i(t)$  for all  $\mathbf{p}_i(t) \in \mathcal{I}_O$ . The output of the system is related to the state by  $\mathbf{y}(t) = \mathbf{C} \int_{t_0}^t \mathbf{A}(\mathbf{x}(\sigma), \sigma) \mathbf{x}(\sigma) d\sigma + \mathbf{C} \mathbf{x}(t_0)$ , which, after a simple computation and substituting  $\mathbf{p}_i(t)$  by  $\mathbf{y}_i(t)$ , yields

$$\mathbf{y}_i(t) = \mathbf{y}_i(t_0) - \int_{t_0}^t (\mathbf{v}(t_0) + \mathbf{S}[\mathbf{y}_i(\sigma)] (\mathbf{b}_\omega(t_0) - \boldsymbol{\omega}(\sigma))) d\sigma,$$

with first time derivative given by

$$\dot{\mathbf{y}}_i(t) = -\mathbf{v}(t_0) - \mathbf{S}[\mathbf{y}_i(t)] (\mathbf{b}_\omega(t_0) - \boldsymbol{\omega}(t)). \quad (22)$$

Comparison of (21) with (22) yields the following equation

$$\mathbf{0} = (\bar{\mathbf{v}}(t_0) - \mathbf{v}(t_0)) - \mathbf{S}[\mathbf{y}_i(t)] (\bar{\mathbf{b}}_\omega(t_0) - \mathbf{b}_\omega(t_0)) \quad (23)$$

for all  $t$  in  $\mathcal{T}$  and  $i \in \mathcal{I}_O$ . When the conditions of Theorem 1 hold, this system yields  $\bar{\mathbf{v}}(t_0) = \mathbf{v}(t_0)$  and  $\bar{\mathbf{b}}_\omega(t_0) = \mathbf{b}_\omega(t_0)$  by a similar reasoning to that used to prove the sufficiency of those conditions to the observability of the system. As mentioned before, the initial condition  $\bar{\mathbf{z}}(t_0)$  is related to that of the nonlinear system (8) by the Lyapunov transformation  $\mathbf{T}(t)$ . Hence, under the conditions of Theorem 1, the initial state of the nonlinear system (8), regarded as LTV, and the initial state of the nonlinear system (7), discarding the non-visible landmarks, are the same and uniquely defined.

The first part of the theorem, now proved, gives insight for the proof of the second part. An observer designed for a LTV system with globally asymptotically stable error dynamics has an estimation error convergent to zero, implying that the estimates asymptotically tend to the true state. Therefore, if the true state of the nonlinear system and the state of the LTV system are one and the same, as proved, the estimation error of the state of the nonlinear system converges to zero too. ■

Given that a GAS observer for system (8) is an observer for the nominal nonlinear system, the design of a globally asymptotically stable observer for the LTV system follows. This step requires that the pair  $(\mathbf{A}(t, \mathbf{y}(t)), \mathbf{C})$  is uniformly completely observable as declared in [9]. The following theorem states the conditions for this property to be verified.

*Theorem 3:* Consider system (8). The pair  $(\mathbf{A}(t, \mathbf{y}(t)), \mathbf{C})$  is uniformly completely observable, if there exists a  $\delta > 0$  such that, for all  $t \geq t_0$ , it is possible to choose  $\{t_1, t_2, t_3\} \in \mathcal{T}_\delta$ ,  $\mathcal{T}_\delta = [t, t + \delta]$ , such that at least one of the following conditions hold:

- (i) There are at least three visible landmarks  $\mathbf{p}_1(t)$ ,  $\mathbf{p}_2(t)$  and  $\mathbf{p}_3(t)$  such that  $(\mathbf{p}_1(t_1) - \mathbf{p}_2(t_1)) \times (\mathbf{p}_1(t_1) - \mathbf{p}_3(t_1)) \neq \mathbf{0}$ ,
- (ii) There exist two visible position landmarks at times  $t_1, t_2$  such that at least one of the landmark sets  $\{\mathbf{p}_1(t_1), \mathbf{p}_2(t_1), \mathbf{p}_2(t_2)\}$  or the  $\{\mathbf{p}_1(t_1), \mathbf{p}_2(t_1), \mathbf{p}_1(t_2)\}$  forms a plane,
- (iii) There is a visible time-varying position landmark whose coordinates, at three different instants of time  $\{t_1, t_2, t_3\}$ , form a plane.

*Proof:* The proof follows similar steps to the proof of Theorem 1 but considering uniform bounds for all  $t \geq t_0$  and intervals  $[t, t + \delta]$ , and is therefore omitted. The overall concept of the proof can be found in [10], with different dynamics. ■

#### IV. SENSOR-BASED SLAM FILTER DESIGN

This section addresses the design of the sensor-based 3D-SLAM filter. A discrete Kalman filter is designed, considering the sample-based/digital characteristics of both sensors needed for this work: an IMU (or more precisely a triad of rate-gyros) and a depth camera (or other tridimensional relative position sensor). Hence, it is important to obtain the discrete-time version of the dynamic system under analysis.

#### A. Discrete Dynamics

Denoting the synchronized sampling period of both sensors as  $T_s$ , the discrete time steps can be expressed as  $t_k = kT_s + t_0$ , where  $k \in \mathbb{N}_0$  and  $t_0$  is the initial time. Thus, the discretized system is characterized by the state  $\mathbf{x}_k := \mathbf{x}(t_k)$ , the dynamics matrix  $\mathbf{A}_k(\mathbf{y}_k) := \mathbf{A}(t_k, \mathbf{y}(t_k))$  and the output matrix  $\mathbf{C}_k := \mathbf{C}(t_k)$ . Finally, the Euler discretization of the system dynamics (8), including system disturbance and measurement noise, yields

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{F}_k(\mathbf{y}_k) \mathbf{x}_k + \boldsymbol{\xi}_k \\ \mathbf{y}_{k+1} = \mathbf{H}_{k+1} \mathbf{x}_{k+1} + \boldsymbol{\theta}_{k+1} \end{cases}, \quad (24)$$

where  $\mathbf{F}_k(\mathbf{y}_k) := \mathbf{I}_{n_x} + T_s \mathbf{A}_k(\mathbf{y}_k)$  and  $\mathbf{H}_{k+1} := \mathbf{C}_{k+1}$ . The disturbance vector  $\boldsymbol{\xi}_k$  and the measurement noise vector  $\boldsymbol{\theta}_k$  are both zero-mean discrete white Gaussian noise, with  $\langle \boldsymbol{\xi}_k \boldsymbol{\xi}_k^T \rangle = \boldsymbol{\Xi}_k$  and  $\langle \boldsymbol{\theta}_k \boldsymbol{\theta}_k^T \rangle = \boldsymbol{\Theta}_k$ , where  $\langle \cdot \rangle$  denotes the expected value of its arguments.

#### B. Prediction Step

System (24) does not include the non-visible landmarks, which must be propagated in open-loop using (7). Thus, with the full state vector,  $\mathbf{x}_{F_k} := [\mathbf{x}_k^T \quad \mathbf{x}_{M_k}^T]^T$ , the prediction step of the Kalman filter is given by

$$\begin{cases} \hat{\mathbf{x}}_{F_{k+1|k}} = \mathbf{F}_{F_{k+1|k}}(\mathbf{y}_k, \hat{\mathbf{x}}_{U_{k|k}}) \hat{\mathbf{x}}_{F_{k|k}} \\ \boldsymbol{\Sigma}_{F_{k+1|k}} = \mathbf{F}_{F_{k+1|k}} \boldsymbol{\Sigma}_{F_{k+1|k}} \mathbf{F}_{F_{k+1|k}}^T + \boldsymbol{\Xi}_{F_{k+1|k}} \end{cases}, \quad (25)$$

where  $\mathbf{F}_{F_{k+1|k}}(\mathbf{y}_k, \hat{\mathbf{x}}_{U_{k|k}}) = \mathbf{I}_{n_x} + T_s \mathbf{A}_{F_{k+1|k}}(\mathbf{y}_k, \hat{\mathbf{x}}_{U_{k|k}})$  and  $\boldsymbol{\Xi}_{F_{k+1|k}} = \text{diag}(\boldsymbol{\Xi}_k, \boldsymbol{\Xi}_{U_{k+1|k}})$ , with  $\mathbf{A}_{F_{k+1|k}}(\mathbf{y}_k, \hat{\mathbf{x}}_{U_{k|k}}) := \mathbf{A}_F(t_k, \mathbf{y}_k, \hat{\mathbf{x}}_{U_{k|k}})$ . This prediction step uses the measurements of the rate-gyros, propagating the state every time a reading is available.

#### C. Update Step

The update step is divided in two different stages, landmark association and the update equations. This step occurs every time 2-D colour and depth images are available from the Kinect. Then, an implementation of SURF [11] detects features in the 2-D picture of the environment. The resulting features are then matched to a pointcloud built with the depth image, returning a set of observed tridimensional landmarks in cartesian coordinates. The SLAM filter does not know a priori if a landmark from this set is in the current map or if it is the first time it is seen. This is when data association takes place, associating the measured data with the known, existing, landmarks. Wrong associations may have a very negative effect on the estimation, so this is a field where a lot of research effort has been put, yielding various algorithms from the community. The algorithm used, the Joint Compatibility Branch and Bound [12], performs a depth-first search only expanding nodes when the joint associations are jointly compatible in a probabilistic sense. Note that both the landmark detection and association algorithms may be substituted by others, as they are independent from the filtering technique described in this paper. The association algorithm provides the innovation vector and its covariance matrix, and also redefines the new sets of visible and non-visible landmarks,

$$\begin{cases} \boldsymbol{\nu}_{k+1} = \mathbf{y}_{k+1} - \mathbf{H}_{k+1} \hat{\mathbf{x}}_{k+1|k} \\ \boldsymbol{\Sigma}_{\boldsymbol{\nu}_{k+1}} = \mathbf{H}_{k+1} \boldsymbol{\Sigma}_{k+1|k} \mathbf{H}_{k+1}^T + \boldsymbol{\Theta}_{k+1}. \end{cases} \quad (26)$$

The update equations are standard, and given by

$$\begin{cases} \mathbf{K}_{k+1} = \boldsymbol{\Sigma}_{k+1|k} \mathbf{H}_{k+1}^T \boldsymbol{\Sigma}_{\boldsymbol{\nu}_{k+1}}^{-1} \\ \hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} \boldsymbol{\nu}_{k+1} \\ \boldsymbol{\Sigma}_{k+1|k+1} = \boldsymbol{\Sigma}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{H}_{k+1} \boldsymbol{\Sigma}_{k+1|k}. \end{cases} \quad (27)$$

#### D. Loop Closing

The loop closing problem is very important in any SLAM algorithm, as it enables the recognition of previously visited places, allowing the reduction of the uncertainty associated with the landmarks. The rather naive loop closing algorithm used consists on using only a subset of the state landmarks in the association algorithm, namely the more recent ones  $\mathcal{I}_{rec}$ , and separating the full state into three subsets, the current, the old,  $\mathcal{I}_{old}$ , and the ones in between,  $\mathcal{I}_{gap}$ . This allows the duplication of landmarks when an area is revisited. Then, periodically, the loop closing algorithm tries to associate landmarks in  $\mathcal{I}_{rec}$  and  $\mathcal{I}_{old}$  using an adapted version of the association algorithm. If the number of jointly compatible associations passes a certain predefined threshold, a loop closure takes place. The loop closure is then incorporated in the filter by means of a noise free measurement: an association between landmark  $i \in \mathcal{I}_{rec}$  and landmark  $j \in \mathcal{I}_{old}$ . This procedure is similar to the one proposed in [4]. Note that neither the association algorithm or the loop closing procedure are the focus of this work, but rather a demonstration that the filter is able to provide consistent estimates. Furthermore, they can be easily replaced by any other method.

### V. TRIDIMENSIONAL INERTIAL MAP AND TRAJECTORY ESTIMATION

The framework of the proposed SLAM filter is completely independent of the inertial frame, as every input and state are expressed in the body-fixed frame. Therefore, the localization of the vehicle is trivial, being the origin of the referred frame. Nevertheless, the usual SLAM algorithms perform the mapping and localization in an inertial frame, and many applications require the inertial map and the trajectory of the vehicle. The sensor-based map is readily available, and, if an inertial estimate is provided, the pose of the vehicle can be estimated by the comparison of these two maps. The problem of computing the transformation that maps the two sets of points is usually called the Procrustes Problem. Its generalization for rotation, translation and scaling can be traced back to [13] and [14]. This section presents an Online Tridimensional Inertial Map and Trajectory Estimate algorithm, henceforth denoted as the *esTIMATE* algorithm, that solves the extended Procrustes problem and provides a measure of the uncertainty associated with its outputs.

#### A. Definition and Solution of the Optimization Problem

Consider the existence of two landmark sets,  $\mathcal{I}_{I_k}$  and  $\mathcal{I}_{B_k}$ , which contain, respectively, the landmarks expressed in the inertial frame and those in the body-fixed frame. Each landmark  ${}^I\hat{\mathbf{p}}_{i_k} \in \mathcal{I}_{I_k}$  corresponds to a landmark  $\hat{\mathbf{p}}_{i_k} \in \mathcal{I}_{B_k}$ , with  $i \in \{1, \dots, N_M\}$ , and that correspondence is expressed by

$${}^I\hat{\mathbf{p}}_{i_k} = {}^I\hat{\mathbf{p}}_k + \hat{\mathbf{R}}_k \hat{\mathbf{p}}_{i_k}, \quad (28)$$

where the pair  $(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k)$ , namely the orientation and position of the vehicle in frame  $\{I\}$ , fully defines the transformation from the body-fixed frame  $\{B\}$  to the inertial frame, as it represents the estimated rotation and translation from  $\{B\}$  to  $\{I\}$ . Given the relation between the two sets, it is possible to define the error function

$${}^I\mathbf{e}_{i_k} = {}^I\hat{\mathbf{p}}_{i_k} - \hat{\mathbf{R}}_k \hat{\mathbf{p}}_{i_k} - {}^I\hat{\mathbf{p}}_k, \quad (29)$$

that represents the error between the inertial landmark estimate  $i \in \mathcal{I}_{I_k}$  and its sensor-based homologous, rotated and

translated with the estimated transformation. Obtaining the pair  $(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k)$  is the purpose of the optimization problem that results from minimizing this error function,

$$(\mathbf{R}_k^*, {}^I\mathbf{p}_k^*) = \arg \min_{\substack{\hat{\mathbf{R}}_k \in \text{SO}(3) \\ {}^I\hat{\mathbf{p}}_k \in \mathbb{R}^3}} G(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k), \quad (30)$$

where the functional  $G(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k) = \sum_{i=1}^{N_T} \sigma_{i_k}^{-2} \|{}^I\mathbf{e}_{i_k}\|^2$  is given, in matrix form, by

$$G(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k) = \frac{1}{N_T} \left\| (\mathbf{Y}_k - \hat{\mathbf{R}}_k \mathbf{X}_k - {}^I\hat{\mathbf{p}}_k \mathbf{1}^T) \boldsymbol{\Sigma}_{I_{e_k}}^{-1/2} \right\|^2, \quad (31)$$

where  $\mathbf{Y}_k = [{}^I\hat{\mathbf{p}}_{1_k} \dots {}^I\hat{\mathbf{p}}_{N_T k}]$  and  $\mathbf{X}_k = [{}^I\hat{\mathbf{p}}_{1_k} \dots {}^I\hat{\mathbf{p}}_{N_T k}]$ ,  $\mathbf{Y}_k, \mathbf{X}_k \in \mathbb{R}^{3 \times N_T}$ , are the concatenation of the landmark vectors expressed in the inertial and body-fixed frames, respectively,  $\mathbf{1} = [1 \dots 1]^T \in \mathbb{R}^{N_T}$  is a vector of ones, and the weight matrix  $\boldsymbol{\Sigma}_{I_{e_k}}$  is a diagonal matrix whose entries are the individual weights  $\sigma_{1_k}^2, \dots, \sigma_{N_T k}^2$  that model the uncertainty of each landmark pair, and that are conservatively defined as

$$\begin{aligned} \sigma_{i_k}^2 &= \lambda_{max}(\boldsymbol{\Sigma}_{I_{\mathbf{p}_{i_k}}}) + \lambda_{max}(\boldsymbol{\Sigma}_{\mathbf{p}_{i_k}}) \\ &\geq \lambda_{max}(\boldsymbol{\Sigma}_{I_{\mathbf{p}_{i_k}}} + \mathbf{R}_k \boldsymbol{\Sigma}_{\mathbf{p}_{i_k}} \mathbf{R}_k^T) \end{aligned} \quad (32)$$

because the true  $\boldsymbol{\Sigma}_{I_{e_k}}$  is not known. Note that  $N_T$  may be different from  $N_M$ , as the landmarks used in the algorithm may be only subsets of  $\mathcal{I}_{I_k}$  and  $\mathcal{I}_{B_k}$ .

The work of [13] proposes a closed-form solution to a similar optimization problem, where no weights were used. The derivation of the solution for (30) follows closely that of the referred work, and is therefore omitted, due to lack of space. However, the introduction of the normalization weight matrix  $\mathbf{W}_k := \boldsymbol{\Sigma}_{I_{e_k}}^{-1} - \frac{1}{N_{W_k}} \boldsymbol{\Sigma}_{I_{e_k}}^{-1} \mathbf{1} \mathbf{1}^T \boldsymbol{\Sigma}_{I_{e_k}}^{-1}$ , where  $N_{W_k} = \mathbf{1}^T \boldsymbol{\Sigma}_{I_{e_k}}^{-1} \mathbf{1}$ , is needed before presenting the actual solution

The optimal translation vector follows directly from rearranging of the function  $G(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k)$ , yielding

$${}^I\mathbf{p}_k^* = \frac{1}{N_{W_k}} (\mathbf{Y}_k - \mathbf{R}_k^* \mathbf{X}_k) \boldsymbol{\Sigma}_{I_{e_k}}^{-1} \mathbf{1} \quad (33)$$

It is straightforward to show that this vector translates the centroid of the body-fixed landmarks rotated to the inertial frame to the centroid of the inertial landmarks.

Finally, it remains to find the optimal rotation from the body-fixed frame onto the inertial frame. [13, Lemma, Section II] shows that the optimal rotation for this problem is given by

$${}^I_{B_k} \hat{\mathbf{R}}_k^* = \mathbf{U} \text{diag}(1, 1, |\mathbf{U}|) \text{diag}(1, 1, |\mathbf{V}|) \mathbf{V}^T, \quad (34)$$

where

$$\mathbf{U} \mathbf{D} \mathbf{V}^T = \text{svd}(\mathbf{B}_k^T), \quad (35)$$

and  $\mathbf{B}_k := \mathbf{X}_k \mathbf{W}_k \mathbf{Y}_k^T$ .

#### B. The *esTIMATE* algorithm

In fact, as the landmarks in  $\mathcal{I}_I$  are static, the correspondence in (28) is also valid between  ${}^I\hat{\mathbf{p}}_{i_{k+1}}$  and  $\hat{\mathbf{p}}_{i_k}$ ,  $i \in \{1, \dots, N_M\}$ . This step is of the utmost importance in the design of the algorithm, because it is only possible to compute the pair  $(\hat{\mathbf{R}}_k, {}^I\hat{\mathbf{p}}_k)$  having, a priori, both landmark sets. However, the computation of  ${}^I\hat{\mathbf{p}}_{i_k}$  requires the transformation between

frames to be available. This algebraic loop is averted by using the update equation

$${}^I \hat{\mathbf{p}}_{i_{k+1}} = {}^I \hat{\mathbf{p}}_k + \hat{\mathbf{R}}_k \hat{\mathbf{p}}_{i_k}, \quad (36)$$

after the computation of the optimal translation and rotation using the sensor-based estimate of instant  $k$  and the inertial estimate computed in the previous one. Note that the solution of the Procrustes problem uses only a subset containing the most recently visible landmarks in  $\mathcal{I}_{B_k}$ , provided that the dimension of the resulting set  $\mathcal{I}_{B_{T_k}}$  is greater, if possible, than a predefined threshold. This is done in order to make the algorithm more computationally efficient, as well as to avoid using older landmarks whose estimation may be possibly worse.

### C. Uncertainty Description of the Inertial Estimates

The work exposed in this section, including the estimates for the vehicle pose, given by (33) and (34), and the update equations (36), allows the real time computation of the vehicle trajectory and of the inertial map. However, the tridimensional inertial map and trajectory estimation algorithm here described, apart from having uncertainties involved, requires the knowledge of the uncertainty of both the inertial and sensor-based landmark estimates. The latter is directly provided by the Kalman filter, but the former is yet to be described. These statistical properties have been the subject of study in works such as [15] and [16]. In these works, perturbation theory was employed. However, some rather limiting assumptions were taken, namely, the absence of weighting in the functional  $G(\hat{\mathbf{R}}_k, {}^I \hat{\mathbf{p}}_k)$ , the use of small rotations, and the same covariance for each landmark. The scope of this section is to provide approximate uncertainty descriptions for the parameters output by this algorithm. That is done by building on the referred work and considering arbitrary rotations and translations, individual weights, and a total covariance matrix for the whole inertial map. A similar approach for the bidimensional case is hinted at [5].

The error models of the known sensor-based and inertial variables are defined as follows

$$\hat{\mathbf{p}}_{i_k} = \hat{\mathbf{p}}_{i_k}^{(0)} + \epsilon \hat{\mathbf{p}}_{i_k}^{(1)} + O(\epsilon^2) \quad (37a)$$

$${}^I \hat{\mathbf{p}}_{i_k} = {}^I \hat{\mathbf{p}}_{i_k}^{(0)} + \epsilon {}^I \hat{\mathbf{p}}_{i_k}^{(1)} + O(\epsilon^2) \quad (37b)$$

$${}^I \hat{\mathbf{p}}_k = {}^I \hat{\mathbf{p}}_k^{(0)} + \epsilon {}^I \hat{\mathbf{p}}_k^{(1)} + O(\epsilon^2) \quad (37c)$$

$${}^I \hat{\mathbf{p}}_{i_{k+1}} = {}^I \hat{\mathbf{p}}_{i_{k+1}}^{(0)} + \epsilon {}^I \hat{\mathbf{p}}_{i_{k+1}}^{(1)} + O(\epsilon^2), \quad (37d)$$

where  $\epsilon$  is the smallness parameter and the notation  $O(\epsilon^n)$  stands for all the terms of order  $n$  or superior, the zero order terms are deterministic, i.e.,  $\langle \mathbf{a}^{(0)} \rangle = \mathbf{a}^{(0)}$ , and the first order terms,  $\mathbf{a}_i^{(1)}$ , are assumed to be Gaussian distributed with zero mean and covariance matrices defined by  $\Sigma_{a_{ij_k}} := \langle \mathbf{a}_i^{(1)} \mathbf{a}_j^{(1)T} \rangle$ .

The rotation matrix from  $\{B\}$  to  $\{I\}$  is assumed to have the special structure

$$\hat{\mathbf{R}}_k = \exp(\mathbf{S}[\Omega_k]) \hat{\mathbf{R}}_k^{(0)} = [\mathbf{I} + \epsilon \mathbf{S}[\Omega_k] + O(\epsilon^2)] \hat{\mathbf{R}}_k^{(0)}, \quad (38)$$

where  $\Omega_k \in \mathbb{R}^3$  denotes the rotation error and  $\mathbf{R}_k^{(0)}$  the true rotation matrix. With all the error models defined, the next step is to compute the expressions that define  ${}^I \hat{\mathbf{p}}_{i_{k+1}}^{(0)}$ ,  ${}^I \hat{\mathbf{p}}_{i_{k+1}}^{(1)}$ ,  ${}^I \hat{\mathbf{p}}_k^{(0)}$ ,  ${}^I \hat{\mathbf{p}}_k^{(1)}$ , and  $\Omega_k$  as well as their expected values and covariance matrices.

1) *Rotation uncertainty*: The rotation matrix obtained by the optimization process described before belongs to the special orthogonal group  $\text{SO}(3)$ , which yields the two constraints  $\mathbf{R}_k^T \mathbf{R}_k = \mathbf{I}$  and  $|\mathbf{R}_k| = 1$ . It is straightforward to see that  $\hat{\mathbf{R}}_k^{(0)}$  also belongs to  $\text{SO}(3)$ , as it must be considered the true rotation matrix, using the restraints of the space and the error models.

Knowing the properties of the deterministic term in  $\mathbf{R}_k$ , it remains to compute the uncertainty associated, which was assumed in (38) to be related to the rotation error  $\Omega_k$ . Thus, the next steps describe the computation of this rotation error and its statistical properties, starting with some properties associated with the closed-form solution of the optimization problem (30). Consider the matrix that was used to compute the estimated rotation,  $\mathbf{B}_k$ . This matrix can be described in terms of its error model, using that of matrices  $\mathbf{X}_k$  and  $\mathbf{Y}_k$ , which are a generalization of (37a) and (37b), yielding

$$\mathbf{B}_k = \mathbf{B}_k^{(0)} + \epsilon \mathbf{B}_k^{(1)}, \quad (39)$$

with  $\mathbf{B}_k^{(0)} = \mathbf{X}_k^{(0)} \mathbf{W}_k \mathbf{Y}_k^{(0)T}$ , and  $\mathbf{B}_k^{(1)} = \mathbf{X}_k^{(1)} \mathbf{W}_k \mathbf{Y}_k^{(0)T} + \mathbf{X}_k^{(0)} \mathbf{W}_k \mathbf{Y}_k^{(1)T}$ . From the proof of [13, Lemma, Section II], it is known that the matrix  $\mathbf{B}_k \mathbf{R}_k$  is symmetrical, and thus, using the error model (38) and (39) the following expression may be derived

$$\begin{aligned} & \text{skew}(\mathbf{B}_k \mathbf{R}_k) \\ &= \text{skew}(\mathbf{B}_k^{(0)} \hat{\mathbf{R}}_k^{(0)}) + \epsilon \text{skew}(\mathbf{B}_k^{(1)} \hat{\mathbf{R}}_k^{(0)} + \mathbf{B}_k^{(0)} \mathbf{S}[\Omega_k] \hat{\mathbf{R}}_k^{(0)}) \\ &= \mathbf{0}, \end{aligned}$$

which shows that each skew operator is null, thus yielding

$$\text{skew}(\mathbf{B}_k^{(0)} \mathbf{S}[\Omega_k] \hat{\mathbf{R}}_k^{(0)}) = -\text{skew}(\mathbf{B}_k^{(1)} \hat{\mathbf{R}}_k^{(0)}). \quad (40)$$

The final objective of this step, as well as of the following, is to compute  $\Omega_k$ . For that purpose, the last relation is rearranged and computed element by element in order to extract the underlying linear matrix equation

$$\mathcal{A}_k \Omega_k = \mathbf{b}_k, \quad (41)$$

where  $\mathcal{A}_k := [\text{tr}(\mathbf{A}_k) \mathbf{I}_3 - \mathbf{A}_k^T]$ , and the vector  $\mathbf{b}_k \in \mathbb{R}^3$  is defined as  $\mathbf{b}_k := [c_{23} - c_{32} \quad c_{31} - c_{13} \quad c_{12} - c_{21}]^T$ ,  $c_{ij} \in \mathbb{R}$  being the element on the  $i$ -th row and  $j$ -th column of  $\mathbf{C}_k = \hat{\mathbf{R}}_k^{(0)} \mathbf{B}_k^{(1)}$  and  $\mathbf{A}_k = \hat{\mathbf{R}}_k^{(0)} \mathbf{B}_k^{(0)}$ .

Note that from the linear equation now derived it is straightforward to obtain  $\Omega_k$ , as long as  $\mathcal{A}_k$  is invertible. The next step in finding  $\Omega_k$  is then to unveil the conditions in which  $\mathcal{A}_k$  is invertible, if any. For that purpose, rewriting it as a sum of terms involving the landmarks (both inertial and sensor-based) provides a better insight on its properties. Consider the matrices  $\mathbf{B}_k^{(0)}$  and  $\mathbf{A}_k$  expressed as summations,

$$\mathbf{B}_k^{(0)} = \sum_{i=1}^{N_T} \sigma_{i_k}^{-2} \left[ \hat{\mathbf{p}}_{i_k}^{(0)I} \hat{\mathbf{p}}_{i_k}^{(0)T} - \frac{1}{N_{W_k}} \sum_{j=1}^{N_T} \sigma_{j_k}^{-2} \hat{\mathbf{p}}_{i_k}^{(0)I} \hat{\mathbf{p}}_{j_k}^{(0)T} \right], \quad (42)$$

and

$$\mathbf{A}_k = \sum_{i=1}^{N_T} \sigma_{i_k}^{-2} \hat{\mathbf{R}}_k^{(0)} \left[ \hat{\mathbf{p}}_{i_k}^{(0)I} \hat{\mathbf{p}}_{i_k}^{(0)T} - \frac{1}{N_{W_k}} \sum_{j=1}^{N_T} \sigma_{j_k}^{-2} \hat{\mathbf{p}}_{i_k}^{(0)I} \hat{\mathbf{p}}_{j_k}^{(0)T} \right], \quad (43)$$

respectively. If  $\mathcal{A}_k$  is non-singular, the only solution for the following expression is the trivial solution  $\mathbf{u} = \mathbf{0}$ .

$$\mathbf{u}^T \mathcal{A}_k \mathbf{u} = 0, \quad \mathbf{u} \in \mathbb{R}^3 \quad (44)$$

A long process of expanding and rearranging (43), (42) and  $\mathbf{u}^T \mathcal{A}_k \mathbf{u} = 0$ , using the equality  ${}^I \hat{\mathbf{p}}_{i_k}^{(0)} = {}^I \hat{\mathbf{p}}_k^{(0)} + \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)}$  allows to show, using the triangle and Young's inequalities,

that, if at least two landmarks are non collinear, matrix  $\mathcal{A}_k$  is invertible, provided that Assumption 1 is true. This yields

$$\Omega_k = \mathcal{A}_k^{-1} \mathbf{b}_k. \quad (45)$$

With  $\Omega_k$  determined, the next step is to compute its statistical properties, namely its expected value and covariance matrix. The expected value of the rotation error is

$$\langle \Omega_k \rangle = \mathcal{A}_k^{-1} \langle [c_{23} - c_{32} \quad c_{31} - c_{13} \quad c_{12} - c_{21}]^T \rangle = \mathbf{0},$$

as it is straightforward to show that  $\mathbf{C}_k$  has zero mean. As  $\Omega_k$  is a zero mean quantity, its covariance matrix is simply given by

$$\begin{aligned} \Sigma_{\Omega_k} &= \langle \Omega_k \Omega_k^T \rangle = \mathcal{A}_k^{-1} \langle \mathbf{b}_k \mathbf{b}_k^T \rangle \mathcal{A}_k^{-1T} \\ &= \mathcal{A}_k^{-1} \left[ \langle (\mathbf{C}_k - \mathbf{C}_k^T)^2 \rangle - \frac{1}{2} \text{tr} \langle (\mathbf{C}_k - \mathbf{C}_k^T)^2 \rangle \mathbf{I}_3 \right] \mathcal{A}_k^{-1T}. \end{aligned} \quad (46)$$

After a long computation that is omitted here due to lack of space, it is shown that the three components of  $\langle (\mathbf{C}_k - \mathbf{C}_k^T)^2 \rangle$  are given by

$$\begin{aligned} \langle \mathbf{C}_k \mathbf{C}_k \rangle &= \sum_{i,j=1}^{N_T} \sigma_{j_k}^{-2} \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} \hat{\mathbf{p}}_{j_k}^{(0)T} \hat{\mathbf{R}}_k^{(0)T} \langle \mathbf{a}_{i_k}^{(1)} \mathbf{a}_{j_k}^{(1)T} \rangle \right. \\ &\quad \left. + \hat{\mathbf{R}}_k^{(0)} \Sigma_{p_{ij_k}} \hat{\mathbf{R}}_k^{(0)T} \mathbf{a}_{i_k}^{(0)} \mathbf{a}_{j_k}^{(0)T} \right] \\ \langle \mathbf{C}_k \mathbf{C}_k^T \rangle &= \sum_{i,j=1}^{N_T} \sigma_{i_k}^{-2} \sigma_{j_k}^{-2} \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} \hat{\mathbf{p}}_{j_k}^{(0)T} \hat{\mathbf{R}}_k^{(0)T} \text{tr}(\mathbf{a}_{i_k}^{(1)} \mathbf{a}_{j_k}^{(1)T}) \right. \\ &\quad \left. + \hat{\mathbf{R}}_k^{(0)} \Sigma_{p_{ij_k}} \hat{\mathbf{R}}_k^{(0)T} \mathbf{a}_{i_k}^{(0)} \mathbf{a}_{j_k}^{(0)T} \right] \\ \langle \mathbf{C}_k^T \mathbf{C}_k \rangle &= \sum_{i,j=1}^{N_T} \sigma_{i_k}^{-2} \sigma_{j_k}^{-2} \left[ \hat{\mathbf{p}}_{i_k}^{(0)T} \hat{\mathbf{p}}_{j_k}^{(0)} \langle \mathbf{a}_{i_k}^{(1)} \mathbf{a}_{j_k}^{(1)T} \rangle \right. \\ &\quad \left. + \text{tr}(\Sigma_{p_{ij_k}}) \mathbf{a}_{i_k}^{(0)} \mathbf{a}_{j_k}^{(0)T} \right], \end{aligned} \quad (47)$$

with

$$\begin{aligned} \langle \mathbf{a}_{i_k}^{(1)} \mathbf{a}_{j_k}^{(1)T} \rangle &= \Sigma_{I_{p_{ij_k}}} + \frac{1}{N_{W_k}^2} \sum_{r,s=1}^{N_T} \sigma_r^{-2} \sigma_s^{-2} \Sigma_{I_{pr_s k}} \\ &\quad - \frac{1}{N_{W_k}} \sum_{r=1}^{N_T} \sigma_r^{-2} \left( \Sigma_{I_{p_{ir_k}}} + \Sigma_{I_{pr_j k}} \right), \end{aligned} \quad (48)$$

where  $\mathbf{a}_{i_k}^{(\cdot)} := I \hat{\mathbf{p}}_{i_k}^{(\cdot)} - \frac{1}{N_{W_k}} \sum_{j=1}^{N_T} I \hat{\mathbf{p}}_{j_k}^{(\cdot)}$  and it was assumed that the body-fixed landmarks are independent from the inertial landmarks, as the latter were calculated in a different time instant.

This finishes the derivations of the uncertainty related with the computation of the rotation between frames. The work proceeds by determining the uncertainty of the inertial map and trajectory estimations, starting by completing the characterization of the transformation between the sensor-based and inertial frames, that is, the uncertainty of the translation vector,  $I \hat{\mathbf{p}}_k$ .

2) *Translation uncertainty:* The translation between frames is, as explained before, given by the position of the vehicle in a given instant. Therefore, to determine its uncertainty it is necessary to recall the vehicle position estimate error model presented in (37c) as well as its definition in (33). The expansion of the latter using the former yields

$$\begin{aligned} I \mathbf{p}_k &= \frac{1}{N_{W_k}} \sum_{i=1}^N \sigma_{i_k}^{-2} \left[ \left( I \hat{\mathbf{p}}_{0_k}^{(i)} + \epsilon I \hat{\mathbf{p}}_{i_k}^{(1)} \right) \right. \\ &\quad \left. - \hat{\mathbf{R}}_k^{(0)} (\mathbf{I}_3 + \epsilon \mathbf{S}[\boldsymbol{\omega}_k]) \frac{1}{N_{W_k}} \sum_{i=1}^N \sigma_{i_k}^{-2} \left( \hat{\mathbf{p}}_{0_k}^{(i)} + \epsilon \hat{\mathbf{p}}_{i_k}^{(1)} \right) \right], \end{aligned}$$

from which, once more neglecting higher order terms, it is possible to see that  $I \hat{\mathbf{p}}_k^{(1)} = \frac{1}{N_{W_k}} \sum_{i=1}^{N_T} \sigma_{i_k}^{-2} \left( I \hat{\mathbf{p}}_{i_k}^{(1)} - \mathbf{S}[\Omega_k] \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} - \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(1)} \right)$ .

It is straightforward to confirm that  $I \hat{\mathbf{p}}_k^{(1)}$  has zero mean, and, after some computation, that the covariance matrix of the position estimate  $\Sigma_{I_{p_k}}$  is approximately given by

$$\begin{aligned} \Sigma_{I_{p_k}} &\approx \frac{1}{N_{W_k}^2} \sum_{i,j=1}^{N_T} \sigma_{i_k}^{-2} \sigma_{j_k}^{-2} \left( \Sigma_{I_{p_{ij_k}}} + \hat{\mathbf{R}}_k^{(0)} \Sigma_{p_{ij_k}} \hat{\mathbf{R}}_k^{(0)T} \right. \\ &\quad \left. + \mathbf{S} \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} \right] \Sigma_{\Omega_k} \mathbf{S}^T \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{j_k}^{(0)} \right] \right), \end{aligned} \quad (49)$$

where all the cross terms between inertial and sensor-based landmarks were omitted, as they are zero, and the cross covariance terms between the rotation error and the landmarks (both inertial and sensor-based) were neglected.

3) *Inertial map uncertainty:* The final step in the process of studying the uncertainty description of the algorithm is computing  $\Sigma_{I_{p_{ijk+1}}}$ , for all  $i, j \in \mathcal{I}_{I_{k+1}}$ . Recall that the inertial map estimate is calculated with the update equation in (36). Using (37d), it is possible to determine the uncertain part of the error model, given by  $I \hat{\mathbf{p}}_{i_{k+1}}^{(1)} = I \hat{\mathbf{p}}_{i_k}^{(1)} + \mathbf{S}[\Omega_k] \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} + \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(1)}$ . Again,  $I \hat{\mathbf{p}}_{i_{k+1}}^{(1)}$  is easily confirmed to have zero mean, and an approximation to the covariance matrix of the position estimate is obtained by neglecting the cross-covariance terms between the vehicle position, rotation and the landmarks:

$$\begin{aligned} \Sigma_{I_{p_{ijk+1}}} &\approx \Sigma_{I_{p_k}} + \hat{\mathbf{R}}_k^{(0)} \Sigma_{p_{ij_k}} \hat{\mathbf{R}}_k^{(0)T} \\ &\quad + \mathbf{S} \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} \right] \Sigma_{\Omega_k} \mathbf{S}^T \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{j_k}^{(0)} \right] \\ &\quad + \mathbf{S}^T \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{i_k}^{(0)} \right] \Sigma_{p_k \Omega_k}^T + \Sigma_{I_{p_k \Omega_k}} \mathbf{S} \left[ \hat{\mathbf{R}}_k^{(0)} \hat{\mathbf{p}}_{j_k}^{(0)} \right], \end{aligned} \quad (50)$$

where once more all the cross terms between inertial and sensor-based landmarks were omitted.  $\Sigma_{I_{p_k \Omega_k}} := \langle I \hat{\mathbf{p}}_k^{(1)} \Omega_k^T \rangle$  is the cross covariance between the translation and rotation estimates, whose columns are given by linear combinations of  $\langle I \hat{\mathbf{p}}_k^{(1)} c_{lm} \rangle$ ,  $l, m \in \{1, 2, 3\}$ . This covariance is approximately given by

$$\begin{aligned} \langle I \hat{\mathbf{p}}_k^{(1)} c_{lm} \rangle &\approx \frac{1}{N_{W_k}} \sum_{i,j=1}^{N_M} \sigma_{i_k}^{-2} \sigma_{j_k}^{-2} \left[ \Sigma_{I_{p_{ij_k}}}(:, m) \hat{\mathbf{R}}_k^{(0)}(l, :) \hat{\mathbf{p}}_{j_k}^{(0)} \right. \\ &\quad - \frac{1}{N_{W_k}} \sum_{r=1}^{N_M} \sigma_r^{-2} \Sigma_{I_{p_{ir_k}}}(:, m) \hat{\mathbf{R}}_k^{(0)}(l, :) \hat{\mathbf{p}}_{j_k}^{(0)} \\ &\quad \left. - \hat{\mathbf{R}}_k^{(0)} \Sigma_{p_{ij_k}} \left( \hat{\mathbf{R}}_k^{(0)}(l, :)^T \mathbf{a}_{j_k}^{(0)}(m) \right) \right], \end{aligned} \quad (51)$$

where all the usual assumptions were taken into account and where  $\mathbf{A}(l, :)$  represents the  $l$ -th row of the matrix  $\mathbf{A}$  and  $\mathbf{a}(m)$  the  $m$ -th component of  $\mathbf{a}$ .

#### D. Final Remarks

It is important to notice that in this procedure, an inertial landmark is only updated if the associated uncertainty decreases in that iteration. Thus, in each iteration, the candidate inertial landmarks covariance matrix is computed, and then the trace of each  $\Sigma_{I_{p_{ijk+1}}}$  is compared to its previous value,  $\text{tr}(\Sigma_{I_{p_{ijk}}})$ . If the uncertainty is raised, then the old covariance is kept and  $\Sigma_{I_{p_{ijk+1}}} = \mathbf{0}$  for all  $j \neq i$ .

The initial estimate is also a relevant point. The usual approach in any localization algorithm is to assume that the vehicle is deterministically at the origin and aligned with the inertial frame. Hence, when the algorithm is started, the position and attitude of the vehicle are fully known, thus allowing the computation of the inertial estimate at  $k = 0$  with  ${}^I\hat{\mathbf{p}}_{i_0} = \mathbf{R}_0\hat{\mathbf{p}}_{i_0} + {}^I\mathbf{p}_0$ . The only non-deterministic quantity here is the sensor-based landmark estimate, and, thus the uncertain part of  ${}^I\hat{\mathbf{p}}_{i_0}$  is simply  ${}^I\hat{\mathbf{p}}_{i_0}^{(1)} = \mathbf{R}_0\hat{\mathbf{p}}_{i_0}^{(1)}$ . The initial covariance matrix that describes the uncertainty associated with the initial estimate is then given by

$$\Sigma_{I p_{i_0}} = \langle {}^I\hat{\mathbf{p}}_{i_0}^{(1)} {}^I\hat{\mathbf{p}}_{i_0}^{(1)T} \rangle = \mathbf{R}_0 \Sigma_{p_{i_0}} \mathbf{R}_0^T, \quad (52)$$

thus ending the theoretical description of the algorithm.

## VI. THE COMPLETE ALGORITHM

The complete algorithm proposed involves two different main algorithms that complement each other: the Sensor-based SLAM Filter on one side, and the Inertial Map and Trajectory Estimation on the other. This was done to separate the inertial estimates from the filtering process, thus allowing to filter in the sensors space, avoiding the representation of the inertial pose in the filter state, and still providing inertial estimates through the second algorithm. The diagram of Fig. 1 represents the flow of information in the algorithm used in the experimental setup.

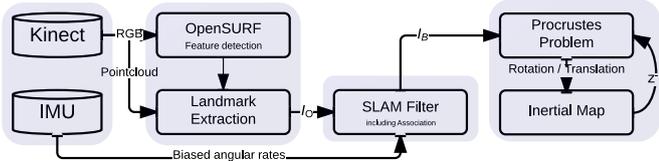


Figure 1. The full algorithm used in the experiments.

1) *Tunable parameters*: The SLAM filter, as any Kalman filter, has tunable parameters, namely the model disturbance noise covariance, the measurement noise covariance and the filter initial conditions. Furthermore, the algorithm here proposed divides the estimated map in subsets of landmarks depending on the last time they were observed, raising the need for time thresholds. The loop closure and state maintenance procedures also have tuning knobs - the first time they occur and the number of landmarks needed to trigger an action. The set of parameters is chosen according to physical aspects, but, nevertheless, was adapted to each situation - with synthesized or real data (shown in parenthesis when different).

The output noise covariance is  $\Theta_k = \sigma_\theta^2 \mathbf{I}_3$ ,  $\sigma_\theta = 0.032$  m (0.022 m), the state disturbance covariance is given by  $\Xi_k = T_s \text{diag}(\sigma_v^2 \mathbf{I}_3, \sigma_{b_\omega}^2 \mathbf{I}_3, \sigma_{p_1}^2 \mathbf{I}_3, \dots, \sigma_{p_N}^2 \mathbf{I}_3)$ , with  $\sigma_v = 0.05$  m/s,  $\sigma_{b_\omega} = 10^{-5}$  rad/s and  $\sigma_{p_i} = 10^{-4}$  m (0.1 m). The initial state covariance was set to  $\Sigma_0 = \text{diag}(\sigma_{v_0}^2 \mathbf{I}_3, \sigma_{b_{\omega_0}}^2 \mathbf{I}_3)$ , with  $\sigma_{v_0} = 0.011$  m/s (0.039 m/s) and  $\sigma_{b_{\omega_0}} = 0.022$  rad/s. Note that at  $k = 0$  there are no landmarks in the state, but any new landmark is initialized with  $\Sigma_{p_{i_0}} = \sigma_{p_{i_0}}^2 \mathbf{I}_3$ ,  $\sigma_{p_{i_0}} = 0.017$  m (0.039 m). Finally, the initial estimates of velocity and angular bias were set to zero in the simulation and to the average of a preliminary acquisition when using real data. As to the SLAM-specific parameters, the recent landmark set is composed by landmarks seen in the last 15 s (0.5 s), and the old by landmarks not seen in more than 100 s (15 s). The first loop closure is tried at 100 s and it is triggered if at least

6 landmarks are associated. Finally, any landmark not visible for more than 200 s is discarded.

## VII. SIMULATION RESULTS

The simulated environment consists of 70 landmarks spread throughout a  $16\text{m} \times 16\text{m} \times 3\text{m}$  map, including a closed 2m wide corridor in outer borders of the map. The trajectory is simply a loop through the corridors at half-height, with the vehicle starting on the floor.

The simulation starts with the vehicle stopped for 50 seconds, which then takes off and circles through the map for around 230 seconds at an average speed of 0.45 m/s. The zero-mean noise added to the angular velocity measurements is normal-distributed with a standard deviation of  $\sigma_{\omega_m} = 5 \times 10^{-4}$  rad/s at each coordinate and the noise included in the landmark observations is also zero-mean Gaussian white noise with a standard deviation of  $\sigma_y = 10^{-3}$  m.

Firstly, the Kalman filter performance can be evaluated through Fig. 2 and Fig. 3. The former depicts the evolution of 5 landmarks, with Fig. 2(b) showing the standard deviation of each sensor-based landmark growing to around 1m, when a loop closure is triggered at  $t = 190$  s and the uncertainty diminishes considerably. In Fig. 2(a) the estimation error is shown.

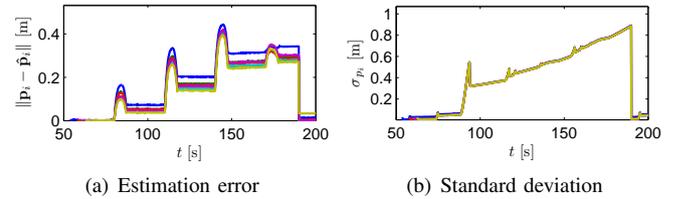


Figure 2. Estimation error and standard deviation of the first 5 landmarks.

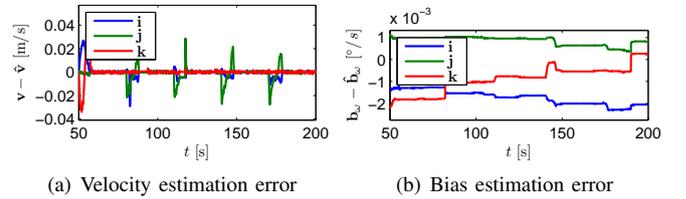


Figure 3. Error and standard-deviation of the vehicle related estimates.

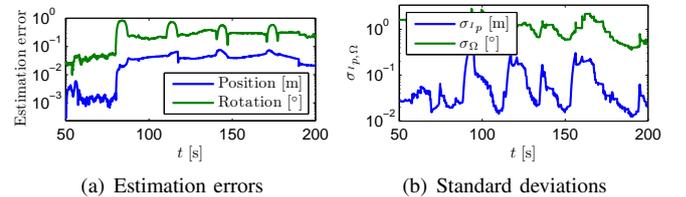


Figure 4. Error and standard-deviation of the position and rotation matrix estimates.

The *esTIMATE* algorithm can be assessed through Fig. 4, where the estimation error (Fig. 4(a)) and the standard deviation (Fig. 4(b)) of the position and attitude of the vehicle is shown.

The statistics regarding the number of landmarks can be found in Fig. 5. In blue, the number of landmarks in the state of

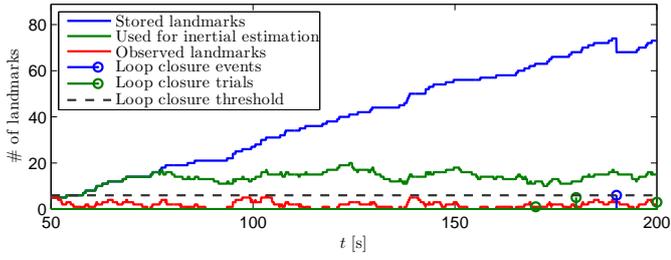


Figure 5. Evolution of the number of landmarks used in the Kalman filter (in blue), the number of landmarks used in the *esTIMATE* algorithm (in green), and the number of visible landmarks (in red). Loop closure trials, threshold and events also present.

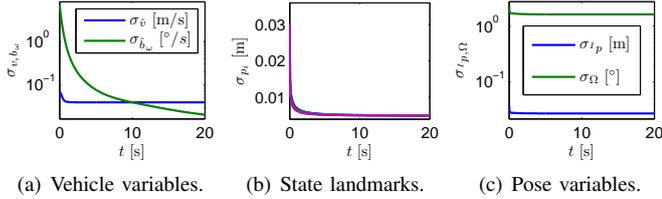


Figure 6. Evolution of the standard deviations of the vehicle variables, the position, and the rotation in the first 20 seconds of simulation.

the Kalman filter, in green the number of landmarks used in the inertial estimation, and in red the number of visible landmarks. The stems represent loop closure trials and events, and the dashed line the minimum number of landmark associations necessary for loop closing.

The convergence of the standard deviation of the state variables and of the pose of the vehicle when the observability conditions are satisfied is presented in Fig. 6. In these first 20 seconds of simulation, the vehicle is immobilized on the ground, and 5 different landmarks are visible.

Finally, the estimated map at  $t = 191$  s, 1 s after a loop closing event, rotated and translated using the true quantities along the true trajectory is shown in Fig. 7. The coloured ellipsoids represent the 95% uncertainty ( $2\sigma$ ) of each landmark and the red star denotes the position of the vehicle at the time. Note that the older landmarks have greater uncertainty, and that the landmarks closer to the position of the vehicle (the ones more affected by the loop closure) have low uncertainty.

This simulation was designed to allow the practical validation of the consistency of the algorithm, by exposing the vehicle to previously visited terrain after exploring new areas, in order to trigger a loop closing. The results show that the sensor-based map is consistent, allowing the loop to be closed repeatedly (see Fig. 5). Moreover, the simulation results here presented demonstrate that the uncertainty is coherent with

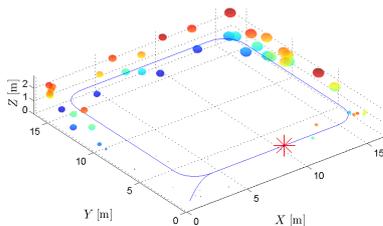


Figure 7. Sensor-based map at  $t = 191$  s, 1 second after a loop closure.

the estimation errors, as shown in Fig. 2, and verify the convergence when the observability conditions are satisfied.

## VIII. PRELIMINARY EXPERIMENTAL RESULTS

The simulation results were consolidated by a preliminary experiment at the Sensor-Based Cooperative Robotics Research Laboratory - SCORE Lab of the Faculty of Science and Technology of the University of Macau. The experi-



Figure 8. The *AscTec Pelican* equipped with a *Microsoft Kinect*, a *Microstrain 3DM-GX3-25* and the *VICON* markers.

mental setup consists of an *AscTec Pelican* quadrotor (see Fig. 8), which is equipped with an *Intel Atom* processor board, and into which was added a *Microstrain 3DM-GX3-25* inertial measurement unit working at 200Hz and a *Microsoft Kinect* camera, at 30Hz. The experiment consisted in moving the quadrotor inside a  $6\text{m} \times 6\text{m}$  room (usable area of  $16\text{m}^2$ ). The room was equipped with a *VICON* motion capture system, which provides accurate estimates of the position, attitude, linear and angular velocities of any vehicle placed inside the working area with the correct markers.

In the first 15 seconds the vehicle was stopped and in the remaining time it was moved in a small lap around the room. Figure 9 shows the comparison of the estimation of the inertial trajectory (in red) with the ground truth provided by the *VICON*, in blue. It can be seen that the estimated trajectory follows very closely the true trajectory of the vehicle, never being more than 20 centimeters off, except after the first 30 seconds, where less observations were available as will be seen in Fig. 12.

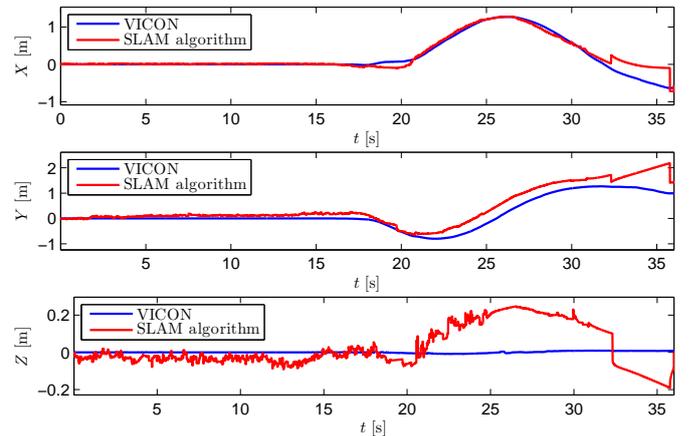


Figure 9. Time evolution of the real and estimated trajectory

Figure 10 compares the ground truth with the estimation of the inertial linear velocity of the vehicle. The latter is obtained by rotating the sensor-based estimate with the estimated rotation. Again, the estimation follows within reasonable accuracy the ground truth, and, once more, it worsens after the 30 seconds, for the reasons explained. Note that the *VICON* estimates of the velocity are obtained by differentiation of

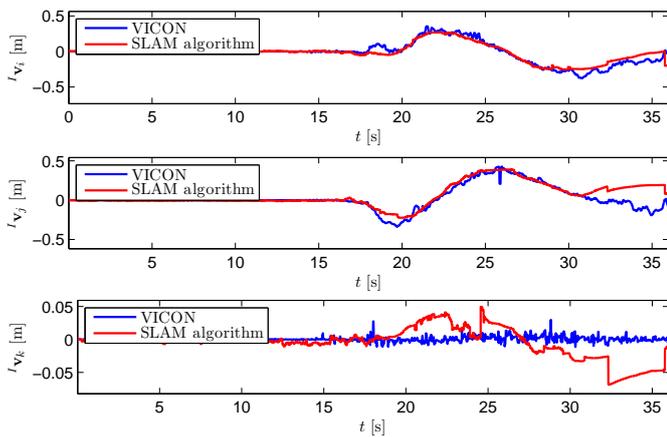


Figure 10. Time evolution of the real and estimated inertial velocity

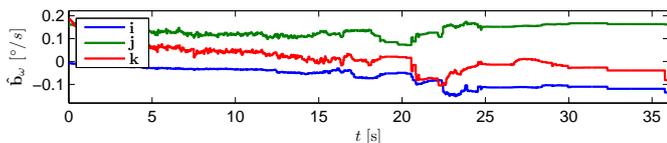


Figure 11. Time evolution of the angular measurement bias estimation.

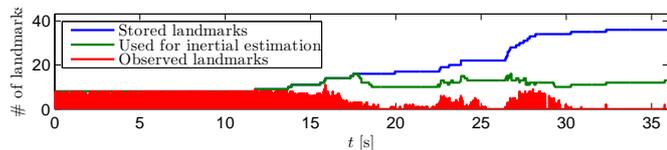


Figure 12. Evolution of the number of landmarks used in the Kalman filter (in blue) and the visible (in green).

the position, thus being somewhat noisy. This allows the evaluation of both the estimation of the sensor-based velocity and of the rotation matrix. The remaining product of the Kalman filter, the angular bias estimation, is then shown in Fig. 11.

Finally, the evolution of the number of landmarks involved in the algorithm are shown. In blue, the number of landmarks in the SLAM filter state is presented, in green is shown the number of landmarks used by the *esTIMATE* algorithm to compute the optimal transformation, and the number of visible landmarks in each observation instant is shown in red. Note that, after the first 30 seconds, the refresh rate of the observations is reduced drastically, due to technical issues with the equipment, while the number of landmarks observed is also small. This explains the degradation in the estimation that occurs around that time.

## IX. CONCLUSIONS AND FUTURE WORK

This paper presented a novel algorithm for Simultaneous Localization and Mapping, reporting the design, analysis, implementation, and validation of this algorithm. The work proposed has various contributions, including a novel sensor-based filter and an uncertainty-wise fully characterized optimization orthogonal Procrustes problem. The filtering framework was designed in the sensors space, thus avoiding the attitude representation in the filter state. This enabled the establishment of observability results that lead to uniformly globally asymptotically stable error dynamics. Furthermore,

the orthogonal Procrustes problem was addressed, defining an optimization problem coupled with a full statistical description.

The first part of the work focused on the observability analysis, which provided theoretical results in observability and, subsequently, on the convergence of the error dynamics of the proposed nonlinear system. Furthermore, the performance and consistency of the algorithm were validated in simulation showing the convergence of the uncertainty in every variable except the non-visible landmarks, as well as the production of a consistent map which allows the closure of a loop with nearly 50 meters. Preliminary experimental results, with ground truth data, showed also the good performance of the SLAM algorithm as a whole.

Future work will address the inclusion of landmark directions, both in the observability analysis and landmark detection. Also in the observability analysis, the necessary conditions are yet to be found. Furthermore, there is information provided by the SURF algorithm, namely the feature descriptor [11], that may help improve the association process if availed. Finally, further experimental validation is recommended for scientific dissemination.

## REFERENCES

- [1] H. Durrant-Whyte and T. Bailey, "Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, 2006.
- [2] J. Castellanos, R. Martínez-Cantin, J. Tardós, and J. Neira, "Robocentric map joining: Improving the consistency of EKF-SLAM," *Robotics and Autonomous Systems*, vol. 55, no. 1, pp. 21–29, 2007.
- [3] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the RGB-D SLAM system," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, St. Paul, MA, USA, May 2012.
- [4] B. Guerreiro, P. Batista, C. Silvestre, and P. Oliveira, "Sensor-based Simultaneous Localization and Mapping - Part I: GAS Robocentric Filter," in *Proceedings of the 2012 American Control Conference*, Montréal, Canada, Jun. 2012, pp. 6352–6357.
- [5] —, "Sensor-based Simultaneous Localization and Mapping - Part II: Online Inertial Map and Trajectory Estimation," in *Proceedings of the 2012 American Control Conference*, Montréal, Canada, Jun. 2012, pp. 6334–6339.
- [6] P. Batista, C. Silvestre, and P. Oliveira, "Single range aided navigation and source localization: Observability and filter design," *Systems & Control Letters*, vol. 60, no. 8, pp. 665–673, 2011.
- [7] R. Brockett, *Finite dimensional linear systems*, ser. Series in decision and control. John Wiley & Sons, 1970.
- [8] P. Batista, C. Silvestre, and P. Oliveira, "Optimal position and velocity navigation filters for autonomous vehicles," *Automatica*, vol. 46, no. 4, pp. 767–774, 2010.
- [9] B. Anderson, "Stability properties of Kalman-Bucy filters," *Journal of the Franklin Institute*, vol. 291, no. 2, pp. 137–144, 1971.
- [10] P. Batista, C. Silvestre, and P. Oliveira, "Sensor-based complementary globally asymptotically stable filters for attitude estimation," in *Proceedings of the 48th IEEE Conference on Decision and Control*, Shanghai, China, 2009, pp. 7563–7568.
- [11] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008, similarity Matching in Computer Vision and Multimedia.
- [12] J. Neira and J. Tardós, "Data association in stochastic mapping using the joint compatibility test," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 6, pp. 890–897, dec 2001.
- [13] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions On Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.
- [14] B. K. P. Horn, H. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," *Journal of the Optical Society America*, vol. 5, no. 7, pp. 1127–1135, 1988.
- [15] C. Goodall, "Procrustes Methods in the Statistical Analysis of Shape," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 53, no. 2, pp. 285–339, 1991.
- [16] R. Sibson, "Studies in the robustness of multidimensional scaling: Per-turbational analysis of classical scaling," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 41, no. 2, pp. 217–229, 1979.