Simultaneous Localization and Mapping with Moving Object Tracking for Drones

Manuel Simas
Instituto Superior Técnico
manuel.simas@tecnico.ulisboa.pt

***Abstract*—This thesis addresses the problem of simultaneous localization and mapping with moving object tracking (SLAMMOT) with application to unmanned aerial vehicles in uncertain and dynamic environments. The first part of the thesis presents the formulation and design of an inertial-based filter for the SLAMMOT problem. The proposed solution includes the implementation of a modified version of the multiple hypothesis tracking method for data association of the environment's objects, the use of the interacting multiple model algorithm for detection and tracking of moving objects (MO), and data fusion is performed using an extended Kalman filter, as a result of the nonlinear kinematics of the system. The consistency and performance of the devised SLAMMOT filter are successfully confirmed with both simulation and experimental results, using an instrumented quadrotor equipped with an RGB-D camera. The second part of the thesis presents the formulation and design of a novel sensor-based filter for the SLAMMOT problem, that builds on the same methods for data association, detection, and tracking of MO. However, due to the sensor-based approach, the system dynamics are formulated in such a way that can be regarded as linear for filter design purposes, from which a Kalman filter is applied for data fusion. The convergence and effectiveness of the devised SLAMMOT filter are successfully confirmed with simulation results.**

I. INTRODUCTION

Unmanned aerial vehicles (UAVs), commonly known as drones, are an emergent aircraft technology that operates by human remote control or autonomously. The drone's industry is facing exponential growth, with forecasts indicating a 100 billion dollar market opportunity for drones, between 2016 and 2020 [1]. The largest portion of this value, about 70%, is related to the military market, while the consumer market is rated at 17 billion dollar. The rising commercial market represents approximately 13% of the total amount. The business potential of drones includes the construction and agriculture industry, the police and fire fighting activities, pipelines inspection, and also mining tasks. Even though many of these applications are operated under human control, this form of labor may not always be available, creating the need for UAVs to perform autonomously [2].

Simultaneous localization and mapping (SLAM) algorithms have become immensely popular among the scientific community. The focus has been on the pursuit of increasing degrees of autonomy, in particular, finding a solution for the dual task of simultaneously creating and updating a map of an unknown environment while extracting information from the surroundings and estimating the vehicle's position within this map. There is a wide range of methods to solve the SLAM problem, with filter-based implementations having a great impact considering the numerous successful implementations [3]–[6]. Nonetheless, SLAM solutions have proven to be insufficient when applied to dynamic environments, as Wang and Thorpe show in [7]. Thus, simultaneous localization and mapping with moving object tracking (SLAMMOT) algorithms have revealed to be part of the solution to accurately perform in dynamic systems, since it considers and deals with the presence of moving objects (MO), with relevant implementations such as [8]–[10].

In SLAM and SLAMMOT literature, filter-based implementations are usually formulated under what can be denominated as an inertial-based approach. This framework is characterized by the representation of the filter state variables in the inertial reference frame, commonly designated as the world-fixed or Earth-fixed reference frame, although the measurements are acquired by the vehicle's sensors, and therefore are described in the sensor's reference frame. Thus, it is necessary to transform the measurements from the sensor's reference frame into the inertial frame, which involves the representation of the vehicle's attitude, using, for this purpose, rotation matrices. This nonlinear process is often linearized and, together with others linearizations, contributes to inconsistent results, as shown in [11]–[13] regarding the extended Kalman filter (EKF) or in [14] with respect to the FastSLAM implementation based on the particle filter. The research community has dedicated a significant effort to avoid these inconsistencies, with special attention to the work by Castellanos et al. [15], where the robocentric map joining algorithm is introduced. The authors proved, with simulation and experimental results, that with a local mapping strategy together with a robot centered representation of the state variables, it is possible to improve the consistency of SLAM. This seminal work gave rise to what is also known as sensor-based SLAM. The sensor-based framework is defined by representing the filter state variables in the sensor's reference frame, that is assumed to be aligned with the body-fixed reference frame. Hence, linearizations, as those previously mentioned, are avoided, thus improving the consistency of the filter. Additional implementations include the work presented in [16], [17]. The work developed by Guerreiro et al. [18] is highlighted, where the nonlinear system dynamics are formulated as a linear time-varying (LTV) system enabling the use of a Kalman filter (KF), rather than an EKF, obtaining a globally asymptotically stable (GAS) filter.

The contribution of this thesis is focused on the design, analysis, and validation of two different solutions to the SLAMMOT problem, in particular, to address the existence of static and dynamic elements in the environment and investigate in which ways the information of these elements, together with inertial sensors and vehicle dynamics information, can be fused together in order to obtain a solution with convergence guarantees that enable its practical application. Withal, the major contribution of this dissertation is the formulation of a novel approach for the SLAMMOT problem, based on the sensor-based framework. Moreover, the results of this thesis led to the following publication:

1) M. Simas, B. J. Guerreiro, and P. Batista, "Preliminary Results on 2-D Simultaneous Localization and Mapping for Aerial Robots in Dynamics Environments," in Proceedings of the 7th International Conference on Robot Intelligence Technology and Applications - ICRiTA 2019, Daejeon, Korea, Nov. 2019 (accepted for publication).

The remainder of this document is structured as follows. Section II and Section III described in detail the formulation of the inertial-based and the sensor-based solutions, respectively. Simulation results which confirm the effectiveness of both approaches are presented in Section IV. The performance of the inertial-based solution is confirmed with experimental results as shown in Section V. Conclusions and future work are discussed in section VI.

A. Notation

Throughout the document, the terms SO and landmark are used interchangeably, the symbols I_n and $0_{n\times m}$ denote, respectively, the identity matrix with size $n \times n$ and a matrix of zeros with size $n \times m$, while $\mathrm{diag}(\mathbf{M}_1, \ldots, \mathbf{M}_n)$ represents a block diagonal matrix. When the dimensions are omitted, it is assumed that the matrices have the appropriate dimensions. Additionally, $\mathbf{S}\big[a\big]$ encodes the 2-D skew-symmetric matrix, i.e., $\mathbf{S}\big[a\big] = \begin{bmatrix} 0 & a \\ -a & 0 \end{bmatrix}$, with $a \in \mathbb{R}$.

II. An Inertial-based solution to SLAMMOT

This section presents in detail the formulation and design of an inertial-based filter for the SLAMMOT problem. The full system is described and includes the vehicle's and objects' dynamics, and the measurement model. Moreover, the filter design is presented, along with the selected methods to perform data association, motion model identification, loop closure and data fusion.

A. System Dynamics

Let W denote the world-fixed reference frame, B the drone body-fixed reference frame, ${}^W\mathrm{p}_B(t) \in \mathbb{R}^2$ the position of B, described in W, and $\mathrm{v}_B(t) \in \mathbb{R}^2$ the linear velocity of the vehicle relative to W, expressed in B. The linear motion kinematics of the drone is given by

$$ {}^W\dot{\mathrm{p}}_B(t) = \mathrm{v}_B(t), \tag{1} $$

with

$$ \mathrm{v}_B(t) = \mathrm{v}(t) \begin{bmatrix} \cos(\psi(t)) \\ \sin(\psi(t)) \end{bmatrix}, \tag{2} $$

where $\mathrm{v}(t) \in \mathbb{R}$ and $\psi(t) \in \mathbb{R}$ represent, respectively, the linear velocity norm and yaw of the drone relative to W, expressed in B.

A dynamic environment is defined by the existence of both SO and MO. Contrarily to SO, whose position is constant throughout time, MO can describe multiple motion models. The proposed formulation, considers that both types of objects describe linear motion models, with SO describing a constant position (CP) model and the MO only describing a constant velocity (CV) model. Let ${}^W\mathrm{p}_i(t) \in \mathbb{R}^2$ denote the position of the i-th object, described in W. Thus, the kinematics of the considered models are given by

$$ {}^W\dot{\mathrm{p}}_i(t) = 0 \tag{3} $$

if the considered object describes a CP model, or

$$ {}^W\ddot{\mathrm{p}}_i(t) = 0 \tag{4} $$

if, in contrast, the object describes a CV model.

The proposed filter requires the acquisition of a series of measurements over time, with respect to the surroundings and the drone's motion. In this formulation, the velocity norm and the yaw are inputs to the system and are denoted by the control vector $\mathrm{u}(t) = \begin{bmatrix} \mathrm{v}(t) & \psi(t) \end{bmatrix}^T \in \mathbb{R}^2$. These variables are acquired by odometry sensors, corrupted by noise, as given by

$$ \mathrm{v}_m(t) = \mathrm{v}(t) + \mathrm{n}_{\mathrm{v}_m}(t) \tag{5} $$

$$ \psi_m(t) = \psi(t) + \mathrm{n}_{\psi_m}(t) \tag{6} $$

with $\mathrm{v}_m(t) \in \mathbb{R}$ and $\psi_m(t) \in \mathbb{R}$ representing, respectively, the sensors measurements of the velocity norm and yaw of the drone. The associated disturbances, $\mathrm{n}_{\mathrm{v}_m}(t) \in \mathbb{R}$ and $\mathrm{n}_{\psi_m}(t) \in \mathbb{R}$, are assumed to be zero-mean white Gaussian noise with standard deviation σ_{v_m} and σ_{ψ_m}, respectively. The information with respect to the surroundings is acquired by measurements that describe the relative position between the position of the drone and environment's objects position, both represented in the world-fixed reference frame. Let $\mathrm{z}_i \in \mathbb{R}^2$ denote the measure with regard of the i-th object, hence, the measurement model is given by

$$ \mathrm{z}_i(t) = ({}^W\mathrm{p}_i(t) - {}^W\mathrm{p}_B(t)) + \mathrm{n}_z(t), \tag{7} $$

that it is also corrupted by zero-mean white Gaussian noise, $\mathrm{n}_z(t) \in \mathbb{R}^2$, with standard deviation σ_{z} in each component.

In this SLAMMOT formulation, the full state vector comprises the information regarding the drone and all the detected objects, whether they are static or moving, described in the world-fixed reference frame. The drone's position is the only variable included on the drone's state space vector and the objects state space vector is composed by the objects' position and velocity, yielding, respectively, $\mathrm{x}_V(t) \coloneqq {}^W\mathrm{p}_B(t)$ and $\mathrm{x}_{O_i}(t) \coloneqq \begin{bmatrix} {}^W\mathrm{p}_i^T(t) & {}^W\mathrm{v}_i^T(t) \end{bmatrix}^T$, with ${}^W\mathrm{v}_i(t) \in \mathbb{R}^2$ denoting the velocity of the i-th object, described in W. Consequently,

with n observed objects, the full state space vector is represented by $\mathrm{x}(t) \coloneqq \begin{bmatrix} \mathrm{x}_V^T(t) & \mathrm{x}_{O_1}^T(t) & \dots & \mathrm{x}_{O_n}^T(t) \end{bmatrix}^T \in \mathbb{R}^{2+4n}$.

B. Filter Design

This section describes in detail the design of the proposed inertial-based SLAMMOT filter. This includes the description of the techniques used for data association, MO detection and tracking, and loop closure. Moreover, the former system is inherently nonlinear, specifically the drone's dynamics (1), thus, data fusion is performed with a discrete-time EKF. Accordingly, the filter's discretization details are also presented.

1) Data Association: The target-measurement assignment problem consists of correctly associating, over time, each available measurement with the corresponding object. This data association process is required to make the SLAMMOT algorithm robust and consistent so that the estimates uncertainty and error may decrease. Moreover, it is essential for convergence of the motion model identification and tracking of MO, and also to enable loop closure. In this formulation, data association of both SO and MO is performed using a modified version of the multiple hypothesis tracking (MHT) method [19]. This method creates a tree of all the potential association hypotheses, considering the available measurements and also the old hypotheses, followed by the likelihood calculation for each hypotheses. This process is considered to be optimal since it contemplate every hypotheses in each step [20], however it involves a high computational cost, that grows exponentially. Consequently, to avoid this intensive memory dependency, it is necessary to apply pruning methods, such as, only preserving the most likely hypotheses at each step. Since this likelihood directly depends on the uncertainty of each known object, and the MO comprises high values of uncertainty, to avoid miss association between SO and MO, it was additionally imposed that a MO is only assigned to a given measurement if its likelihood is higher than a given threshold. Despite its inherent complexity, the MHT algorithm is a robust and effective method, as mentioned in [8], [21] and [22].

2) Moving Objects Detection and Tracking: In SLAMMOT, SO are the source of valuable information for the filter operation since their uncertainty is only associated with the sensors noise. On the other hand, MO have additional uncertainty that is related with their motion, and it is essential to detect and track these objects. Thus, accurate identification of the motion model described by the environment's objects is crucial to retrieve better data association of SO and tracking of MO, that consequently improve the filter estimates. In this formulation this task is accomplished using the interacting multiple model (IMM) algorithm [23], [24]. This algorithm uses r filters to estimate the state of r models, then computes the likelihood of a given object describe a given model, and finally mixes all this information into one estimate. Since only two models are considered for each object, CP and CV, the IMM algorithm uses two filters. Moreover, the mixing step is not performed and the full algorithm only uses the probabilities calculated by the IMM. In an extensive comparative study by Pitre et al. [25], results have shown that the IMM algorithm has a robust performance, among all the considered methods, when compared to its low computational complexity. Additionally, Li and Jilkov [26] refer to IMM algorithm as the standard tool for target tracking.

3) Loop Closure: The filter effectiveness to identify a previously visited place is pivotal to decrease the uncertainty and error regarding the landmarks and the vehicle's estimations. The loop closure problem is not considered to be the main focus of this thesis, but instead the formulation and validation of a filter implementation that addresses the existence of dynamic objects in the environment, resulting in consistent results that enable the use of any loop closure technique. In the tested scenarios, the combination of the MHT and IMM algorithms provide enough accuracy in the estimates and their uncertainty, enabling the filter to detect and associate an old landmark with a new measurement, after one loop. Nonetheless, in larger and more complex scenarios, this consistency may not hold, and techniques, as [27] or [28], should be implemented.

4) Extended Kalman Filter: A discrete-time EKF was designed for data fusion purposes. Every time that a measurement is available, the MHT algorithm performs data association, the motion model of the objects is defined by the IMM algorithm, and finally the filter compute the estimations. Otherwise, the system is propagated in open loop. Consider $\mathrm{x}_k \coloneqq \mathrm{x}(t_k)$, $\mathrm{z}_k \coloneqq \begin{bmatrix} \mathrm{z}_1^T(t_k) & \dots & \mathrm{z}_n^T(t_k) \end{bmatrix}^T$ and $\mathrm{u}_k \coloneqq \mathrm{u}(t_k)$, with $t_k = t_0 + kT_s$, $k \in \mathbb{N}_0$ and t_0 as the initial time. Thus, the forward Euler discretization of the system dynamics yields

$$\begin{cases} \mathrm{x}_{k+1} = f(\mathrm{x}_k, \mathrm{u}_{k+1}) + \mathrm{w}_{k+1} \\ \mathrm{z}_{k+1} = h(\mathrm{x}_{k+1}) + \mathrm{n}_{k+1} \end{cases}, \tag{8}$$

where f is the state transition function, obtained via (1) and (3) and/or (4), and h is the observation model function, obtained using (7), with $\mathrm{w}_k \in \mathbb{R}^{2+4n}$ and $\mathrm{n}_k \in \mathbb{R}^{2n}$ representing the correspondent process and measurement noise, which are assumed to be zero-mean, discrete-time white Gaussian noise, uncorrelated and time-invariant. The prediction equations handle the full state vector, however, the objects state vectors are only propagated based on the transition matrix of the motion model that the object describes. Therefore, the prediction equations are

$$\begin{cases} \widehat{\mathrm{x}}_{k+1|k} = f(\widehat{\mathrm{x}}_{k|k}, \mathrm{u}_{k+1}) \\ \Sigma_{k+1|k} = \mathbf{F}_{k+1} \Sigma_{k|k} \mathbf{F}_{k+1}^T + \Xi_{k+1} \end{cases}, \tag{9}$$

where $\widehat{\mathrm{x}}_{k|k}$ denotes the estimated state vector and $\Sigma_{k|k}$ the estimated covariance matrix. The process noise covariance matrix is defined as $\Xi_{k+1} = \mathrm{diag}(\Xi_V, \Xi_O)$ with $\Xi_V \in \mathbb{R}^{2\times 2}$ and $\Xi_O \in \mathbb{R}^{4n\times 4n}$ representing the process noise covariance matrix regarding the vehicle and objects dynamics, respectively. The former, is defined as $\Xi_V = \Theta \Lambda \Theta^T$, with $\Lambda = \mathrm{diag}(\sigma_{\mathrm{v}}^2, \sigma_{\psi}^2)$, with σ_{v} and σ_{ψ} denoting the standard deviation of the process noise concerning the velocity norm and yaw of the drone, respectively. The jacobian matrix of the system equation with respect to the process noise is defined as $\Theta = \left.\frac{\partial f}{\partial u}\right|_{u_{k+1}}$. On the other hand, the objects' process noise

matrix is described as $\Xi_O = \mathrm{diag}(\Xi_{O1}, \ldots, \Xi_{On})$, with Ξ_{On} representing the process noise of each individual object. This matrix adapts according to the motion model described by each object, thus, if an object is describing a CP model, the matrix is given as $\Xi_{On} = T_s \mathrm{diag}(\sigma^2_{\mathrm{p}_{so}} \mathrm{I}_2, \sigma^2_{\mathrm{v}_{so}} \mathrm{I}_2)$, whereas if the object is describing a CV model, the matrix is defined as $\Xi_{On} = T_s \mathrm{diag}(\sigma^2_{\mathrm{p}_{mo}} \mathrm{I}_2, \sigma^2_{\mathrm{v}_{mo}} \mathrm{I}_2)$, with $\sigma_{\mathrm{p}_{so}}$, $\sigma_{\mathrm{v}_{so}}$, $\sigma_{\mathrm{p}_{mo}}$, and $\sigma_{\mathrm{v}_{mo}}$, denoting the the process noise standard deviations of the position and velocity, respectively, of SO and MO. Additionally, the full transition matrix is defined as $\mathbf{F}_k = \mathrm{diag}(\mathbf{F}_V, \mathbf{F}_O)$, with $\mathbf{F}_V$ denoting the vehicle transition matrix, given by $\mathbf{F}_V = \mathrm{I}_2$. The transition matrix of the objects is given by $\mathbf{F}_O = \mathrm{diag}(\mathbf{F}_{O1}, \ldots, \mathbf{F}_{On})$ with $\mathbf{F}_{On}$ representing the transition matrix of each individual object. For objects describing a CP model, it is given by $\mathbf{F}_{On} = \mathrm{diag}(\mathrm{I}_2, 0_{2\times 2})$ and for those that describe a CV model, it is given by

$$\mathbf{F}_{On} = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{10}$$

Every time that a set of object position measurements is available, the measurements vector z_k is updated. If the measurements are with respect to new objects the state vector is augmented, otherwise only the measurements vector is changed. Then, the update step occurs and follows the standard EKF update equations, given as

$$\begin{cases} \widetilde{\mathrm{y}}_{k+1} = \mathrm{z}_{k+1} - h(\widehat{\mathrm{x}}_{k+1|k}) \\ S_{k+1} = H_{k+1} \Sigma_{k+1|k} H^T_{k+1} + \Gamma_{k+1} \\ K_{k+1} = \Sigma_{k+1|k} H^T_{k+1} S^{-1}_{k+1} \end{cases}, \tag{11}$$

where the measurement residual is defined as $\widetilde{\mathrm{y}}_{k+1}$, S_{k+1} represent the residual covariance, K_{k+1} is the Kalman gain, and $\Gamma_{k+1} \in \mathbb{R}^{2n\times 2n}$ expresses the measurement noise covariance. Then, the updated estimate can be computed using

$$\begin{cases} \widehat{\mathrm{x}}_{k+1|k+1} = \widehat{\mathrm{x}}_{k+1|k} + K_{k+1} \widetilde{\mathrm{y}}_{k+1} \\ \Sigma_{k+1|k+1} = (\mathrm{I} - K_{k+1} H_{k+1}) \Sigma_{k+1|k} \end{cases}, \tag{12}$$

where $\widehat{\mathrm{x}}_{k+1|k+1}$, $\Sigma_{k+1|k+1}$ and $H_k \in \mathbb{R}^{2n\times 2+4n}$ denote, respectively, the updated state, the update covariance matrix and the measurement matrix.

III. A Sensor-based solution to SLAMMOT

This section presents in detail the formulation and design of a sensor-based filter for the SLAMMOT problem. The full system is described and includes the vehicle's and objects' dynamics, and the measurement model. Moreover, the filter design is presented, along with the selected methods to perform data association, motion model identification, loop closure and data fusion.

A. System Dynamics

Let W denote the world-fixed reference frame, B the body-fixed reference frame and ${}^W_B\mathbf{R}(t) \in \mathrm{SO}(2)$ encode the rotation matrix from B to W. Let also ${}^W\mathrm{p}_B(t) \in \mathbb{R}^2$ denote the position of B, described in W, and $\mathrm{v}_B(t) \in \mathbb{R}^2$ the linear velocity of the vehicle relative to W, expressed in B. The linear motion kinematics of the drone is given by

$$\begin{cases} {}^W\dot{\mathrm{p}}_B(t) = {}^W_B\mathbf{R}(t)\ \mathrm{v}_B(t) \\ {}^W_B\dot{\mathbf{R}}(t) = {}^W_B\mathbf{R}(t)\ \mathbf{S}\big[\omega(t)\big] \end{cases}, \tag{13}$$

where ${}^W_B\mathbf{R}(t) = \mathbf{R}(\psi(t))$, with $\psi(t)$ representing the yaw angle of the drone and $\omega(t) \in \mathbb{R}$ represents the z-component of the angular velocity of the drone relative to W, expressed in B. In this formulation, the objects' dynamics and respective motion models, described in the world-fixed reference frame, are the same as proposed in the previous formulation (Section II). In the sensor-based framework the filter is designed in the space of the sensors and therefore the objects position, as well as their kinematics, are represented in B. Consider that $\mathrm{p}_i(t) \in \mathbb{R}^2$ denotes the i-th object's position described in B, given by

$$\mathrm{p}_i(t) = {}^W_B\mathbf{R}^T(t)\ ({}^W\mathrm{p}_i(t) - {}^W\mathrm{p}_B(t)). \tag{14}$$

Taking the derivative of both sides of this equation, the motion kinematics of the i-th object in the body frame are given by

$$\dot{\mathrm{p}}_i(t) = -\mathbf{S}\big[\omega(t)\big]\ \mathrm{p}_i(t) - \mathrm{v}_B(t) + {}^W_B\mathbf{R}^T(t)\ {}^W\mathrm{v}_i(t), \tag{15}$$

where ${}^W\mathrm{v}_i(t) \in \mathbb{R}^2$ represents the linear velocity of the i-th object, relative to W. The proposed filter requires measurements of the vehicle's angular velocity and attitude, as well as measurements regarding the environment objects position since these are used in the system dynamics and are crucial to guarantee the convergence of the filter. Considering that the inertial measurement unit (IMU) of the vehicle is aligned with its body-fixed reference frame, the angular velocity measurements $\omega_m(t) \in \mathbb{R}$, disturbed by noise and bias, is given by

$$\omega_m(t) = \omega(t) + \mathrm{b}(t) + \mathrm{n}_{\omega_m}(t), \tag{16}$$

where $\mathrm{b}(t) \in \mathbb{R}$ represents the slowly time-varying bias and $\mathrm{n}_{\omega_m}(t) \in \mathbb{R}$ denotes the disturbance noise, which is assumed to be white Gaussian noise, uncorrelated and time-invariant, with zero mean and standard deviation σ_{ω_m}. The measurements regarding the attitude of the drone ${}^W_B\mathbf{R}_m(t)$ are provided by an attitude and heading reference system (AHRS), also corrupted by noise, as given by

$${}^W_B\mathbf{R}_m(t) = {}^W_B\mathbf{R}(\psi(t) + \mathrm{n}_{\psi_m}(t)), \tag{17}$$

where $\mathrm{n}_{\psi_m}(t) \in \mathbb{R}$ expresses the measurement noise, which is also assumed to be white Gaussian noise, with standard deviation σ_{ψ_m}. In conformity with the sensor-based approach, the position measurements represent the relative position of the environment objects with regard to the drone's position, described in B. Let $\mathrm{y}_i(t) \in \mathbb{R}^2$ denote the measure of the i-th object. Thus, the measurement model is given by

$$\mathrm{y}_i(t) = \mathrm{p}_i(t) + \mathrm{n}_{\mathrm{y}}(t), \tag{18}$$

where $\mathrm{n_y}(t) \in \mathbb{R}^2$ is the sensor's white Gaussian noise, with standard deviation σ_y in each component. Accordingly, via (3) and (15), the motion kinematics of the SO can be written as

$$\dot{\mathrm{p}}_i(t) = -\big[\omega_m(t) - \mathrm{b}(t)\big]\ \mathbf{S}\ \mathrm{p}_i(t) - \mathrm{v}_B(t), \tag{19}$$

with $\mathbf{S} := \mathbf{S}\big[1\big]$. Additionally, the MO motion kinematics can be deduced from (4) and (15), yielding

$$\begin{aligned}\dot{\mathrm{p}}_i(t) = &-\big[\omega_m(t) - \mathrm{b}(t)\big]\ \mathbf{S}\ \mathrm{p}_i(t) - \mathrm{v}_B(t) \\ &+ {}^{W}_{B}\mathbf{R}^T_m(t)\ {}^{W}\mathrm{v}_i(t) \end{aligned}. \tag{20}$$

In this SLAMMOT formulation, the full state vector $\mathrm{x}_F(t) = \begin{bmatrix}\mathrm{x}_V^T(t) & \mathrm{x}_M^T(t)\end{bmatrix}^T \in \mathbb{R}^{n_F}$ encompasses the drone related variables, defined by $\mathrm{x}_V(t) \in \mathbb{R}^{n_V}$ and also information regarding the detected objects, for either static and moving objects, represented by $\mathrm{x}_M(t) \in \mathbb{R}^{n_M}$. The vehicle's state does not include any variables described in the world-fixed reference frame, but the drone's linear velocity and the rate gyros bias, all described in B, i.e. $\mathrm{x}_V(t) = \begin{bmatrix}\mathrm{v}_B^T(t) & \mathrm{b}(t)\end{bmatrix}^T$. The full objects' state can be decomposed into each individual object state that contains the objects' position and linear velocity, described, respectively, in B and W, defined by $\mathrm{x}_i(t) = \begin{bmatrix}\mathrm{p}_i^T(t) & {}^{W}\mathrm{v}_i^T(t)\end{bmatrix}^T$. Moreover, this vector can be reorganized so that the first n_O objects denote the ones that are observed and the next n_U represent the unobserved objects, such that $\tau_O = \{1, \ldots, n_O\}$, $\tau_U = \{n_O + 1, \ldots, n_O + n_U\}$ and $\tau = \{\tau_O, \tau_U\}$, and therefore, $\mathrm{x}_M = \begin{bmatrix}\mathrm{x}_O^T(t) & \mathrm{x}_U^T(t)\end{bmatrix}^T$, where $\mathrm{x}_O(t) = \{\mathrm{x}_i(t)\} \in \mathbb{R}^{n_O}$ for all $i \in \tau_O$, and $\mathrm{x}_U(t) = \{\mathrm{x}_i(t)\} \in \mathbb{R}^{n_U}$ for all $i \in \tau_U$. Accordingly, the full system dynamics can be written as

$$\begin{cases} \dot{\mathrm{x}}_V(t) = 0 \\ \dot{\mathrm{x}}_i(t) = \mathbf{A}_{V_i}(\mathrm{x}_i(t))\ \mathrm{x}_V(t) + \mathbf{A}_{M_i}(t)\ \mathrm{x}_i(t)\ \forall_{i\in\tau} \\ \mathrm{y}_i(t) = \mathrm{C_i}\ \mathrm{x}_i(t)\ \forall_{i\in\tau_O} \end{cases}, \tag{21}$$

with $\mathrm{C_i} = \mathrm{diag}(\mathrm{I}_2, 0_{2\times 2})$ and

$$\mathbf{A}_{V_i}(\mathrm{x}_i(t)) = \begin{bmatrix} -\mathrm{I}_2 & \mathbf{S}\ \mathrm{p}_i(t) \\ 0_{2\times 2} & 0_{2\times 2} \end{bmatrix}, \tag{22}$$

$$\mathbf{A}_{M_i}(t) = \begin{bmatrix} -\omega_m(t)\ \mathbf{S} & {}^{W}_{B}\mathbf{R}^T_m(t) \\ 0_{2\times 2} & 0_{2\times 2} \end{bmatrix}, \tag{23}$$

Note that in (23), since the velocity of the SO is zero, the term ${}^{W}_{B}\mathbf{R}^T_m(t)$ has no influence in these objects' dynamics. Moreover, for the purpose of filter design, notice that $\mathrm{y}_i(t)$ is available, enabling the use of the time-varying matrix $\mathbf{A}_{V_i}(\mathrm{y}_i(t))$, in which case the system can be considered as linear in the state.

B. Filter Design

This filter is build on the same techniques for data association, MO detection and tracking, and loop closure, used in the previous formulation (Section II). Distinctly, the system was formulated in such a way that it can be considered as linear in the state, and therefore, a KF was implemented for data fusion. Accordingly, this section describes all the filter's discretization details.

1) Kalman Filter: A discrete-time KF was designed for data fusion purposes. Every time that a measurement is available, the MHT algorithm performs data association, the motion model of the objects is defined by the IMM algorithm, and finally the filter compute the estimations. Otherwise, the system is propagated in open loop. Consider that $t_k = t_0 + kT_s$, $k \in \mathbb{N}_0$ with t_0 as the initial time and T_s denoting the sampling period. Thus, with $\mathrm{y}(t)$ representing the full output vector, denoting the state vector as $\mathrm{x}_k := \begin{bmatrix}\mathrm{x}_V^T(t_k) & \mathrm{x}_O^T(t_k)\end{bmatrix}^T$, and $\mathrm{C}_k := \mathrm{diag}(\mathrm{C}_1, \ldots, \mathrm{C}_{n_O})$ the discretization of the system dynamics yields

$$\begin{cases} \mathrm{x}_{k+1} = \mathbf{A}_k \mathrm{x}_k + \mathrm{w}_k \\ \mathrm{y}_{k+1} = \mathrm{H}_{k+1}\mathrm{x}_{k+1} + \mathrm{n}_{k+1} \end{cases}, \tag{24}$$

with $\mathrm{H}_k := \begin{bmatrix}0_{n_O\times n_V} & \mathrm{C}_k\end{bmatrix}$ and $\mathbf{A}_k = \mathbf{A}(t_k, \mathrm{y}(t_k))$, such that

$$\mathbf{A}(t_k, \mathrm{y}(t_k)) := \begin{bmatrix} \mathrm{I}_{n_V} & 0_{n_V\times n_O} \\ \mathbf{A}_V(\mathrm{y}(t_k)) & \mathbf{A}_M(t_k) \end{bmatrix}, \tag{25}$$

where

$$\mathbf{A}_V(\mathrm{y}(t_k)) := \big[\mathbf{A}^T_{V_1}(\mathrm{y}_1(t_k)), \ldots, \mathbf{A}^T_{V_{n_O}}(\mathrm{y}_{n_O}(t_k))\big]^T, \tag{26}$$

$$\mathbf{A}_M(t_k) := \mathrm{diag}(\mathbf{A}_{M_1}(t_k), \ldots, \mathbf{A}_{M_{n_O}}(t_k)), \tag{27}$$

and

$$\mathbf{A}_{V_i}(\mathrm{y}_i(t_k)) = \begin{bmatrix} -\mathrm{I}_2\ T_s & T_s\ e^{-\omega_m(t_k)\ \mathbf{S}\ T_s}\ \mathrm{p}_i(t_k)\ \mathbf{S} \\ 0_{2\times 2} & 0_{2\times 2} \end{bmatrix}, \tag{28}$$

$$\mathbf{A}_{M_i}(t_k) = \begin{bmatrix} e^{-\omega_m(t_k)\ \mathbf{S}\ T_s} & T_s\ {}^{W}_{B}\mathbf{R}^T_m(t_k) \\ 0_{2\times 2} & T_s\ \mathcal{M} \end{bmatrix}, \tag{29}$$

with $\mathcal{M} = 0_{2\times 2}$ if the i-th object is a SO or, otherwise, $\mathcal{M} = \mathrm{I}_{2\times 2}$ if it is a MO. Moreover, $\mathrm{w}_k \in \mathbb{R}^{n_V + n_O}$ and $\mathrm{n}_k \in \mathbb{R}^{n_O}$ represent, respectively, the process and measurement noise, which are assumed to be zero-mean, discrete-time white Gaussian noise, uncorrelated and time-invariant. Note that in (28) and (29) special care was given to the discretization of the rotational dynamics present in the objects' position dynamics, while the discretization of the vehicle and objects' velocity dynamics is performed using the forward Euler discretization. The prediction equations handle the full state vector, however, the system (24) only considers the observable objects. Hence, using the full system state $\mathrm{x}_{F_k} := \mathrm{x}_F(t_k)$ the prediction equations are

$$\begin{cases} \widehat{\mathrm{x}}_{F_{k+1|k}} = \mathbf{F}_k \widehat{\mathrm{x}}_{F_{k|k}} + \mathrm{w}_k \\ \Sigma_{F_{k+1|k}} = \mathbf{F}_k \Sigma_{F_{k|k}} \mathbf{F}_k^T + \Xi_k \end{cases}, \tag{30}$$

where $\widehat{\mathrm{x}}_{F_{k|k}}$ and $\Sigma_{F_{k|k}}$ denote, respectively, the estimated full state vector and the estimated full covariance matrix. The full transition matrix $\mathbf{F}_k$ can be inferred from the system dynamics (21), extending the matrices defined in (26) and (27) with the objects in τ_U, while the process noise covariance matrix is defined as $\Xi_k = \mathrm{diag}(\Xi_V, \Xi_O)$ with Ξ_V and Ξ_O

representing, respectively, the process noise covariance matrix with respect to the vehicle and objects' dynamics. The former is defined as $\Xi_V = T_s \text{diag}(\sigma^2_{\text{v}_B} \text{I}_2, \sigma^2_b)$, with $\sigma_{\text{v}_B} \in \mathbb{R}$ and $\sigma_b \in \mathbb{R}$ representing the process noise standard deviations, regarding the velocity of the body and gyro rate bias, respectively. The latter, Ξ_O, is given by $\Xi_O = \text{diag}(\Xi_{O_1}, \ldots, \Xi_{O_{n_M}})$ with $\Xi_{O_{n_M}}$, representing the process noise of each individual object, either observed or unobserved. This matrix also changes according to the motion model of each object and, for SO is specified as $\Xi_{O_{n_M}} = T_s \text{diag}(\sigma^2_{\text{p}_{so}} \text{I}_2, \sigma^2_{\text{v}_{so}} \text{I}_2)$, whereas for MO is defined as $\Xi_{O_{n_M}} = T_s \text{diag}(\sigma^2_{\text{p}_{mo}} \text{I}_2, \sigma^2_{\text{v}_{mo}} \text{I}_2)$, with $\sigma_{\text{p}_{so}}$, $\sigma_{\text{v}_{so}}$, $\sigma_{\text{p}_{mo}}$, and $\sigma_{\text{v}_{mo}}$, denoting the the process noise standard deviations of the position and velocity, respectively, of SO and MO. Every time that a set of object position measurements is available, the measurements vector y_{k+1} is updated. If the measurements contain new objects, the state vector is augmented, otherwise only the measurements vector is changed. Then, the update step occurs and follows the standard KF update equations, given as

$$\begin{cases} \widetilde{\text{y}}_{k+1} = \text{y}_{k+1} - \text{H}_{k+1} \widehat{\text{x}}_{F_{k+1|k}} \\ S_{k+1} = H_{k+1} \Sigma_{F_{k+1|k}} H^T_{k+1} + \Gamma_{k+1} \\ K_{k+1} = \Sigma_{F_{k+1|k}} H^T_{k+1} S^{-1}_{k+1} \end{cases}, \tag{31}$$

where $\text{y}_k = \text{y}(t_k)$, $\widetilde{\text{y}}_{k+1}$ and S_{k+1} represent, respectively, the measurement residual and the residual covariance, K_{k+1} is the Kalman gain, and the measurement noise covariance matrix is represented by $\Gamma_{k+1} \in \mathbb{R}^{2n_O \times 2n_O}$. Then, the updated estimate can be computed using

$$\begin{cases} \widehat{\text{x}}_{F_{k+1|k+1}} = \widehat{\text{x}}_{F_{k+1|k}} + K_{k+1} \widetilde{\text{y}}_{k+1} \\ \Sigma_{F_{k+1|k+1}} = (\text{I} - K_{k+1} H_{k+1}) \Sigma_{F_{k+1|k}} \end{cases}, \tag{32}$$

where $\widehat{\text{x}}_{F_{k+1|k+1}}$ and $\Sigma_{F_{k+1|k+1}}$ denote the updated state and covariance matrix, respectively.

IV. Simulation Results

This chapter describes the simulation details and presents the results for each proposed solution. These results show the effectiveness and consistency of each formulation. All simulations have been performed in MATLAB R2018b.

A. Environment

The simulated environment consists of 34 SO, reproducing existent corners along the trajectory path, and 4 different MO, with constant velocities and orientations. The simulation time is 165 seconds, and the drone is considered to be static during 50 seconds, wherein after it starts to move at an average speed of 0.5 m/s. This stationary period is a necessary condition for the sensor-based implementation so that the gyro rate bias estimation converges since this estimate is essential for the remaining filter estimations. The vehicle performs a circular path, detecting a total of 32 SO, with all MO being detected on the first lap. It is considered a sampling frequency of 100 Hz for the odometry, rate gyro and AHRS sensors, while the sensor that provides measurements of the surroundings works at 20 Hz. The drone's trajectory and all objects are represented in Fig. 5.

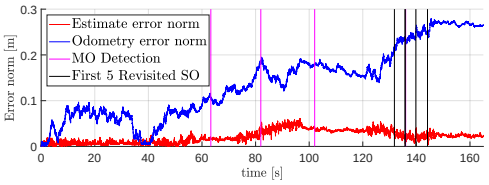


Fig. 1. Inertial-based: Estimation and odometry error norm, for complete time

B. Inertial-based Solution

This section describes the simulation details and presents the results for the inertial-based SLAMMOT solution, and the interpretation of these results. The simulated system disturbances are as follow, $\sigma_{\text{v}_m} = 0.15$ m/s, $\sigma_{\psi_m} = 3$ deg, and $\sigma_{\text{z}} = 2.5$ cm.

1) Filter Parameters: For data fusion, the drone's process noise standard deviation is assumed to be $\sigma_{\text{v}} = 0.3$ m/s and $\sigma_{\psi} = 3$ deg, while for SO it is given by $\sigma_{\text{p}_{so}} = 2.5$ mm and $\sigma_{\text{v}_{so}} = 0$ m/s, and for MO is considered to be $\sigma_{\text{p}_{mo}} = 1.5$ m and $\sigma_{\text{v}_{mo}} = 4$ m/s. The measurement noise covariance matrix is given as $\Gamma_{k+1} = 2\sigma^2_{\text{z}} \text{I}_{2n}$. When a object is observed for the first time, the uncertainty, in each direction, regarding its position and velocity is initialized with σ^2_{z}. In order to avoid an exponential growth of the number of hypothesis generated by the MHT, at each step, only the two hypothesis with the highest probability are kept.

2) Results: The overall filter performance can be evaluated through Fig. 1, which shows that the estimation error norm is considerably lower than the odometry error norm, particularly while the drone is stationary and after 130 seconds of simulation, where multiple and consecutive loop closure occurrences start to happen.

In Fig 2 and Fig 3 the standard deviation and estimation error of the nine revisited SO is shown, respectively. When the objects are detected for the first time their uncertainty is initialized, for instance, around $t = 60$ s, but soon decreases while measurements regarding these objects are available. The uncertainty increases when the objects are not being observed, which is expected since there is process noise associated with the SO position. As soon as these objects are revisited the algorithm performs a loop closure, and, consequently, the uncertainty regarding these landmarks substantial decreases. Nonetheless, when this happens, the estimation error for some objects increases. This is justified by the fact that these landmarks have associated uncertainty, when loop closure occurs, which induces the filter to correct these estimations, using an estimate of the drone's position that is worst than in prior time. Nonetheless, notice that the estimation error norm is less than 3 cm.

The final map with all the estimates of the trajectories and final positions is shown in Fig. 5, while the final estimated velocities of two MO are compared with the true values in Fig. 4. These results compare well to the true values, with an effective estimation of the drone and objects' states. It is worth mention that the filter correctly identifies the MO in less then

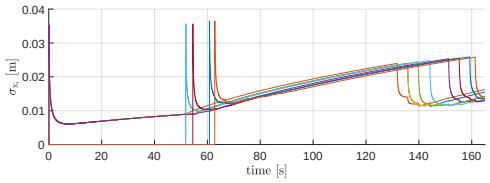


Fig. 2. Inertial-based: Standard deviation of the nine revisited SO positions (x-coordinate only), for complete time

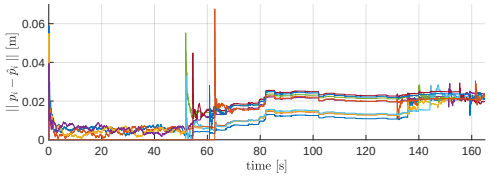


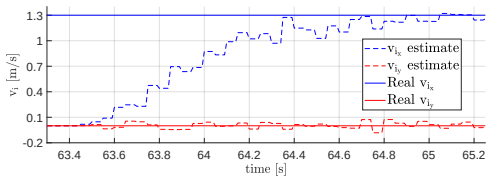
Fig. 3. Inertial-based: Error norm of the nine revisited SO positions, for complete time

0.5 seconds and the corresponding velocity estimate converges to the true value in less than 2 seconds.

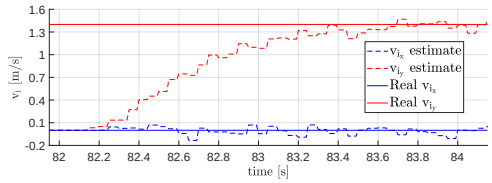
C. Sensor-based Solution

This section describes the simulation details and presents the results for the devised SLAMMOT sensor-based filter. These results aim to show the effectiveness, convergence and consistency of the filter based on the vehicle and objects' state and variance estimations. Apart from the vehicle velocity, all other state variables are described in the body-fixed frame. Nonetheless, using the closed-form algorithm proposed in [29] it is possible to compute the position of the vehicle and the objects described in the world-fixed reference frame. The simulated system disturbances are $\mathrm{b}(t) = 0.2$ deg/s, $\sigma_{\omega_m} = 3$ deg/s, $\sigma_{\psi_m} = 3$ deg, and $\sigma_{\mathrm{y}} = 2.5$ cm.

1) Filter Parameters: For data fusion, the drone's process noise standard deviation is assumed to be $\sigma_{\mathrm{v}_B} = 0.1$ m/s and $\sigma_b = 10^{-5}$ deg/s, while for SO it is given by $\sigma_{\mathrm{p}_{so}} = 2$ cm and $\sigma_{\mathrm{v}_{so}} = 0$ m/s, and for MO is considered to be $\sigma_{\mathrm{p}_{mo}} = 1.2$ m and $\sigma_{\mathrm{p}_{mo}} = 2.88$ m/s. The measurement noise covariance is



(a) MO 1.



(b) MO 2.

Fig. 4. Inertial-based: Comparison between the true and estimated linear velocity for two MO.

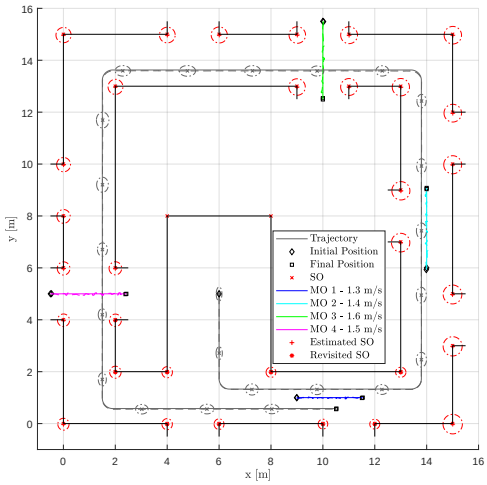


Fig. 5. Inertial-based: Simulated environment with true and estimated trajectories and objects. Dashed lines represent estimated trajectories and ovals represent the uncertainty multiplied by a factor of 200.

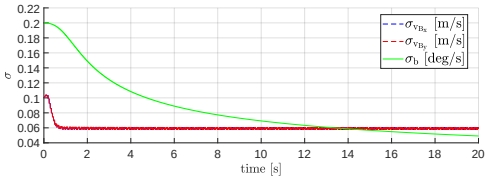


Fig. 6. Sensor-based: Standard deviation of the vehicle-related variables, for the first 20 seconds

given as $\Gamma_{k+1} = 2\sigma_{\mathrm{y}}^2 \mathrm{I}_{2n_O}$. When a object is observed for the first time, the uncertainty, in each direction, regarding its position and velocity is initialized with σ_{y}^2. In order to avoid an exponential growth of the number of hypothesis generated by the MHT, at each step, only the two hypothesis with the highest probability are kept.

2) Results: The overall performance and convergence of the filter can be evaluated through the analysis of the uncertainty associated with the state variables. The uncertainty regarding the vehicle state variables, for the first 20 seconds, is shown in Fig. 6, from which it is possible to notice the convergent behavior. The complete time evolution of these uncertainties can be seen in Fig 8 and 9. The uncertainty associated with the position of the nine revisited SO is presented in Fig. 7. It is possible to notice that the uncertainty increases when the objects are not being observed and depends on the orientation of the vehicle, which is expected. This behavior is desirable given that the uncertainty regarding the position of the vehicle is not estimated and, therefore, without this increase of uncertainty, it would not be possible to close any loop. Thus, as the old objects are observed, after a long period without any associated measurement, their related uncertainty decreases.

The drone's velocity and the bias estimates are shown, respectively, in Fig. 8 and Fig. 9. These results show that the filter can successfully estimate the drone's velocity without resorting any measurements related to these variables. The bias

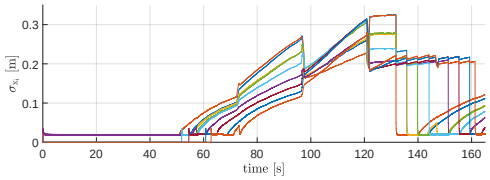
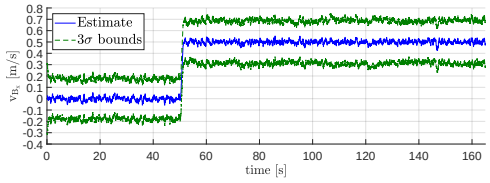
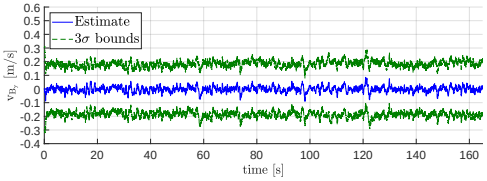


Fig. 7. Sensor-based: Standard deviation of the nine revisited SO positions (x-coordinate only), for complete time



(a) Vehicle x-coordinate velocity.



(b) Vehicle y-coordinate velocity.

Fig. 8. Sensor-based: Vehicle's linear velocity estimate, with 3σ bounds.

estimation is accurately accomplished, emphasizing that when the loop closure occurs, the bias estimation improves and the related uncertainty decreases.

To robustly avoid the effect of MO in the vehicle's variables estimation, it is essential that the filter, firstly, correctly identifies a MO, and then its velocity. Figure 10 shows both true and estimated velocity for all the MO. These results show that the filter can identify a MO in less than 0.2 seconds, and quickly converges to the correct correspondent velocity. The evolution of the number and type of objects in the state is shown in 11, which corresponds to the true number of SO and MO.

V. Experimental Results

This section describes the experimental results for the inertial-based solution. The experimental work was developed at ISR-Lisbon using an *Intel Aero RTF Drone*, equipped with a *Intel Atom* processor board and an *Intel RealSense R200 Camera*, working at 60 Hz.

A. Environment

The experiment was performed in a controlled room equipped with an OptiTrack motion capture system, working

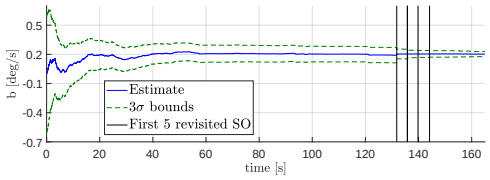
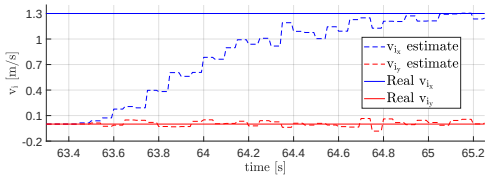
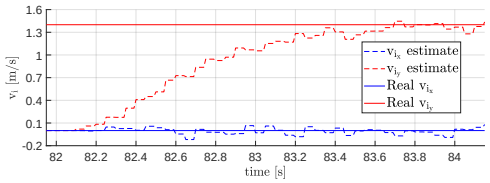


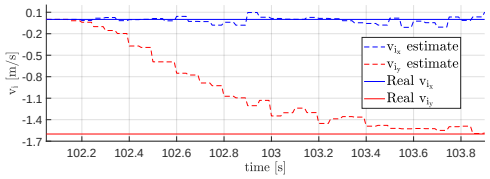
Fig. 9. Sensor-based: Gyro rate bias estimate, with 3σ bounds.



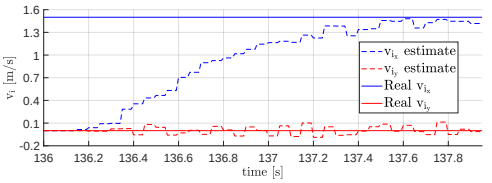
(a) MO 1.



(b) MO 2.



(c) MO 3.



(d) MO 4.

Fig. 10. Sensor-based: Comparison between the true and estimated linear velocity for all MO.

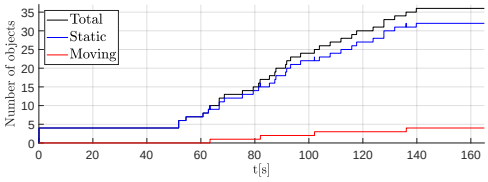


Fig. 11. Sensor-based: Evolution of the number and type of objects.

at 240 Hz. The quadrotor was hand-driven to perform a circular path, for about 70 seconds. During this trajectory, the drone captures RGB-D images of the environment in which some artificial markers were placed to facilitate the consecutive detection of these objects. Moreover, the drone captures an object that is moving and that stops moving at a certain point, becoming a static object. The surrounding measurements were obtained using the KAZE [30] algorithm to extract the strongest features of each RGB image. Then, the features were matched with the corresponding 3-D point on the depth image. In addition, this set of 3-D points that are represented on the body-fixed frame are changed to the world-fixed frame, applying a rotation based on the drone's attitude. The odometry of the drone, velocity norm and yaw, were computed using the positions provided by the OptiTrack system. Then, normal white Gaussian noise with zero-mean was added

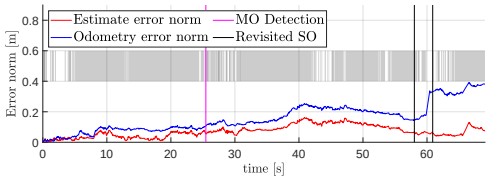


Fig. 12. Inertial-based: Estimation and odometry error norm, for complete time. Gray areas represent time with no measurements available.

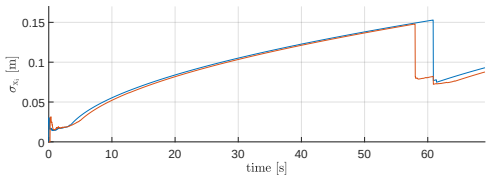


Fig. 13. Inertial-based: Standard deviation of the two revisited SO positions (x-coordinate only), for complete time.

to these variables, to simulate the odometry disturbance.

B. Inertial-based Solution

This section describes the experimental details and presents the results for the inertial-based SLAMMOT solution. The purpose of these results is to validate the simulation results and to confirm the effectiveness of the filter.

1) Filter Parameters: The system disturbances are $\sigma_{\mathrm{v}_m} = 0.10$ m/s and $\sigma_{\psi_m} = 10$ deg. No noise was added to the measurements concerning the surroundings since the data processing procedures already had associate disturbances. For data fusion, the measurement noise is assumed to be $\sigma_{\mathrm{z}} = 3.5$ cm, while the drone's process noise standard deviation is assumed to be $\sigma_{\mathrm{v}} = 0.2$ m/s and $\sigma_{\psi} = 20$ deg/s, while for SO it is given by $\sigma_{\mathrm{p}_{so}} = 2$ cm and $\sigma_{\mathrm{v}_{so}} = 0$ m/s, and for MO is considered to be $\sigma_{\mathrm{p}_{mo}} = 9$ cm and $\sigma_{\mathrm{v}_{mo}} = 18$ cm/s. The measurement noise covariance matrix is given as $\Gamma_{k+1} = \sigma_{\mathrm{z}}^2 \mathrm{I}_{2n}$. When a object is observed for the first time, the uncertainty, in each direction, regarding its position and velocity is initialized with $3\sigma_z^2$ and $100\sigma_z^2$, respectively. In order to avoid an exponential growth of the number of hypothesis generated by the MHT, at each step, only the two hypothesis with the highest probability are kept.

2) Results: The estimation and odometry error norm is shown in Fig. 12, where the estimation error is lower than the odometry error, confirming the simulation results. Around $t = 58$ s the drone starts to revisit a previous location and the algorithm successfully closes the loop, identifying the revisited landmark. These correct associations of previously seen landmarks contribute to a decrease in the uncertainty of these old landmarks, as shown in Fig. 13.

In Fig. 14 the evolution of the number and type of objects is shown, from which it is possible to note that the algorithm effectively identifies the MO and adjusts its model to CP when it stops moving.

The estimated trajectory and objects, with the corresponding uncertainties, are represented in Fig. 15, along with the true trajectory and the odometry estimate. Results show that the

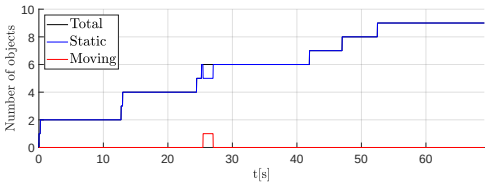


Fig. 14. Inertial-based: Evolution of the number of different objects.

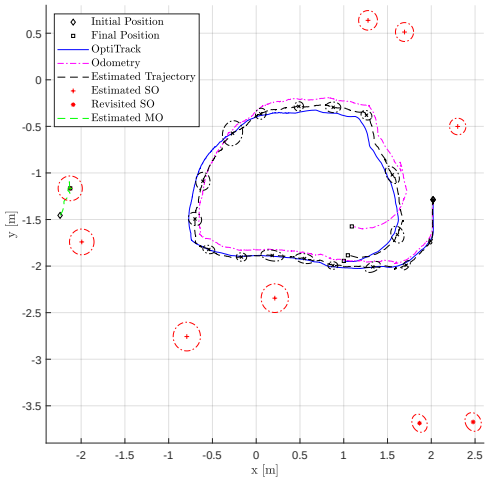


Fig. 15. Inertial-based: True, odometry and estimated trajectories, along with the objects estimates, with ovals representing the uncertainty.

revisited objects have lower uncertainty when compared to the other objects and that the estimate outperforms the odometry, especially by the end of the experiment.

VI. CONCLUSIONS

The aim of this thesis was to devise two different approaches for the SLAMMOT problem, in particular for aerial vehicles in uncertain and dynamic environments, and to provide both simulation and experimental results that validate the proposed formulations. Firstly, the formulation and design of an inertial-based solution for SLAMMOT were presented, that builds on the state of the art techniques for data fusion, data association, and detection and tracking of MO. The effectiveness and performance of the filter were then validated with simulation results and confirmed by a preliminary experiment. Then, a novel approach for the SLAMMOT problem was proposed. This solution was designed using the sensor-based framework and, as a result, linearizations in the system dynamics were avoided. Consequently, this formulation provided the necessary conditions for the use of a KF for data fusion purposes. Data association and detection and tracking of MO was accomplished by employing the same methods of the inertial-based solution. The performance and consistency of the proposed SLAMMOT filter were successfully validated with simulation results showing the filter is resilient to the presence of dynamic objects, even when those are included in the filter estimations. It is not reasonable to compare the performance of these approaches since each formulation had a

different measurement model, i.e., different information. While in the inertial-based formulation was assumed that the filter had access to odometry measurements, in the sensor-based formulation this was not included. Nonetheless, it was possible to confirm the consistency and effectiveness of both approaches. Future research includes the observability and convergence analysis to the devised sensor-based solution, addressing the necessary conditions for the filter convergence. Moreover, future results should include the experimental validation of the proposed filter. Concerning the implementation, further studies should investigate the possibility of incorporation more motion models into the IMM algorithm, for example, a constant acceleration (CA) model. The improvement of the pruning methods applied to the MHT algorithm would be very beneficial to reduce the computational cost of the data association without the loss of possibly relevant hypotheses, especially when applied in more complex environments. This can be accomplished by implementing the modifications suggested in [31] or by the implementation of a hybrid approach for data association, where SO are associated using a less computational expensive algorithm, such as the global nearest neighbor (GNN) [32] or the joint compatibility branch and bound (JCBB) [33] algorithms. Future developments should also include the implementation of an appropriate loop closure method to avoid inconsistencies when applied to larger scenarios. In addition, it is important to formulate both approaches in such a way that enables the direct comparison.

REFERENCES

[1] Goldman Sachs, "Drones Reporting for Work," 2016, retrieved 15 October, 2019, from https://www.goldmansachs.com/insights/technology-driving-innovation/drones/.

[2] V. M. Becerra, "Autonomous control of unmanned aerial vehicles," *Electronics*, vol. 8, no. 4, pp. 452–456, 2001.

[3] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on robotics and automation*, vol. 17, no. 3, pp. 229–241, 2001.

[4] J. E. Guivant and E. M. Nebot, "Optimization of the simultaneous localization and map-building algorithm for real-time implementation," *IEEE transactions on robotics and automation*, vol. 17, no. 3, pp. 242–257, 2001.

[5] M. Montemerlo and S. Thrun, "Simultaneous localization and mapping with unknown data association using fastslam," in *ICRA*, 2003, pp. 1985–1991.

[6] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, *et al.*, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *IJCAI*, 2003, pp. 1151–1156.

[7] C.-C. Wang and C. Thorpe, "Simultaneous localization and mapping with detection and tracking of moving objects," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 3. IEEE, 2002, pp. 2918–2924.

[8] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 889–916, 2007.

[9] T.-D. Vu, J. Burlet, and O. Aycard, "Grid-based localization and local mapping with moving object detection and tracking," *Information Fusion*, vol. 12, no. 1, pp. 58–69, 2011.

[10] D. F. Wolf and G. S. Sukhatme, "Mobile robot simultaneous localization and mapping in dynamic environments," *Autonomous Robots*, vol. 19, no. 1, pp. 53–65, 2005.

[11] S. J. Julier and J. K. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, vol. 4. IEEE, 2001, pp. 4238–4243.

[12] J. A. Castellanos, J. Neira, and J. D. Tardós, "Limits to the consistency of ekf-based slam," *IFAC Proceedings Volumes*, vol. 37, no. 8, pp. 716–721, 2004.

[13] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the ekf-slam algorithm," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 3562–3568.

[14] T. Bailey, J. Nieto, and E. Nebot, "Consistency of the fastslam algorithm," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 424–429.

[15] J. A. Castellanos, R. Martinez-Cantin, J. D. 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.

[16] A. N. Bishop and P. Jensfelt, "A stochastically stable solution to the problem of robocentric mapping," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 1615–1622.

[17] A. Burguera, Y. González, and G. Oliver, "Underwater slam with robocentric trajectory using a mechanically scanned imaging sonar," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 3577–3582.

[18] B. J. Guerreiro, P. Batista, C. Silvestre, and P. Oliveira, "Globally asymptotically stable sensor-based simultaneous localization and mapping," *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1380–1395, 2013.

[19] D. Reid, "An algorithm for tracking multiple targets," *IEEE transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[20] Y. Bar-Shalom and X.-R. Li, *Multitarget-multisensor tracking: principles and techniques*. YBs Storrs, CT, 1995, vol. 19.

[21] C. Rago, P. Willett, and R. Streit, "A comparison of the jpdaf and pmht tracking algorithms," in *1995 International conference on acoustics, speech, and signal processing*, vol. 5. IEEE, 1995, pp. 3571–3574.

[22] S. Jouannin, L. Trassoudaine, and J. Gallice, "Comparison of data association methods. application to road obstacle tracking using a doppler effect radar," *IFAC Proceedings Volumes*, vol. 31, no. 3, pp. 489–494, 1998.

[23] H. A. Blom, "An efficient filter for abruptly changing systems," in *The 23rd IEEE Conference on Decision and Control*. IEEE, 1984, pp. 656–658.

[24] H. A. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.

[25] R. R. Pitre, V. P. Jilkov, and X. R. Li, "A comparative study of multiple-model algorithms for maneuvering target tracking," in *Signal Processing, Sensor Fusion, and Target Recognition XIV*, vol. 5809. International Society for Optics and Photonics, 2005, pp. 549–560.

[26] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part v. multiple-model methods," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1255–1321, 2005.

[27] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *The International Journal of Robotics Research*, vol. 21, no. 4, pp. 311–330, 2002.

[28] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1271–1278.

[29] B. J. Guerreiro, P. Batista, C. Silvestre, and P. Oliveira, "Sensor-based simultaneous localization and mappingpart ii: Online inertial map and trajectory estimation," in *2012 American Control Conference (ACC)*. IEEE, 2012, pp. 6334–6339.

[30] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "Kaze features," in *European Conference on Computer Vision*. Springer, 2012, pp. 214–227.

[31] I. J. Cox and S. L. Hingorani, "An efficient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 18, no. 2, pp. 138–150, 1996.

[32] Y. Bar-Shalom, T. E. Fortmann, and P. G. Cable, "Tracking and data association," 1990.

[33] J. Neira and J. D. Tardós, "Data association in stochastic mapping using the joint compatibility test," *IEEE Transactions on robotics and automation*, vol. 17, no. 6, pp. 890–897, 2001.