11-4-2020

Cooperative Aerial Search and Localization Using Lissajous Patterns

Josiah Steckenrider
United States Military Academy, john.steckenrider@westpoint.edu

Spencer Leamy
Corvus Labs, spencer.leamy@corvuslabs.io

Tomonari Furukawa
University of Virginia, tf4rp@virginia.edu

Recommended Citation

J. J. Steckenrider, S. Leamy and T. Furukawa, "Cooperative Aerial Search and Localization Using Lissajous Patterns," 2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Abu Dhabi, UAE, 2020, pp. 233-240, doi: https://doi.org/10.1109/SSRR50563.2020.9292577

2020 IEEE International Symposium on
Safety, Security, and Rescue Robotics (SSRR)
November 4-6, Abu Dhabi, UAE

Cooperative Aerial Search and Localization Using Lissajous Patterns

J. Josiah Steckenrider[1], Spencer Leamy[2], and Tomonari Furukawa[3]

***Abstract*— This paper presents a cooperative aerial search-and-localization framework for applications where knowledge about the target of concern is minimal. The proposed framework leverages the sweeping oscillatory properties of Lissajous curves to improve an agent's chances of encountering a target. To accurately estimate the states of cooperative search drones, a discrete-time linear Lissajous motion model approximation is presented in such a way that uncertainty in physical model parameters can be accounted for. These uncertainties are propagated through estimation formulas to improve both agent and target localization relative to a static base station. Numerous experiments conducted in a physics-driven simulation environment show that Lissajous search patterns are a logical and effective substitute for many existing search pattern standards. Furthermore, parametric Monte Carlo simulation studies validate the proposed estimation framework as a more accurate target localizer than other traditional methods which do not account for inaccuracy in the motion model. These techniques hold promise for both static and dynamic target search-and-localization scenarios, allowing for robust estimation by eliminating the need for knowledge of low-level control input to search agents.**

I. INTRODUCTION

A. Background

Historically, aerial search missions have largely been executed manually, with emphasis placed on basic comprehensive area coverage. As autonomous systems become increasingly robust, however, considerations must also be given to higher-level tasks such as target localization, tracking, and engagement. The two focuses of this paper, autonomous search and localization, are both fundamental tasks for target engagement in applications ranging from rescue to defense. Recent years have seen the growing use of drones for such search-and-localization (SAL) missions, due to their ability to observe large areas of a landscape at once. In general, the target of an SAL mission may be fully or partially obscured, static or dynamic, near or far, or any combination of the above. Information on such a target, including its location and motion, may occasionally be obtained *a priori*. In most cases, however, target information is unavailable or outdated.

B. Related Work

Work related to autonomous cooperative drone search and tracking can be classified into a study of search pattern types and the development of information-theoretic techniques.

[1]Department of Civil and Mechanical Engineering, United States Military Academy, West Point NY, 10996, USA john.steckenrider@westpoint.edu
[2]Corvus Labs, LLC, Blacksburg VA, 24060, USA spencer.leamy@corvuslabs.io
[3]Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville VA, 22904, USA tomonari@virginia.edu

Area-coverage search patterns are effective when the target of concern is static or not well known. They may be determined by many factors including the geometry of the search space, the initial drone location, topological features of the environment, etc. [1]. Search patterns prescribed for aerial search-and-rescue (SAR) missions are commonly standardized by maritime and aeronautical authorities [2]. The International Maritime Organization (IMO) and International Civil Aviation Organization (ICAO) jointly put forward the International Aeronautical and Maritime Search and Rescue (IAMSAR) manual which includes specifications for several kinds of search patterns for use in various SAR scenarios [3] (see Fig. 1). While these approaches have been successfully used for many kinds of SAR problems, they focus more on comprehensive local coverage rather than pseudo-random sweep-searching over an entire area. As a result, the chances of "coincidental" detection are reduced and the likelihood that a dynamic target evades the searcher (intentionally or otherwise) is increased.

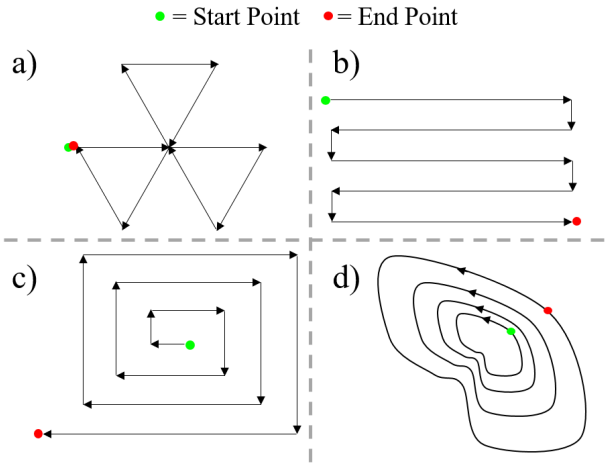

Fig. 1. IAMSAR search patterns: a) sector search, b) parallel sweep, c) expanding square, d) contour search.

Information-theoretic approaches, on the other hand, leverage known information about a target to search for and track it. Furukawa, et al. [4], [5] introduced the first robotic information-theoretic search-and-tracking approach by implementing grid-based non-Gaussian estimation and unifying the search and tracking objectives. Hoffman, et al. [6], [7] proposed mobile sensor network control using mutual information methods and particle filters. A Kalman filter (KF) based approach for coordinated control was first introduced by Grocholsky, et al. [8], and more recent work includes that

of Guanathillake, et al. [9] who developed a KF technique combined with a nonlinear least-squares method for decentralized target SAL. Mahler [10] developed a Probability Hypothesis Density (PHD) filter, and Dames [11] used it to estimate both the number of targets and their positions. The ideal estimation technique to use in a given context depend on the characteristics of the specific problem being addressed. Regardless of context, however, information-theoretic approaches are not effective when information on the target is unavailable or unreliable.

C. Proposed Approach

This paper presents a cooperative aerial SAL framework for contexts where little information is available on the target of concern. To maximize the chance of detecting such an unknown target, the proposed framework designs drone paths as Lissajous curves, which cover arbitrarily large rectangular search spaces smoothly at flexible resolutions. Deviation from planned paths due to motion and sensor uncertainties is handled through a specially augmented form of recursive Bayesian estimation (RBE) which is an extension of a Kalman filter. The target is locally observed by each search drone and globally transformed by a base station which utilizes belief fusion to enhance target estimation. The framework is not specific to any particular drone type, and could be deployed on fixed-wing or rotor-based systems.

The following section establishes fundamental formulations relating to the original contributions of this work, while section III describes these contributions in detail. A results section demonstrates the efficacy of both the Lissajous search pattern and the associated proposed estimation formulations. Finally, conclusions and future work are addressed in section V.

II. ESTIMATION, BELIEF FUSION, AND LISSAJOUS CURVES

A. Estimation

RBE is a framework which propagates state estimates probabilistically through prediction, observation, and correction stages. Under this framework, state belief is defined mathematically by arbitrary probability distribution functions (PDFs). The Kalman Filter is an RBE approach in which belief takes the form of a Gaussian distribution:

$$\mathcal{N}(\mathbf{x};\overline{\mathbf{x}},\mathbf{\Sigma}_x) = \frac{1}{\sqrt{|2\pi\mathbf{\Sigma}_x|}}\exp\Big(-\tfrac{1}{2}(\mathbf{x}-\overline{\mathbf{x}})^T\mathbf{\Sigma}_x^{-1}(\mathbf{x}-\overline{\mathbf{x}})\Big). \tag{1}$$

Since the Gaussian is completely defined by its mean vector $\overline{\mathbf{x}} \in \mathbb{R}^N$ and covariance matrix $\mathbf{\Sigma}_x$, the KF efficiently propagates only these parameters through RBE.

1) Prediction: The prediction stage propagates belief from step $k-1$ to step k based on some known motion model of the system at hand. Let $\mathbf{x}^d_{k-1}$ represent the state vector of a drone in N-D space at time step $k-1$. As a drone with autopilot moves smoothly through waypoint navigation, it is valid to describe its motion linearly. A linear motion model with state matrix $\mathbf{A}^d$, input matrix $\mathbf{B}^d$, and input vector $\mathbf{u}^d_{k-1}$ is given by:

$$\mathbf{x}^d_k = \mathbf{A}^d\mathbf{x}^d_{k-1} + \mathbf{B}^d\mathbf{u}^d_{k-1} + \mathbf{w}^d_{k-1}. \tag{2}$$

The vector $\mathbf{w}^d_{k-1}$ represents zero-mean Gaussian system noise with covariance $\mathbf{\Sigma}^d_{w_{k-1}}$, which makes prediction stochastic. The linear KF (LKF) propagates the mean vector of drone belief according to the following equation:

$$\overline{\mathbf{x}}^d_{k|1:k-1} = \mathbf{A}^d\overline{\mathbf{x}}^d_{k-1|1:k-1} + \mathbf{B}^d\mathbf{u}^d_{k-1}, \tag{3}$$

where the state subscripts convey a probabilistic estimate of a drone's state at a step given observations from step 1 to $k-1$. The covariance matrix corresponding to predicted drone belief at step k is then given by:

$$\mathbf{\Sigma}^d_{x_{k|1:k-1}} = \mathbf{A}^d\mathbf{\Sigma}^d_{x_{k-1|1:k-1}}{\mathbf{A}^d}^\top + \mathbf{\Sigma}^d_{w_{k-1}}. \tag{4}$$

2) Observation: Observation is governed by a sensor or observer model. If states are observed in their coordinate frame, the sensor model can also be assumed to be linear and Gaussian under the LKF. An observation $\mathbf{z}^d_k \in \mathbb{R}^M$ of the drone at step k can be expected to follow an observation model of the following form:

$$\mathbf{z}^d_k = \mathbf{C}^d\mathbf{x}^d_k + \mathbf{v}^d_k, \tag{5}$$

where $\mathbf{C}^d$ is known as the output matrix and $\mathbf{v}^d_k$ is zero-mean Gaussian sensor noise with covariance $\mathbf{\Sigma}^d_{v_k}$. Observations of a drone obtained from a sensor or observer at this stage are then input to the state correction stage below.

3) Correction: State correction combines belief coming from prediction with that coming from observation to give an improved estimate of the drone's state at step k. The mean vector and covariance matrix obtained from correction are given by:

$$\overline{\mathbf{x}}^d_{k|1:k} = \overline{\mathbf{x}}^d_{k|1:k-1} + \mathbf{K}^d(\mathbf{z}^d_k - \mathbf{C}^d\overline{\mathbf{x}}^d_{k|1:k-1}), \tag{6a}$$

$$\mathbf{\Sigma}^d_{x_{k|1:k}} = (\mathbf{I} - \mathbf{K}^d\mathbf{C}^d)\mathbf{\Sigma}^d_{x_{k|1:k-1}}, \tag{6b}$$

where

$$\mathbf{K}^d = \mathbf{\Sigma}^d_{x_{k|1:k-1}}{\mathbf{C}^d}^\top(\mathbf{C}^d\mathbf{\Sigma}^d_{x_{k|1:k-1}}{\mathbf{C}^d}^\top + \mathbf{\Sigma}^d_{v_k})^{-1}. \tag{7}$$

Finally, in order to re-seed the estimation algorithm, corrected belief is fed back to the prediction stage at the next time step: $\mathbf{x}^d_{k|1:k} \rightarrow \mathbf{x}^d_{k-1|1:k-1}$.

B. Belief Fusion

Belief fusion refers to the probabilistic combination of information about a state $\mathbf{x}$ coming from multiple independent sources $\mathbf{z}_i$. The PDF corresponding to fused belief of n constituent PDFs, $p(\mathbf{x}|\mathbf{z}_{1:n})$, is defined by their normalized product [12]:

$$p(\mathbf{x}|\mathbf{z}_{1:n}) = \frac{\prod_{i=1}^n p(\mathbf{x}|\mathbf{z}_i)}{\int_{\mathcal{X}}\prod_{i=1}^n p(\mathbf{x}|\mathbf{z}_i)d\mathbf{x}}. \tag{8}$$

When each constituent PDF $p(\mathbf{x}|\mathbf{z}_i)$ is Gaussian, it can be shown that $\prod_{i=1}^n p(\mathbf{x}|\mathbf{z}_i)$ is a non-normalized Gaussian distribution, so $p(\mathbf{x}|\mathbf{z}_{1:n})$ is Gaussian. For the fusion of two

Gaussians $p(\mathbf{x}|\mathbf{z}_1)$ and $p(\mathbf{x}|\mathbf{z}_2)$, $p(\mathbf{x}|\mathbf{z}_{1:2})$ is defined by:

$$p(\mathbf{x}|\mathbf{z}_{1:2}) = \mathcal{N}(\mathbf{x}; \overline{\mathbf{x}}_{1:2}, \mathbf{\Sigma}_{x_{1:2}}), \tag{9}$$

where

$$\mathbf{\Sigma}_{x_{1:2}} = \left[\mathbf{\Sigma}_{x_1}^{-1} + \mathbf{\Sigma}_{x_2}^{-1}\right]^{-1}, \tag{10a}$$

$$\overline{\mathbf{x}}_{1:2} = \mathbf{\Sigma}_{x_{1:2}}\left[\mathbf{\Sigma}_{x_1}^{-1}\overline{\mathbf{x}}_1 + \mathbf{\Sigma}_{x_2}^{-1}\overline{\mathbf{x}}_2\right]. \tag{10b}$$

Because belief fusion is associative, n PDFs can be fused in $n-1$ sequential pair-wise fusion steps. It can be shown that, when $\mathbf{C}^d = \mathbf{I}$, Eqs. (6)-(7) reduce to Eqs. (10).

C. Lissajous Curves

Lissajous functions describe curves which exhibit independent simple harmonic motion in two dimensions. The x and y coordinates of a Lissajous curve at any time t are defined by the following parametric equations:

$$x(t) = A_x \sin(\omega_x t + \varphi_x), \tag{11a}$$

$$y(t) = A_y \sin(\omega_y t + \varphi_y). \tag{11b}$$

These formulations can describe circles, ellipses, lemniscates, and more complex patterns. The amplitude parameters A_x and A_y govern the width and height of the rectangular space bounding a Lissajous pattern, while varying the ratio of ω_x to ω_y and the difference between φ_x and φ_y results in functional changes. Figure 2 demonstrates four examples of Lissajous curves corresponding to different parameter values.

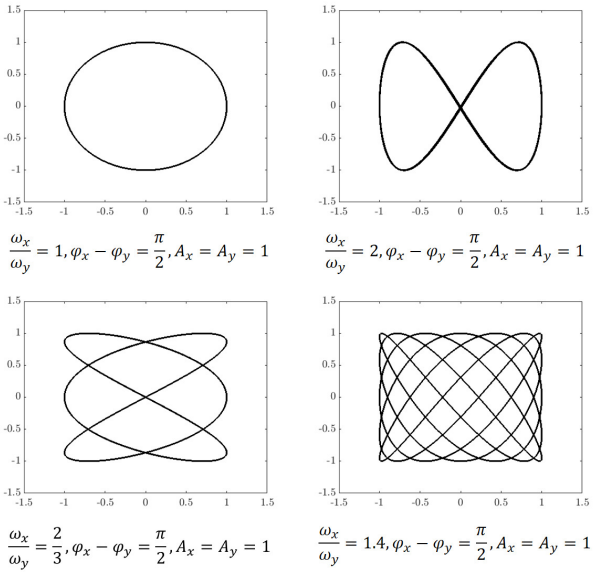

Fig. 2. Some Lissajous curves

III. COOPERATIVE LISSAJOUS DRONE SEARCH AND LOCALIZATION

Figure 3 summarizes the proposed framework as it is developed in this section. The original contributions detailed below formulate the Lissajous pattern as a linear state-space motion model, as well as establish the estimation formulas for Gaussian belief propagation when model parameters are uncertain.

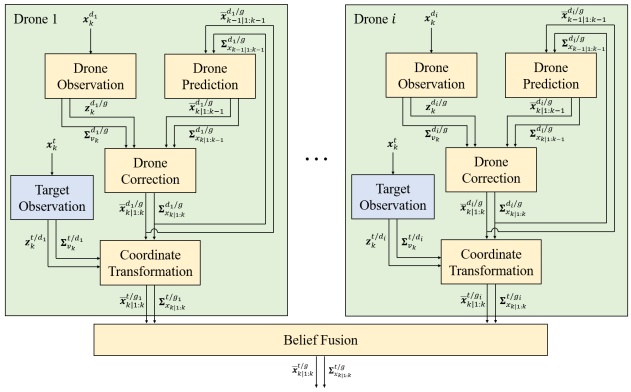

Fig. 3. Multi-agent Lissajous search-and-localization framework diagram. Yellow blocks are processed by the base station, while blue blocks are processed by individual agents.

A. The Lissajous Drone Search Pattern

In most controls-focused contexts, the mathematical model driving Kalman estimation captures input-output dynamics of a drone as a plant. This requires knowledge of control inputs $\mathbf{u}^d$ which govern the large-scale motion of the aerial agent. However, when this primitive level of control is determined offline by a flight planner, knowledge of control efforts applied to the drone may no longer be necessary depending on the form of the flight pattern. A major advantage of a Lissajous flight pattern is that the evolution of a drone's location projected into two-dimensional (2-D) physical space is deterministic at a macroscopic scale, eliminating the need for knowledge of, for example, voltage inputs to the aerial vehicle's motors. Since aerial search of ground targets can often be modeled well in two dimensions, Lissajous patterns are ideal flight path candidates. However, when the ratio of a search region's altitude range to its width is large, 2-D approaches may fail and a contour search may become necessary.

In order to formulate a joint drone-target RBE problem without knowing a drone's control inputs, it is useful to model its macroscopic motion with a discrete-time state-space equation. From observation of Eqs. (11), it is apparent that the x and y trajectories of an aerial vehicle following a Lissajous pattern are the solutions to second-order linear ordinary differential equations (ODEs). These continuous-time (CT) ODEs can be written as:

$$\ddot{x}(t) + \omega_x^2 x(t) = 0, \quad \ddot{y}(t) + \omega_y^2 y(t) = 0, \tag{12}$$

where the amplitudes and phase shifts are given in terms of the initial conditions as in the following:

$$A_x = \sqrt{(x(0))^2 + \left(\frac{\dot{x}(0)}{\omega_x}\right)^2}, \quad \varphi_x = \tan^{-1}\left(\frac{\omega_x x(0)}{\dot{x}(0)}\right). \tag{13}$$

A CT state-space form of these equations is given in two dimensions by:

$$\begin{bmatrix}\dot{x}(t)\\ \ddot{x}(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\omega_x^2 & 0\end{bmatrix}\begin{bmatrix}x(t)\\ \dot{x}(t)\end{bmatrix} \rightarrow \dot{\mathbf{x}}(t) = \mathbf{A}_x^{(C)}\mathbf{x}(t), \quad (14a)$$

$$\begin{bmatrix}\dot{y}(t)\\ \ddot{y}(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\omega_y^2 & 0\end{bmatrix}\begin{bmatrix}y(t)\\ \dot{y}(t)\end{bmatrix} \rightarrow \dot{\mathbf{y}}(t) = \mathbf{A}_y^{(C)}\mathbf{y}(t). \quad (14b)$$

The superscripts (C) on the state matrices denote that they compose continuous-time motion models. In order to implement an RBE algorithm, these models are discretized using the following well-known formula:

$$\mathbf{A}^{(D)} = \exp\left(\Delta t \mathbf{A}^{(C)}\right), \quad (15)$$

where Δt is the sampling period, (D) denotes the discrete-time (DT) state matrix, and $\exp(\bullet)$ signifies the matrix exponential. For small Δt, the matrix exponential can be approximated by the first three terms of its power series:

$$\mathbf{A}^{(D)} \approx \mathbf{I} + \Delta t \mathbf{A}^{(C)} + \frac{1}{2}\left(\Delta t \mathbf{A}^{(C)}\right)^2. \quad (16)$$

This approximation allows the DT state-space equations to be written in closed form as finite functions of the sampling period and frequencies:

$$\mathbf{A}^{(D)} \approx \begin{bmatrix}1 - \left(\frac{\Delta t\omega}{2}\right)^2 & \Delta t\\ -\Delta t\omega^2 & 1 - \left(\frac{\Delta t\omega}{2}\right)^2\end{bmatrix}. \quad (17)$$

Let the state of an i^{th} drone d_i following a Lissajous pattern be defined by the vector $\mathbf{x}_k^{d_i}$:

$$\mathbf{x}_k^{d_i} = \begin{bmatrix}x_k^{d_i} & \dot{x}_k^{d_i} & y_k^{d_i} & \dot{y}_k^{d_i}\end{bmatrix}^\top. \quad (18)$$

The drone's DT state matrix $\mathbf{A}^{d_i}$ is then obtained by assembling the x and y state matrices as follows:

$$\mathbf{A}^{d_i} = \begin{bmatrix}1 - \left(\frac{\Delta t\omega_x^{d_i}}{2}\right)^2 & \Delta t & 0 & 0\\ -\Delta t{\omega_x^{d_i}}^2 & 1 - \left(\frac{\Delta t\omega_x^{d_i}}{2}\right)^2 & 0 & 0\\ 0 & 0 & 1 - \left(\frac{\Delta t\omega_y^{d_i}}{2}\right)^2 & \Delta t\\ 0 & 0 & -\Delta t{\omega_y^{d_i}}^2 & 1 - \left(\frac{\Delta t\omega_y^{d_i}}{2}\right)^2\end{bmatrix}. \quad (19)$$

(Note that the (D) notation has been dropped for conciseness.) Finally, the linear 4-D state-space motion model of drone d_i is given by:

$$\mathbf{x}_k^{d_i} = \mathbf{A}^{d_i}\mathbf{x}_{k-1}^{d_i} + \mathbf{w}_{k-1}^{d_i}, \quad (20)$$

where $\mathbf{w}_{k-1}^{d_i}$ again accounts for additive system noise. In practice, the desired Lissajous path could be determined offline by a flight planner using waypoints, while an estimator in the field would simply model the path as above.

B. Estimation of Drones With Respect To Ground

In order to estimate the location of a target in the field with respect to a global reference frame (e.g. a static base station located at some global origin), it is first necessary to ensure accurate estimation of the search agents by the base station. For a drone following a Lissajous pattern, the state can be expected to evolve according to Eqs. (19)-(20). In general, the KF is a good candidate for estimating the drone's position. However, because the motion of the aerial vehicle is modeled at the macroscopic scale, it is likely that disturbances from wind, waypoint deviation, and other factors would cause the drone's motion to differ somewhat from what is expected. This deviation can be modeled by treating the model parameters ω_x and ω_y as random, rather than deterministic, variables. Because the shape of the Lissajous curve is highly sensitive to variation in these frequency parameters, a ground-truth state matrix that differs even slightly from that of the motion model can result in poor estimation. This requires an augmentation to the prediction stage of the KF.

Let the mean in prediction be given by the following expression:

$$\overline{\mathbf{x}}_{k|1:k-1}^{d_i/g} = \overline{\mathbf{A}}^{d_i}\overline{\mathbf{x}}_{k-1|1:k-1}^{d_i/g}, \quad (21)$$

where the superscript signifies estimation of drone d_i with respect to the ground g. Note how the mean estimated state matrix is now used here; the uncertainty in this estimate comes from uncertainty in ω_x and ω_y. State covariance propagation in prediction is formulated to account for this parametric uncertainty:

$$\mathbf{\Sigma}_{x_{k|1:k-1}}^{d_i/g} = \overline{\mathbf{A}}^{d_i}\mathbf{\Sigma}_{x_{k-1|1:k-1}}^{d_i/g}\overline{\mathbf{A}}^{d_i\top} + \mathbf{\Sigma}_{w_{k-1}}^{d_i/g} + \mathbf{J}_\omega^{d_i}\mathbf{\Sigma}_\omega^{d_i}\mathbf{J}_\omega^{d_i\top}. \quad (22)$$

The addition of the third term above augments predictive covariance propagation under the KF as in Eq. (4). In this expression,

$$\mathbf{\Sigma}_\omega^{d_i} = \begin{bmatrix}(\sigma_{\omega_x}^{d_i})^2 & \mathrm{cov}(\omega_x^{d_i}, \omega_y^{d_i})\\ \mathrm{cov}(\omega_y^{d_i}, \omega_x^{d_i}) & (\sigma_{\omega_y}^{d_i})^2\end{bmatrix} \quad (23)$$

and the Jacobian $\mathbf{J}_\omega^{d_i}$ is

$$\mathbf{J}_\omega^{d_i} = \begin{bmatrix}\frac{\partial\overline{x}_{k|1:k-1}^{d_i/g}}{\partial\omega_x} & \frac{\partial\overline{x}_{k|1:k-1}^{d_i/g}}{\partial\omega_y}\\ \frac{\partial\overline{\dot{x}}_{k|1:k-1}^{d_i/g}}{\partial\omega_x} & \frac{\partial\overline{\dot{x}}_{k|1:k-1}^{d_i/g}}{\partial\omega_y}\\ \frac{\partial\overline{y}_{k|1:k-1}^{d_i/g}}{\partial\omega_x} & \frac{\partial\overline{y}_{k|1:k-1}^{d_i/g}}{\partial\omega_y}\\ \frac{\partial\overline{\dot{y}}_{k|1:k-1}^{d_i/g}}{\partial\omega_x} & \frac{\partial\overline{\dot{y}}_{k|1:k-1}^{d_i/g}}{\partial\omega_y}\end{bmatrix} = \mathbf{T}\overline{\mathbf{V}}_{k-1|1:k-1}^{d_i/g}\overline{\mathbf{\Omega}}, \quad (24)$$

where

$$\mathbf{T} = \begin{bmatrix}\Delta t^2 & 0 & 0 & 0\\ -2\Delta t & -\Delta t^2 & 0 & 0\\ 0 & 0 & \Delta t^2 & 0\\ 0 & 0 & -2\Delta t & -\Delta t^2\end{bmatrix}, \quad (25a)$$

$$\overline{\mathbf{V}}_{k-1|1:k-1}^{d_i/g} = \begin{bmatrix}\overline{x}_{k-1|1:k-1}^{d_i/g} & 0\\ \overline{\dot{x}}_{k-1|1:k-1}^{d_i/g} & 0\\ 0 & \overline{y}_{k-1|1:k-1}^{d_i/g}\\ 0 & \overline{\dot{y}}_{k-1|1:k-1}^{d_i/g}\end{bmatrix}, \quad (25b)$$

$$\overline{\mathbf{\Omega}} = \begin{bmatrix}\overline{\omega}_x & 0\\ 0 & \overline{\omega}_y\end{bmatrix}. \quad (25c)$$

For the sake of simplicity, it is assumed that the states of

each drone are directly observable so that the linear model of Eq. (5) applies, with $\mathbf{C}^{d_i} = \mathbf{I}$. Under this assumption, correction becomes belief fusion, where

$$\mathbf{\Sigma}^{d_i/g}_{x_{k|1:k}} = \left[(\mathbf{\Sigma}^{d_i/g}_{x_{k|1:k-1}})^{-1} + (\mathbf{\Sigma}^{d_i/g}_{v_k})^{-1}\right]^{-1}, \quad (26)$$

and

$$\overline{\mathbf{x}}^{d_i/g}_{k|1:k} = \mathbf{\Sigma}^{d_i/g}_{x_{k|1:k}}\left[(\mathbf{\Sigma}^{d_i/g}_{x_{k|1:k-1}})^{-1}\overline{\mathbf{x}}^{d_i/g}_{k|1:k-1} + (\mathbf{\Sigma}^{d_i/g}_{v_k})^{-1}\mathbf{z}^{d_i/g}_{k}\right], \quad (27)$$

in accordance with Eqs. (10). In these expressions, $\mathbf{z}^{d_i/g}_k$ is an observation of drone d_i received by a sensor on the grounded base station, and $\mathbf{\Sigma}^{d_i/g}_{v_k}$ is the covariance matrix characterizing the uncertainty of this sensor.

Estimation of the target's location with respect to ground (see next section) requires a transformation from the drone's local reference frame to the global reference frame. As such, it is necessary to have an accurate estimate of the drone's global heading in addition to its position and velocity. While a drone's heading is not always tied to its trajectory, the general case where a drone always faces its direction of travel is addressed here. Figure 4 shows the movement of an aerial vehicle from step $k-1$ to k, from which the heading $\phi^{d_i/g}_k$ is derived. The mean in heading estimation can be approximated by the following expression:

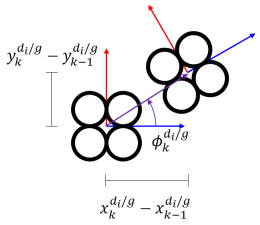

Fig. 4. Drone heading at step k

$$\overline{\phi}^{d_i/g}_{k|1:k} = \tan^{-1}\left(\frac{\overline{y}^{d_i/g}_{k|1:k} - \overline{y}^{d_i/g}_{k-1|1:k-1}}{\overline{x}^{d_i/g}_{k|1:k} - \overline{x}^{d_i/g}_{k-1|1:k-1}}\right) \approx \tan^{-1}\left(\frac{\overline{y}^{d_i/g}_{k|1:k}}{\overline{x}^{d_i/g}_{k|1:k}}\right). \quad (28)$$

The uncertainty in the heading, $\Sigma^{d_i/g}_{\phi_{k|1:k}}$ (or more conventionally denoted $(\sigma^{d_i/g}_{\phi_{k|1:k}})^2$ since this value is scalar), is a variance given in terms of the means and variances of the estimated velocities by the following:

$$\Sigma^{d_i/g}_{\phi_{k|1:k}} = \left(1 + \frac{\overline{y}^{d_i/g}_{k|1:k}}{\overline{x}^{d_i/g}_{k|1:k}}\right)^{-2}\Sigma^{d_i/g}_{\mathcal{V}}, \quad (29a)$$

$$\Sigma^{d_i/g}_{\mathcal{V}} = \left(\frac{\overline{y}^{d_i/g}_{k|1:k}}{\overline{x}^{d_i/g}_{k|1:k}}\right)^2\left(\left(\frac{\sigma^{d_i/g}_{\dot{y}_{k|1:k}}}{\overline{y}^{d_i/g}_{k|1:k}}\right)^2 + \left(\frac{\sigma^{d_i/g}_{\dot{x}_{k|1:k}}}{\overline{x}^{d_i/g}_{k|1:k}}\right)^2 - 2\frac{\mathrm{cov}(\dot{x}^{d_i/g}_{k|1:k}, \dot{y}^{d_i/g}_{k|1:k})}{\overline{y}^{d_i/g}_{k|1:k}\,\overline{x}^{d_i/g}_{k|1:k}}\right). \quad (29b)$$

C. Estimation of Target With Respect To Ground

Let the x-y location of a target t with respect to the ground g at step k be given by $\mathbf{x}^{t/g}_k \in \mathbb{R}^{2\times 1}$. Because the target dynamics and inputs are generally unknown, a motion model is unavailable and so prediction is unfeasible. Instead, belief is enhanced by leveraging the fusion of observations coming from multiple aerial search agents. An observation of the target by drone d_i is assumed to be linear with $\mathbf{C}^{t/d_i} = \mathbf{I}$. Therefore, the target's estimated state at step k with respect to this drone is characterized by the following mean and covariance:

$$\overline{\mathbf{x}}^{t/d_i}_k = \mathbf{z}^{t/d_i}_k, \quad \mathbf{\Sigma}^{t/d_i}_{x_k} = \mathbf{\Sigma}^{t/d_i}_{v_k}, \quad (30)$$

where $\mathbf{z}^{t/d_i}_k$ is the k^{th} observation of the target received by drone d_i and $\mathbf{\Sigma}^{t/d_i}_{v_k}$ is the covariance matrix characterizing the uncertainty of the drone's on-board sensor.

Obtaining an estimate of the target's location in the global reference frame requires a coordinate system transformation. Let $\mathbf{x}^{t/g_i}_k$ be the target's state at step k relative to the ground as perceived by drone i. The mean of this state estimate is:

$$\overline{\mathbf{x}}^{t/g_i}_{k|1:k} = \overline{\mathbf{x}}^{d_i/g}_{k|1:k} + \mathbf{T}^{d_i/g}_{k|1:k}\overline{\mathbf{x}}^{t/d_i}_k, \quad (31)$$

where

$$\mathbf{T}^{d_i/g}_{k|1:k} = \begin{bmatrix}\cos(\overline{\phi}^{d_i/g}_{k|1:k}) & -\sin(\overline{\phi}^{d_i/g}_{k|1:k}) \\ \sin(\overline{\phi}^{d_i/g}_{k|1:k}) & \cos(\overline{\phi}^{d_i/g}_{k|1:k})\end{bmatrix}, \quad (32)$$

and the velocity estimates are removed from $\overline{\mathbf{x}}^{d_i/g}_{k|1:k}$ such that it is 2-D. The covariance matrix $\mathbf{\Sigma}^{t/g_i}_{x_{k|1:k}}$ corresponding to the i^{th} drone's global estimate of the target is obtained from uncertainty propagation of Eq. (31) as given below:

$$\mathbf{\Sigma}^{t/g_i}_{x_{k|1:k}} = \mathbf{\Sigma}^{d_i/g}_{x_{k|1:k}} + \mathbf{T}^{d_i/g}_{k|1:k}\mathbf{\Sigma}^{t/d_i}_{x_k}{\mathbf{T}^{d_i/g}_{k|1:k}}^{\top} + \mathbf{J}^{d_i/g}_{\phi}\Sigma^{d_i/g}_{\phi_{k|1:k}}{\mathbf{J}^{d_i/g}_{\phi}}^{\top}, \quad (33)$$

where

$$\mathbf{J}^{d_i/g}_{\phi} = \begin{bmatrix}-\sin(\overline{\phi}^{d_i/g}_{k|1:k}) & -\cos(\overline{\phi}^{d_i/g}_{k|1:k}) \\ \cos(\overline{\phi}^{d_i/g}_{k|1:k}) & -\sin(\overline{\phi}^{d_i/g}_{k|1:k})\end{bmatrix}\begin{bmatrix}\overline{x}^{t/d_i}_k \\ \overline{y}^{t/d_i}_k\end{bmatrix}. \quad (34)$$

The final stage of the proposed framework is the fusion of all agents' global target estimates. This fusion is accomplished efficiently by leveraging Eqs. (10) as applied to a team of I search agents. The fused target state estimate is defined by the following covariance matrix and mean position vector:

$$\mathbf{\Sigma}^{t/g}_{x_{k|1:k}} = \left[\sum_{i=1}^{I}\left[\mathbf{\Sigma}^{t/g_i}_{x_{k|1:k}}\right]^{-1}\right]^{-1}, \quad (35a)$$

$$\overline{\mathbf{x}}^{t/g}_{k|1:k} = \mathbf{\Sigma}^{t/g}_{x_{k|1:k}}\left[\sum_{i=1}^{I}\left[\mathbf{\Sigma}^{t/g_i}_{x_{k|1:k}}\right]^{-1}\overline{\mathbf{x}}^{t/g_i}_{k|1:k}\right]. \quad (35b)$$

The uncertainty represented by the covariance matrix $\mathbf{\Sigma}^{t/g}_{x_{k|1:k}}$ is lower than those of all agents' individual uncertainties. Correspondingly, the target's mean position vector $\overline{\mathbf{x}}^{t/g}_{k|1:k}$ will on average be more accurate than any constituent estimate.

IV. RESULTS

This section validates the proposed framework for search and localization tasks separately.

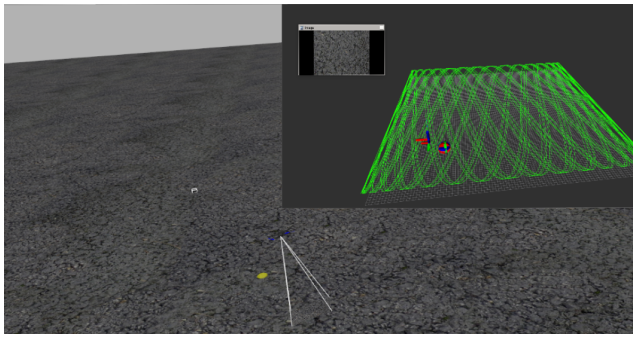

Fig. 5. Screen capture of Gazebo-based drone search simulation with Rviz rendering of commanded drone path, pose, and target pose overlaid.

A. Target Search

In order to compare the efficacy of the Lissajous search pattern with established search patterns, a Software-In-The-Loop (SITL) simulation environment was built. The Gazebo robotics simulation platform was used, with PX4 firmware running Pixhawk autopilot in a simulated drone controlled with Robot Operating System (ROS). The ROS wrapper used for mavlink communications protocol was MAVROS. In order to focus solely on the viability of the proposed pattern as opposed to other area coverage techniques, a standard quadrotor model, the 3DR Iris, and a minimalist simulated world were chosen for the experimental environment. A downward-looking simulated camera model was affixed to the simulated quadrotor. Camera parameters were set to model the Logitech c920 Webcam which was calibrated to real-world specifications. Target detection was based on fiducial tags using the Apriltags library. The open source "mavros_controllers" package developed by Jaeyong Lim [13] was chosen to implement a geometric controller that provides tracking control of a quadrotor UAV on SE(3) and utilizes differential flatness to model the quadrotor's dynamics based on Faessler's work [14]. Figure 5 shows a drone in this computer-generated environment.

Lissajous search patterns were compared with the sector search (SS), parallel sweep (PS), and expanding square (ES) patterns. The time-to-find-target, T_f, was recorded for each trial and each search pattern. The search region was a 100 meter $\times$ 100 meter grid with the starting location of the quadrotor fixed at the center. The chosen surrogate for a dynamic target was a turtlebot equipped with a top-mounted fiducial tag covering. The turtlebot's motion was given by a random walk with a fixed speed, while a custom rosnode was written to control the reference path for the Lissajous patterns. The target's starting location was randomized, selected from a cell within the search region, but outside of a 10 meter $\times$ 10 meter grid space centered on the origin.

The four Lissajous patterns of Fig. 1 were implemented, with L1 denoting the circular path and L2-L4 corresponding to the patterns of increasing complexity. For each search pattern, the altitude was fixed to 10 meters and 7 trials were performed. Each trial run had a hard limit of 1800 seconds to find the target; the number of trials for which the target was found before this limit was also reported. Table I summarizes the results.

TABLE I
NUMBER OF TRIALS TARGET WAS FOUND, MEAN AND STANDARD DEVIATION OF TIME TAKEN TO FIND TARGET

	SS	PS	ES	L1	L2	L3	L4
# Trials Found	1/7	**6/7**	5/7	0/7	1/7	5/7	**6/7**
Mean T_f	134	493	304.40	-	**87**	185	364
Std. T_f	-	224	174	-	-	138	343

As the table shows, the more complex Lissajous patterns locate the target more effectively than the lower-order ones since the latter do not cover the full search area. Under Lissajous patterns 3 and 4, the reduction in average detection time compared with the parallel sweep and the increased consistency as opposed to the sector and square searches is significant since incidental evasion by the randomly moving target is less likely under L3 and L4. Though a larger parametric study covering Lissajous patterns of varying orders is warranted, such a study is excluded here for the sake of brevity. We hypothesize that the order of the Lissajous search pattern can be optimized to achieve a balance between the consistency and time of detection, and that this optimization may change depending on even the most basic knowledge about a target (e.g. its speed limitations and randomness).

B. Target Localization

In order to validate the proposed Lissajous localization formulations, a simulation study was conducted in Matlab 2018a. For this study, a two-drone team was deployed to localize a moving target which remained in each drone's field of view (FOV) for the duration of each simulation. The target's path was chosen arbitrarily to be circular, while the drones followed Lissajous patterns with frequency parameters specified in Table II. The parameters δ and ϵ were introduced to efficiently vary these critical Lissajous frequencies and their variances. (Note: for these simulations, all parameters were unitless in order to preserve generality.)

TABLE II
SIMULATION PARAMETERS (FREQUENCIES)

$\overline{\omega}_x^{d_1}$	$\overline{\omega}_y^{d_1}$	$\overline{\omega}_x^{d_2}$	$\overline{\omega}_y^{d_2}$
$\pi/2$	$\delta\omega_x^{d_1}$	$\pi/2$	$\delta\omega_x^{d_2}$
$\Sigma_{\omega_x^{d_1}}$	$\Sigma_{\omega_y^{d_1}}$	$\Sigma_{\omega_x^{d_2}}$	$\Sigma_{\omega_y^{d_2}}$
$\epsilon\overline{\omega}_x^{d_1}$	$\epsilon\overline{\omega}_y^{d_1}$	$\epsilon\overline{\omega}_x^{d_2}$	$\epsilon\overline{\omega}_y^{d_2}$

Ground-truth frequencies were randomly generated from normal distributions characterized by the estimated parameters from the above table (e.g. $\omega_x^{d_1} \sim \mathcal{N}(\omega; \overline{\omega}_x^{d_1}, \Sigma_{\omega_x^{d_1}})$). Initial conditions were also randomly generated from normal distributions defined by the parameters of Table III. System

and sensor noise covariances are given in Table V of the appendix.

TABLE III
SIMULATION PARAMETERS (INITIAL CONDITIONS)

$\bar{\mathbf{x}}_0^{d_1}$	$\boldsymbol{\Sigma}_{x_0}^{d_1}$			
100	2	−0.9	−0.5	1
0	−0.9	3.2	0.4	−0.9
0	−0.5	0.4	2.3	0.8
100	1	−0.9	0.8	3.8
$\bar{\mathbf{x}}_0^{d_2}$	$\boldsymbol{\Sigma}_{x_0}^{d_2}$			
−100	2.4	−1.1	−0.4	0.8
0	−1.1	2.9	0.8	−0.7
0	−0.4	0.8	2.2	0.8
100	0.8	−0.7	0.8	3.4

For the simulated experiments performed, the sample time Δt was held at 0.05 and simulations ran for 400 time steps. Three estimation types were simulated: 1) sensor measurements only (no filtering), 2) basic Kalman filtering (no model parameter uncertainty propagation), and 3) the proposed formulations. Figures 6 and 7 demonstrate the performance of these three target localization approaches for a sample simulation.

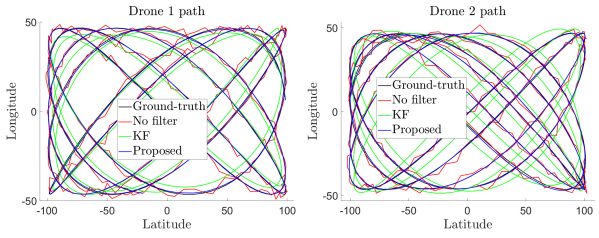

Fig. 6. Drone paths for which $\delta = 1.4$ and $\epsilon = 0.01$.

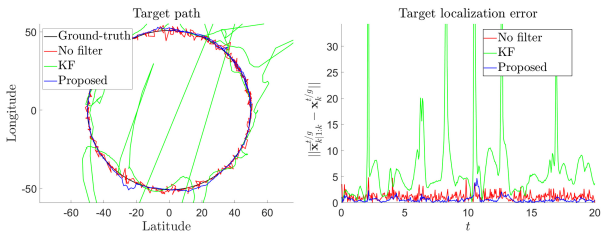

Fig. 7. Target estimation results for which $\delta = 1.4$ and $\epsilon = 0.01$.

As the figures show, even small drone estimation error translates to significant target estimation error because belief must be propagated through transformations which are especially sensitive to uncertainty in drone heading. By incorporating this uncertainty and fusing target estimates coming from each drone, the proposed localization formulas effectively smooth raw sensor observations to provide a more accurate estimate of target location as it evolves over time. It is worth noting that the particular target path shape does not play a role in the quality of estimation, as the formulations of section III do not presume a target motion model.

In order to investigate the robustness of target localization using the proposed formulations, a parametric Monte Carlo study was conducted. The sensitivity of estimation to pattern type and parameter uncertainty was studied by running 1,000 simulations for various combinations of δ and ϵ. The median errors and percent wins of each target localization approach over the 1,000 simulations are reported in Table IV.

TABLE IV
PARAMETRIC STUDY RESULTS (N = NO FILTERING, K = KALMAN FILTERING, P = PROPOSED FORMULATIONS)

		ϵ					
		0			0.001		
			% Wins	RMSE		% Wins	RMSE
δ	0.5	N	0.2	2.7892	N	5.4	1.2714
		K	**99.8**	**0.248**	K	0	5.3058
		P	**99.8**	**0.248**	P	**94.6**	**0.7088**
	0.75	N	0.3	1.8666	N	3.9	1.215
		K	**99.7**	**0.2275**	K	0.2	4.6408
		P	**99.7**	**0.2275**	P	**95.9**	**0.6231**
	1.4	N	0	1.1671	N	2.9	1.1624
		K	**100**	**0.2051**	K	0	4.0372
		P	**100**	**0.2051**	P	**97.1**	**0.5808**
	2.3	N	0	1.2182	N	0.4	1.1723
		K	**100**	**0.2043**	K	0	3.4974
		P	**100**	**0.2043**	P	**99.6**	**0.5303**
	3	N	0	1.1329	N	0.6	1.135
		K	**100**	**0.1956**	K	0	3.2499
		P	**100**	**0.1956**	P	**99.4**	**0.487**
		ϵ					
		0.01			0.1		
			% Wins	RMSE		% Wins	RMSE
δ	0.5	N	6.7	1.2869	N	**73.8**	**1.3145**
		K	0	17.2581	K	0	51.9602
		P	**93.3**	**0.8359**	P	26.2	1.7753
	0.75	N	6.8	1.2112	N	**79.8**	**1.2452**
		K	0	15.2444	K	0	44.598
		P	**93.2**	**0.7737**	P	20.2	1.8362
	1.4	N	7.3	1.1817	N	**81**	**1.196**
		K	0	14.3683	K	0	40.3666
		P	**92.7**	**0.7764**	P	19	1.7573
	2.3	N	6.7	1.1909	N	**80.7**	**1.2031**
		K	0	11.7647	K	0	36.1307
		P	**93.3**	**0.7685**	P	19.3	1.7633
	3	N	5.9	1.1698	N	**79.4**	**1.2014**
		K	0	11.3988	K	0	36.0664
		P	**94.1**	**0.7339**	P	20.6	1.7099

As the table shows, the performance of the proposed approach decreases as ϵ increases, while performance is only

negligibly dependent on δ. The former can be explained by the fact that higher model parameter uncertainty causes prediction to be weighted less in the correction stage of RBE, allowing sensor noise to dominate. The latter is evidence that the proposed target localization formulations are unspecific to the particular kind of Lissajous pattern implemented. It is worth noting that for $\epsilon = 0$, the KF and the proposed technique deliver identical results. This validates the original formulations of section III as an extension and improvement on well-established information-theoretic techniques.

V. CONCLUSIONS AND FUTURE WORK

The work presented in this paper formulates the Lissajous pattern for aerial search-and-localization in contexts where target motion is unknown and agent motion is uncertain. This framework eliminates the need for low-level dynamic modeling of search drones by formulating a search path which itself can be modeled by a linear state-space recurrence relation. The estimation formulations developed here reduce target localization error by both enhancing agent motion prediction and fusing target estimates coming from multiple agents. Monte Carlo simulation studies show that the Lissajous pattern is a valid area-coverage search pattern and that the proposed multi-agent estimation framework reliably improves target estimation in comparison with traditional filtering techniques.

A potential criticism of the proposed framework is that a drone will often need to cease its search pattern once the target is found, thereby changing its macro-motion model. While this is sometimes the case, there are many circumstances where the search agent should continue its pattern even after a target is found. For example, in scenarios where multiple targets are sought, a drone must continue its search until all targets are located. This requires simultaneously localizing past targets and searching for future targets. Developing such a multi-agent, multi-target framework remains as future work. Additionally, future iterations of this research will address scenarios where the target does not always remain in view during estimation. Finally, a version of this framework is being developed where drone motion model parameters are corrected during the SAL mission to further improve localization. SDG

TABLE V
SIMULATION PARAMETERS (SYSTEM AND SENSOR NOISE)

$$\mathbf{\Sigma}_w^{d_1/g} = 0.005 \times \begin{bmatrix} 0.16 & 0.03 & 0.07 & -0.11 \\ 0.03 & 0.21 & -0.09 & -0.12 \\ 0.07 & -0.09 & 0.19 & 0.03 \\ -0.11 & -0.12 & 0.03 & 0.32 \end{bmatrix}$$

$$\mathbf{\Sigma}_w^{d_2/g} = 0.005 \times \begin{bmatrix} 0.13 & 0.04 & 0.09 & -0.08 \\ 0.04 & 0.24 & -0.11 & -0.09 \\ 0.09 & -0.11 & 0.18 & 0.03 \\ -0.08 & -0.09 & 0.03 & 0.29 \end{bmatrix}$$

$$\mathbf{\Sigma}_v^{d_1/g} = 5 \times \begin{bmatrix} 0.2 & 0.1 & -0.06 & 0.13 \\ 0.1 & 0.4 & 0.04 & -0.11 \\ -0.06 & 0.04 & 0.6 & 0.19 \\ 0.13 & -0.11 & 0.19 & 0.9 \end{bmatrix}$$

$$\mathbf{\Sigma}_v^{d_2/g} = 5 \times \begin{bmatrix} 0.22 & 0.09 & -0.07 & 0.12 \\ 0.09 & 0.43 & 0.07 & -0.15 \\ -0.07 & 0.07 & 0.54 & 0.16 \\ 0.12 & -0.15 & 0.16 & 1.1 \end{bmatrix}$$

$$\mathbf{\Sigma}_v^{t/d_1} = 0.1 \times \begin{bmatrix} 0.28 & 0.12 \\ 0.12 & 0.31 \end{bmatrix}$$

$$\mathbf{\Sigma}_v^{t/d_2} = 0.1 \times \begin{bmatrix} 0.33 & -0.14 \\ -0.14 & 0.26 \end{bmatrix}$$

REFERENCES

[1] O. Arslan, D.E. Koditschek, "Voronoi-based coverage control of heterogeneous disk-shaped robots," In IEEE international conference on robotics and automation, pp. 4259–4266, 2016.

[2] S. Bernardini, M. Fox, D. Long, "Planning the Behaviour of Low-Cost Quadcopters for Surveillance Missions," International Conference on Automated Planning and Scheduling (ICAPS), Portsmouth, NH, USA, 2014.

[3] International Maritime Organization, *IAMSAR manual: international aeronautical and maritime search and rescue manual*, London, 2008.

[4] F. Bourgault, T. Furukawa and H.F. Durrant-Whyte, "Coordinated Decentralized Search for a Lost Target in a Bayesian World," Proceedings of the 2003 IEEE/RSJ Intenrational Conference on Intelligent Robots and Systems, October 2003, Las Vegas, NE, pp. 48-53, 2003.

[5] T. Furukawa, F. Bourgault, B. Lavis and H.F. Durrant-Whyte, "Recursive Bayesian search-and-localization Using Coordinated UAVs for Lost Targets," Proceedings of the 2006 International Conference on Robotics and Automation, May 2006, Orland, FL, pp. 2521-2526, 2006.

[6] G.M. Hoffman, S.L. Waslander and C.J. Tomlin, "Mutual Information Methods with Particle Filters for Mobile Sensor Network Control," Proceedings of the 45th IEEE Conference on Decision and Control, December 2006, San Diego, CA, pp. 1019-1024, 2006.

[7] G.M. Hoffman and C.J. Tomlin, "Mobile Sensor Network Control Using Mutual Information Methods and Particle Filters," IEEE Transactions on Automatic Control, vol. 55, no. 1, pp. 32-47, 2010.

[8] B. Grocholsky, A. Makarenko and H.F Durrant-Whyte, "Information-theoretic coordinated control of multiple sensor platforms," Proceedings of the 2003 IEEE Intenrational Conference on Robotics and Automation, September 2003, Taipei, Taiwan, pp. 1521-1526, 2003.

[9] A. Guanathillake, A.V. Savkin and A.P. Jayasumana, "Robust Kalman Filter-based Decentralised Target Search and Prediction with Topology Maps," IET Wireless Sensor Systems, Vol. 8, no. 2, pp. 60-67 2018.

[10] R. Mahler, "PHD Filters for Nonstandard Targets, I: Extended Targets," 12th International Conference on Information Fusion, Seatle, WA, pp. 915-921, 2009.

[11] P.M. Dames, "Distributed Multi-target Search and Tracking Using the PHD Filter," Autonomous Robots, vol. 44, pp. 673-689, 2020.

[12] J. J. Steckenrider, T. Furukawa, "Multi-Dimensional Belief Fusion of Multi-Gaussian Structures," Information Fusion, vol. 57, pp. 71-88, 2020.

[13] J. Lim, mavros_controllers - Aggressive trajectory tracking using mavros for PX4 enabled vehicles, Mar. 2019, https://doi.org/10.5281/zenodo.2652888.

[14] M. Faessler, A. Franchi, D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories." IEEE Robotics and Automation Letters, vol. 3, no. 2, pp. 620-626, 2018.