Bearing-Only Measurement Self-Localization, Velocity Consensus and Formation Control

Mengbin Ye, Brian D.O. Anderson, and Changbin Yu

***Abstract*—Self-localization and formation control tasks are considered when each agent in a multi-agent formation observes its neighbors but does not communicate. Each agent is restricted to a predefined motion type on a two-dimensional plane and collects bearing-only measurements over a time interval to localize neighboring agents. The localization process is used by a three agent formation to achieve velocity consensus combined with formation shape control. Simulations are provided and noisy bearing measurements are investigated.**

***Index Terms*—bearing measurement, velocity consensus, formation shape control, limited sensing, multi-agent localization**

I. Introduction

Autonomous systems have an increasingly large presence in the world and with a wide range of applications in areas such as military, industrial, and daily living. A multi-agent systems (MAS) is a group of agents acting in coordination. The term "agent" is used to describe a controllable subsystem such as an Unmanned Aerial Vehicle (UAVs), ground based vehicle or a mobile sensor in a sensor network [1].

Formation control is a sub-field of MAS and two typical formation control tasks are: forming and maintaining a specified formation shape and translating the formation (i.e. achieving velocity consensus among agents). A formation of UAVs can cover a large search area in a short amount of time [2]. By tracking the motion of a virtual leader, a formation can follow a planned path [3]. The need for a formation to take up a particular shape can arise when attempting to localize an object outside the formation (for example a geographical feature).

In this paper, we will consider distributed control of formations [4]. The majority of established formation control algorithms aimed at achieving, or stabilizing some desired formation require an agent to have knowledge of the relative position of its neighboring agents. A common assumption is that agents can directly measure both a bearing and distance to its neighbor, even if only distances are being controlled [5]. Alternatively, one can assume that each agent can measure its own position in a global coordinate frame (e.g. with GPS) and transmits this information to other neighboring agents [6]. Cooperative estimation of a single target by a group of agents is combined with formation control in [7]. By cooperative, we mean that inter-agent communications is required; future references to "cooperative" also carry this meaning.

M. Ye, B.D.O. Anderson and C. Yu are with the Research School of Engineering, Australian National University. B.D.O. Anderson and C. Yu are also with Hangzhou Dianzi University, Hangzhou, China. B.D.O. Anderson is also with Data61-CSIRO (formerly NICTA), Canberra, A.C.T., Australia.
{Mengbin.Ye, Brian.Anderson, Brad.Yu}@anu.edu.au

Many reasons exist which provide motivation for eliminating direct sensing of the distance variable and somehow inferring it using bearing-only measurements. It is also desirable that localization is achieved without communication or coordination, i.e. self-localization [8]. In particular, there is direct interest in application for military UAVs and the problem presented in this paper was initially posed by the Australian Defence Science and Technology Group (DSTG). For military operations, it is often undesirable to transmit radio signals (e.g. radar), which are detectable by the enemy. An optical camera attached to a fixed wing UAV can passively collect bearing measurements. Furthermore, there are payload or other forms of constraints for UAVs, especially for smaller aircraft such as the *Aerosonde* currently operated by the DSTG. Reducing the number of sensors on board which are employed just to maintain a formation can increase operational capabilities by freeing payload to apply to the main mission tasks. UAVs may typically self-localize by GPS but there may be intermittent jamming of GPS signals and restricted communication with neighbors. In GPS-denied situations agents only have a local coordinate frame, typically obtained from inertial navigation sensing. In such cases, it is not meaningful for an agent to communicate its own position or velocity to other agents.

There are a number of results on multi-agent localization and formation control in limited sensing environments. In the context of localization, a formation of UAVs cooperatively achieves localization in [9] using range-only measurements between agents in addition to bearing-only measurements to two known landmarks. Bearing-only measurements together with a single distance measurement is used for cooperative localization of a formation in [10]. Using bearing-only measurements and a set of three sensors, [11] studies a centralized method to localize a target. The work [12] studies cooperative self-localization of a formation, requiring at least two different landmarks with known locations. In [13], a group of UAVs using heterogeneous passive sensors cooperatively localize a stationary target emitter. A single agent successfully tracked a moving target in [14] using a priori knowledge of the target dynamics. Cooperative localization involving a GPS-equipped UAV and a GPS-denied UAV using bearing measurements is studied in [15]. Multiple mobile sensors taking asynchronous bearing measurements were used to localize a target in [16], [17]. The work [18] studied tracking of a moving target by a single sensor taking bearing-only measurements. A moving target is tracked using bearing measurements obtained from optical cameras by a team of cooperative UAVs in [19]. Bearing-only measurement based Simultaneous Localization and Mapping (SLAM) is studied in [20].

In the context of formation control, a strategy called "stop-and-go" is introduced in [21]. An algorithm in [22] used range-only measurements to achieve formation shape control and velocity consensus. The work [23] required both range-only and bearing-only measurements to two neighboring agents to maintain a rigid formation. Formation shape control using bearing-only measurements is studied in [24], [25]. However, the control laws can only achieve a formation shape *up to a scaling*. An additional distance measurement and associated controller is required to achieve the correct scale of the formation shape.

It is impossible for one agent, in its own local coordinate basis, to localize using bearing-only measurements another agent which is undergoing arbitrary motion in a two-dimensional plane. This paper will demonstrate that it is possible, in a two-dimensional plane, *to use bearing-only measurements to achieve localization and complete formation control tasks under certain constraints*. Firstly, it is clear that localization cannot be achieved using instantaneous bearing-only measurements; a form of observer must be built which uses measurements collected over an interval. Secondly, to enable localization, we assume that agents move with the same motion type that is known by all agents a priori.

Apart from a preliminary work [26] by the authors the result is believed to be novel. Below is a summary of the preliminary work and the additional contributions of this paper:

- In [26], localization was demonstrated for agents performing stationary circular orbits. In this paper, the agents perform a linear combination of a circular orbit and a linear translation (the point will become clearer when we discuss Fig. 2).
- In [26] a Fast Fourier Transform (FFT) is used to estimate the magnitude of the angular velocity of the agent being localized. The need to use the FFT is removed in this paper by providing that each agent with limited a priori knowledge of its neighboring agents.
- In [26] a two-dimensional grid search is used in the localization algorithm. By a change of variables, we reduced significantly the computational time required to achieve localization through an improved algorithm searching a one-dimensional grid.
- In [26], the use of the tangent function created issues when noisy bearing measurements are close to 90 or 270 degrees. The new algorithm avoids this.
- Velocity consensus, and velocity consensus combined with formation shape control are considered. Because we use bearing-only measurements to obtain inter-agent distances and velocities, unlike [24], [25], we are also able to control the scale of the formation shape.
- We provide an adaptive algorithm which controls the radius of circular orbit to ensure the agent motions are consistent with a minimum turning radius constraint.

The rest of the paper is structured as follows. Section II establishes the framework and defines the localization problem geometrically. Section III details the proposed localization algorithm. The tasks of velocity consensus and formation shape control are investigated in Section IV. Simulations for localization, velocity consensus and formation shape control are presented in Section V. The effects of noise in the bearing measurements are investigated in Section VI. Concluding arguments are given in Section VII.

II. The Localization Problem

The key to the proposed approach for formation shape control is the ability of an agent to localize its neighbors in its own coordinate basis using bearing-only measurements; there are already well developed consensus and formation control algorithms that can be exploited once localization is achieved. This section will therefore develop the framework for the localization task.

In an ambient two-dimensional space, each agent of a pair is assumed to perform a linear combination of simple circular and translational motion. To begin with, consider each motion separately. Each agent executes a simple circular orbit about its respective center. Then, in addition, the circle center of each agent has a translational velocity. Each agent has its own arbitrarily defined local Cartesian coordinate system, whose origin is located at the center of the circular orbit. Agent 1 continuously collects measurements of the bearing between an axis of its local coordinate system and Agent 2. From this information, Agent 1 must localize Agent 2, in Agent 1's coordinate basis. By doing this Agent 1 obtains the inter-agent distance and the distance between the two circle centers at any instance in time and formation control tasks can be executed. Similarly, Agent 2 will localize Agent 1 using bearing-only measurements. The key is to ensure that each agent's motion can be characterized by a finite set of parameters (which are constant over the interval being measured). The localization problem then concerns learning these parameters in order to then obtain the agent state (position and velocity); this latter task is an observer design problem.

While the circular motion has been shown to provide a consistent approach to achieving localization [26], an agent cannot remain stationary if formation control tasks are to be executed. The addition of a translational velocity rectifies this, allowing more practical trajectories. It is important to note that the translational velocity is that of the circle center; the absolute velocity of the agent itself changes continuously in a sinusoidal fashion. In order to simplify the control problem, we aim to achieve velocity consensus of the circle centers of each agent, rather than the agents themselves. Equally, formation shape control will aim to maintain the distances between circle centers, rather than the inter-agent distances.

A. Geometric Model

The localization problem is now described in terms of geometry, from the perspective of Agent 1 and can be seen in Fig. 1. The origin of a local coordinate basis is located at the center of orbit of Agent 1 with the $x, y-$axes pointing in some arbitrary direction (as seen from a global coordinate basis) that is determined by Agent 1. Counter-clockwise rotation is positive. For $t \in [kT, (k+1)T)$, for $k = 0, 1, 2, 3...$, Agent 1 measures the bearing angle, $\theta(t)$, relative to its $x-$axis. Let T be a given interval length that is constant and known to all

agents. The value of T is selected to be of sufficient length for there to be enough bearing measurements to form the system of equations described in Section III; this becomes clear in the sequel. The linear inter-agent distance, $z(t)$, and the linear distance between circle centers, $d(t)$, are both time varying and not measurable. The position of Agent 2's circle center relative to Agent 1's coordinate basis at $t = kT$ is denoted by (x_2, y_2). For $i = 1, 2$:

- r_i is the fixed radius of orbit of Agent i about center i.
- ω_i is the constant angular velocity of Agent i.
- ϕ_i is the phase of Agent i relative to Agent 1's x-axis at $t = k$.

From the perspective of Agent 1, the circle center of Agent 2 will have a relative velocity $\boldsymbol{v}_{rel} = [v_{xr}, v_{yr}]^\top$, where v_{xr} and v_{yr} are the relative velocities in the x and y directions of Agent 1's coordinate basis. Agent 1 knows ω_1, r_1, ϕ_1 and must find $\omega_2, r_2, \phi_2, x_2, y_2, v_{xr}, v_{yr}$ to localize Agent 2. We assume that v_{xr} and v_{yr} are constant for $kT < t < (k+1)T$, $\forall k = 0, 1, 2, 3...$ and at the end of each period T, each agent updates its estimate of v_{xr} and v_{yr} and makes changes to the translational velocity of its own circle center, normally with the aim of moving towards a formation control task. This is summarised by Algorithm 1, which gives an overview of the localization and formation control process for Agent 1. Other agents in the formation, e.g. Agent 2, will follow an identical control algorithm within their own coordinate frames.

Algorithm 1 Localization and Control Process for Agent 1, $t \in [kT, (k+1)T)$, for any given k.

1) Agent 1 measures $\theta(t)$ for $t \in [kT, (k+1)T)$
2) Agent 1 uses the localization algorithm in Section III to obtain the unknowns $r_2, \phi_2, \omega_2, v_{yr}, v_{xr}, x_2, y_2$, thus localising Agent 2.
3) At $t = (k+1)T$, Agent 1 updates its own flight parameters according to a control law to achieve a desired objective
   a) Achieving velocity consensus (Section IV-A).
   b) Achieving simultaneous velocity consensus and formation shape control (Section IV-B).
   c) Agent 1 adjusts r_1 to ensure that its flight path exceeds a minimum turn radius (Section IV-C).

For this work, it is assumed the agents are synchronized in time, such that the time periods $kT, \forall k = 0, 1, 2, 3...$ are the same for all agents. The agents simultaneously make changes to their translational velocities at the end of each period, in accordance with Algorithm 1.

From the geometric model in Fig. 1 we can obtain:

$$\frac{\sin(\theta(t))}{\cos(\theta(t))} = \frac{y_2 + v_{yr}t + r_2\sin(\omega_2 t + \phi_2) - r_1\sin(\omega_1 t + \phi_1)}{x_2 + v_{xr}t + r_2\cos(\omega_2 t + \phi_2) - r_1\cos(\omega_1 t + \phi_1)} \tag{1}$$

and note $z(t)$ and $d(t)$ are absent. Equation (1) is rearranged so that all the quantities appearing on the left-hand side are known for all time, while all the unknown parameters associated with Agent 2 are on the right-hand side. The variable

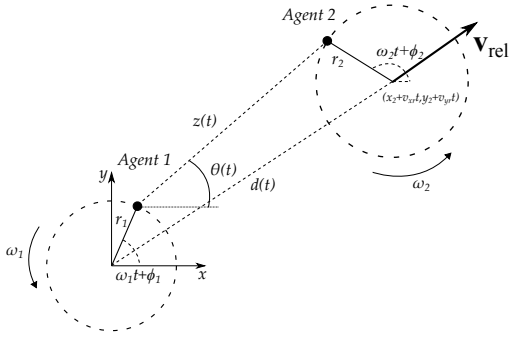


Figure 1. Geometric model

ϕ_2 is replaced by the substitute variables, $p = r_2\sin(\phi_2)$ and $q = r_2\cos(\phi_2)$ to give:

$$\begin{aligned} r_1[\,&\cos(\theta(t))\sin(\omega_1 t + \phi_1) - \sin(\theta(t))\cos(\omega_1 t + \phi_1)] \\ &= \cos(\theta(t))y_2 - \sin(\theta(t))x_2 \\ &\quad + \cos(\theta(t))v_{yr}t - \sin(\theta(t))v_{xr}t \\ &\quad + p\big[\cos(\theta(t))\cos(\omega_2 t) + \sin(\theta(t))\sin(\omega_2 t)\big] \\ &\quad + q\big[\cos(\theta(t))\sin(\omega_2 t) - \sin(\theta(t))\cos(\omega_2 t)\big] \end{aligned} \tag{2}$$

Equation (2) becomes the key equation in obtaining the localization solution. In the preliminary work [26] we proposed a grid search over the two unknowns ϕ_2 and ω_2 which made the problem linear in the remaining unknowns x_2, y_2, r_2 (recall that $v_{xr}, v_{yr} = 0$ in [26]). In the following section, we show that a reparametrization using the constants p, q, as in expression (2), enables a more efficient one dimensional grid search over ω_2 while ensuring that (2) is linear in the remaining unknowns $p, q, x_2, y_2, v_{xr}, v_{yr}$. Furthermore, [26] uses the quantity $\tan(\theta(t))$ in the solution algorithm which, in the presence of noisy θ measurements, had the undesired effect of significantly magnifying noise when θ was close to $\pm 90°$. The expression in (2) avoids this problem.

For future simulation comparison and providing insight and comments, we now introduce a global Cartesian coordinate system. Because each agent operates within its own local coordinate system, the orientation of the global coordinates with respect to the local coordinate systems does not affect analysis. For simplicity, this work therefore assumes that each local coordinate system is aligned with the global coordinate system. The equation of motion for agent i in the global coordinate system, for $t \in [kT, (k+1)T)$ for any k, is

$$X_i(t) = X_{0,i} + V_{x,i}t + r_i\cos(\omega_i t + \phi_i) \tag{3a}$$

$$Y_i(t) = Y_{0,i} + V_{y,i}t + r_i\sin(\omega_i t + \phi_i) \tag{3b}$$

where $X_i(t), Y_i(t)$ denotes the global position of agent i at time t and $X_{0,i}, Y_{0,i}$ denotes the position of the agent's center of circular orbit at $t = kT$. The translational velocity of the agent's center of circular orbit is $V_{x,i}, V_{y,i}$ and r_i, ω_i, ϕ_i were defined at the start of Section II-A.

B. Constraints on ω_2

It will become apparent in Section III-A that determining ω_2 is key to the localization solution. It is also the most

problematic of the unknowns to obtain. We therefore postulate that some constraints on ω_2 are known a priori, a reasonable assumption for cooperating agents. Suppose that before Agent 1 begins its mission, it is supplied with values $l_2, u_2 \in \mathbb{R}$ such that $\omega_2 \in [l_2, u_2]$ (and Agent 2 is given $l_1, u_1 \in \mathbb{R}$ such that $\omega_1 \in [l_1, u_1]$). As further explored in Section III-C, the localization problem cannot be solved if $\omega_1 = \omega_2$. Therefore, $[l_1, u_1]$ and $[l_2, u_2]$ cannot overlap.

III. Localization Algorithm

A. An Overdetermined System of Linear Equations

Equation (2) is highly nonlinear with trigonometric functions. However, the equation becomes linear in the six constants $y_2, x_2, v_{yr}, v_{xr}, p, q$ if the value of ω_2 is known. A one-dimensional search grid with a finite but closely spaced set of ω_2 values is created based on a priori knowledge about ω_2. By solving an overdetermined system of linear equations for each value in the search grid, we avoid having to solve a nonlinear equation. Let the search grid be $\widehat{\boldsymbol{\Omega}}_2$ with a constant step size $\delta\omega$, and the i^{th} value of the search grid is $\widehat{\omega}_{2,i}$. Since Agent 1 has prior knowledge of the interval that ω_2 can exist in, $\widehat{\boldsymbol{\Omega}}_2 = [l_2, l_2 + \delta\omega, l_2 + 2\delta\omega, ..., u_2]$. A value of $\delta\omega \leq 1e^{-3}$ was found to solve to reasonable accuracy for simulations involving parameters likely to be encountered while operating UAVs. Firstly, for ease of notation, let $a(t) = r_1[b(t)\sin(\omega_1 t + \phi_1) + c(t)\cos(\omega_1 t + \phi_1)]$, $b(t) = \cos(\theta(t))$, $c(t) = -\sin(\theta(t))$, $d_i(t) = b(t)\cos(\widehat{\omega}_{2,i}t) - c(t)\sin(\widehat{\omega}_{2,i}t)$, and $e_i(t) = b(t)\sin(\widehat{\omega}_{2,i}t) + c(t)\cos(\widehat{\omega}_{2,i}t)$.

For a given interval T with constant sampling frequency f_s, there are a total of f_sT bearing measurements. There are bearing measurements, $\theta(t)$, at points in time $t_1, t_2, t_3, ..., t_{f_sT}$. An overdetermined system of n linear equations is formed, where $6 < n < f_sT$ and $n \in \mathbb{Z}$, using n values of $\theta(t)$ sampled at points in time, $t_1, t_2, ..., t_n$. This is expressed as

$$\boldsymbol{u} = \boldsymbol{A}_i \boldsymbol{x}_i \tag{4}$$

where $\boldsymbol{u} = [a(t_1), a(t_2), \ldots, a(t_n)]^\top$, $\boldsymbol{x}_i = [y_2, x_2, v_{yr}, v_{xr}, p, q]^\top$ and the j^{th} row of $\boldsymbol{A}_i$ is given as $\boldsymbol{A}_i(j) = \begin{bmatrix} b(t_j) & c(t_j) & t_j b(t_j) & t_j c(t_j) & d_i(t_j) & e_i(t_j) \end{bmatrix}$. Note that $\boldsymbol{u}$ and the first four columns of $\boldsymbol{A}_i$ are independent of $\widehat{\omega}_{2,i}$ while the fifth and sixth columns of $\boldsymbol{A}_i$ are dependent on $\widehat{\omega}_{2,i}$. In the noiseless case, unless for some i there holds $\omega_2 = \widehat{\omega}_{2,i}$, then $\boldsymbol{u} = \boldsymbol{A}_i\boldsymbol{x}_i$ cannot be solved by any $\boldsymbol{x}_i$. In the presence of noise, (4) does not admit a solution even when $\omega_2 = \widehat{\omega}_{2,i}$. An approximate (least squares) solution can be found by looking for $\boldsymbol{x}_i^*$ with the property $\boldsymbol{x}_i^* = \arg\min \|\boldsymbol{u} - \boldsymbol{A}_i\boldsymbol{x}_i\|^2$ where $\|\cdot\|$ is the Euclidean norm. As is known $\boldsymbol{x}_i^* = (\boldsymbol{A}_i^\top \boldsymbol{A}_i)^{-1}\boldsymbol{A}_i^\top \boldsymbol{u}$. For each $\widehat{\omega}_{2,i}$, the overdetermined system is solved for $\boldsymbol{x}_i^*$ and the value of $\|\boldsymbol{u} - \boldsymbol{A}_i\boldsymbol{x}_i\|$ is calculated. The localization solution is the $\boldsymbol{x}_i^*$ with the minimum $\|\boldsymbol{u} - \boldsymbol{A}_i\boldsymbol{x}_i\|$ value. We denote this minimizer as $\boldsymbol{x}^*$. The corresponding $\widehat{\omega}_{2,i}^*$ is taken to be the angular velocity of Agent 2, taking note of the direction. The constants $y_2, x_2, v_{yr}, v_{xr}, p, q$ are found in $\boldsymbol{x^*}$ and r_2, ϕ_2 come trivially from p, q. If $\exists\ \widehat{\omega}_{2,i} \in \widehat{\boldsymbol{\Omega}}_2 \mid \widehat{\omega}_{2,i} = \omega_2$ then the system of equations is solved exactly for noiseless $\theta(t)$ and $\|\boldsymbol{u} - \boldsymbol{A}_i\boldsymbol{x}_i\|^2 = 0$ for $\boldsymbol{x}^*$. The values of $y_2, x_2, v_{yr}, v_{xr}, r_2, \phi_2$ are exact. In practical scenarios, it is more likely that ω_2 will lie between two grid frequencies and $0 < \|\boldsymbol{u} - \boldsymbol{A}_i\boldsymbol{x}_i\|$ with a consequential error in localization. The effect of $\delta\omega$ and n on localization accuracy is studied in Section V-B (noiseless $\theta(t)$) and Section VI (noisy $\theta(t)$).

B. When the interval $[l_2, u_2]$ is unavailable

In the rare instance that Agent 1 does not know the interval $[l_2, u_2]$, Agent 1 can use an FFT on the measured $\theta(t)$. The peaks of the resulting FFT are associated with ω_1 and ω_2.

Theorem 1 ([26]): The function $\theta(t)$, with $v_{xr} = v_{yr} = 0$, can be expressed as a linear combination of a constant and an infinite series of sinusoids whose angular frequencies are integer-weighted linear combinations of ω_1 and ω_2

In [26], the agents are stationary (i.e. $v_{xr} = v_{yr} = 0$). We present the following conjecture, although the result is not necessary for the logical completion of this paper.

Conjecture 1: The function $\theta(t)$, with $v_{xr}, v_{yr} \neq 0$, can be expressed as a linear combination of a constant and an infinite series of sinusoids whose angular frequencies are integer-weighted linear combinations of ω_1 and ω_2.

Observe that the numerator of the right hand side of (1) can be expressed as the sum of a time-varying term $y_2 + v_{yr}t$, and two sinusoids with angular frequencies of ω_1 and ω_2. It is straightforward to conclude that the term $y_2 + v_{yr}t$ provides zero contribution to the nonzero frequency content of the numerator. An identical conclusion can be drawn regarding $x_2 + v_{xr}t$ for the denominator. In other words, the additional translational velocity of the agent orbits affects the zero frequency component of the numerator and denominator of (1) but does not introduce new nonzero frequency components. We therefore conjecture that $\theta(t)$, which is a nonlinear function of the right hand side of (1), will only have frequencies which are integer-weighted linear combinations of ω_1 and ω_2.

From the FFT, Agent 1 is able to obtain $|\widetilde{\omega}_2|$, which is Agent 1's estimate of the magnitude of ω_2. Detailed discussions on the FFT and the creation of $\widetilde{\Omega}_2$ from the $|\widetilde{\omega}_2|$ is provided in [26] and we thus omit it from this paper. However, do note that $\widetilde{\Omega}_2$ should encompass two intervals, $[-\widetilde{\omega}_2 - \Delta, -\widetilde{\omega}_2 + \Delta]$, $[\widetilde{\omega}_2 - \Delta, \widetilde{\omega}_2 + \Delta]$ because only $|\widetilde{\omega}_2|$ is known.

C. Special Scenarios

The situation where $\omega_1 = \omega_2$ is a sufficient condition for the algorithm to fail as $\boldsymbol{A}$ in (4) is no longer full rank. The proof is provided in Appendix A. It follows that in a noisy environment, one should expect that $\omega_1 \approx \omega_2$ would cause problems so this should be avoided. The value of $\|\boldsymbol{u} - \boldsymbol{A}\boldsymbol{x}\|$ is also minimized at $\widehat{\omega}_{2,i} = \omega_1$. These two observations provide the motivation to ensure that $[l_1, u_1]$ and $[l_2, u_2]$ do not overlap.

Although intuition suggests that $\omega_2 = n\omega_1,\ n \in \mathbb{Z}_+, n > 1$ could be an issue, simulations of several different scenarios showed that a unique localization solution is obtained. Because of this, we postulate that the scenario where $\omega_2 = n\omega_1,\ n \in \mathbb{Z}_+, n > 1$ is *not* a special scenario; it is generically solvable.

The problem is also solvable in the situation where $\omega_2 = -\omega_1$, provided that the direction of rotation is known a priori. For example, $\omega_1 \in [-0.5, -0.2]$ and $\omega_2 \in [0.2, 0.5]$ is a valid

constraint; the possibility that $\omega_2 = -\omega_1$ exists but both agents know the rotational direction of the other.

D. Aliasing and Sampling Frequency

The passive bearing sensor on a UAV may be an optical camera attached on a rotating platform so that it may track another UAV over a full 360°. This is particularly likely if both UAVs are undergoing a circular orbit. Internal communication with DSTG indicated that a typical optical camera attached to a UAV has a sampling frequency, f_s, between 1 to 10 Hz. An alternative is to use catadioptric cameras, which provide panoramic views (and in some cases full 360° view). These have been previously used for formation control [27].

The lowest aliasing frequencies are $f_a = |(m\omega_1 + n\omega_2)/2\pi - f_s|$. If $2\pi f_a \in \widetilde{\mathbf{\Omega}}_2$ then a minimum may exist at $2\pi f_a$, giving an invalid localization solution. UAVs (and in indeed most mechanical agents of interest such as marine surface vessels or ground vehicles) will typically have $\omega_i \leq 1$ rads^{-1}. A sensor with a sampling frequency greater than 2 Hz is therefore sufficient in order to avoid aliasing.

IV. Formation Shape Control

For the formation shape control problem, the definitions of variables given in Section II-A still hold. A multi-agent system can be represented by an undirected graph, $G = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, with M agents, where each agent is represented as a node in the set $\mathcal{V} = \{v_1, \ldots, v_M\}$, with a corresponding set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. An edge of G is $e_{ij} = (v_i, v_j)$. The adjacency matrix $\mathcal{A}$ of G has elements $a_{ij} \in 0, 1$. The elements are defined such that $a_{ij} = a_{ji} = 1$ if $e_{ij} \in \mathcal{E}$ while $a_{ij} = 0$ if $e_{ij} \notin \mathcal{E}$ and $a_{ii} = 0, \forall i$. The neighbors of v_i are $\mathcal{N}_i = \{v_j \in \mathcal{V} : (v_i, v_j) \in \mathcal{E}\}$.

A. Velocity Consensus

In velocity consensus, the objective is to ensure that the centers of the agent's circular orbit converge to the same velocity. That is, $||\boldsymbol{v}_{rel}|| \to 0$ as $t \to \infty$ for all agents. Recall from Section II that we have assumed that the linear velocity component for each agent is constant for $kT < t < (k+1)T$, $\forall k = 0, 1, 2, 3...$ In addition, each agent updates its estimate of v_{xr}, v_{yr}, and makes changes to its own velocity at the end of each period T. The consensus algorithm selected is the discrete time consensus algorithm given by [28]:

$$\boldsymbol{v}_i\left((k+1)T\right) = \boldsymbol{v}_i(kT) + \epsilon \sum_{j \in \mathcal{N}_i} \left(\boldsymbol{v}_j(kT) - \boldsymbol{v}_i(kT)\right) \quad (5)$$

where $\boldsymbol{v_i} = [V_{x,i}, V_{y,i}]^\top$ is the translational velocity of Agent i's circle center in the global coordinate frame, as defined below (3). Equally, $\boldsymbol{v_j} = [V_{x,j}, V_{y,j}]^\top$ is the translational velocity of Agent j's circle center in the global coordinate frame. Although the above equation is given in a global coordinate frame, a straightforward calculation shows the algorithm is independent of a global coordinate system and each agent can execute (5) in its own local coordinate frame. Work in [29] showed for distance-based formation shape control on digraphs, algorithms running in an agent's local coordinate frame require only the formation to be infinitesimally rigid. This is readily extended to (5) and the algorithms in the next section. In agent i's local coordinate frame, $\boldsymbol{v}_j(kT) - \boldsymbol{v}_i(kT) = \boldsymbol{v}_{rel} = [v_{xr}, v_{yr}]^\top$. The step size, ϵ, is in the interval $(0, 1/\Delta)$, where $\Delta = \max_i \sum_{i \neq j} a_{ij}$ is the maximum degree of the network.

B. Formation Shape Control and Velocity Consensus Combined

Almost all algorithms which combine formation shape control with velocity consensus have been formulated in continuous time. One such algorithm in [30] is given as:

$$\dot{\boldsymbol{p}}_i = \boldsymbol{v}_i \quad (6a)$$

$$\dot{\boldsymbol{v}}_i = \sum_{j \in \mathcal{N}_i} (\boldsymbol{v}_j - \boldsymbol{v}_i) + 2 \sum_{j \in \mathcal{N}_i} ({d_{ij}^*}^2 - {d_{ij}}^2)(\boldsymbol{p}_i - \boldsymbol{p}_j) \quad (6b)$$

where $\boldsymbol{p}_i, \boldsymbol{p}_j$ is, respectively, the position of the circle center of Agent i, j in the global coordinate frame. From Agent i's perspective, $\boldsymbol{p}_i - \boldsymbol{p}_j = -[x_0 + v_{xr}t, y_0 + v_{yr}t]^\top$. For $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$, the definition from the previous section holds. The desired and current distance between the circle centers of Agents i and j is d_{ij}^* and d_{ij} respectively. Consider a formation where the consensus graph satisfies conditions for consensus and the formation shape graph is an undirected and rigid graph and the desired edge lengths correspond to an infinitesimally rigid formation. Then algorithm (6) causes all solutions of this system to converge to a point $(\boldsymbol{p}_c, \boldsymbol{v}_c)$, where $\boldsymbol{p}_c$ is such that $d_{ij}^* = d_{ij}, \forall i \in \mathcal{N}_j$ and $\boldsymbol{v}_c = \boldsymbol{0}$. For a three agent formation both velocity consensus and the desired formation shape are achieved globally. For four agent and larger formations, the initial formation must start in a local neighborhood of the desired formation due to multiple equilibria; the algorithm is exponentially stable about the equilibrium point [29]. Note that the algorithm in equation (6) cannot be directly implemented as the localization process occurs after the bearing is measured over a time interval T.

The stability of a discretized version of (6) is analyzed in [22] and takes the form:

$$\dot{\boldsymbol{p}}_i = \boldsymbol{v}_i \quad (7a)$$

$$\boldsymbol{v}_i\left((k+1)T\right) = \boldsymbol{v}_i(kT) + \epsilon_1 \sum_{j \in \mathcal{N}_i} \left(\boldsymbol{v}_j(kT) - \boldsymbol{v}_i(kT)\right) + 2\epsilon_2 \sum_{j \in \mathcal{N}_i} ({d_{ij}^*}^2 - {d_{ij}}^2)(\boldsymbol{p}_i(kT) - \boldsymbol{p}_j(kT)) \quad (7b)$$

where ϵ_i are positive constants. Note that, since $\boldsymbol{v}_i$, $\boldsymbol{v}_j$ are piecewise constant, the discretization of (6a) is equivalent to the continuous function $\dot{\boldsymbol{p}}_i = \boldsymbol{v}_i$. Further note that $\boldsymbol{p}_i(kT)$, $\boldsymbol{p}_j(kT)$ refer to the position at the end of the interval kT, rather than the start.

Remark 1: It is apparent that the algorithms in (5) and (7) require agent i to change $V_{x,i}, V_{y,i}$ instantaneously at the end of each time interval, which is impossible in the physical world. A second time interval may be introduced at the end of each interval which allows each agent to smoothly change its velocity in accordance with its physical capabilities. During this second time period, each agent discontinues measurement

of $\theta(t)$. At the end of the second time interval, the next interval begins; agents have updated their velocity to $\boldsymbol{v}_i((k+1)T)$ and resume measuring $\theta(t)$.

C. Adaptive Selection of Radii and Angular Velocity

Although the primary focus is on velocity consensus of the circle centers, consideration must also be given to the actual trajectory of each agent. In certain scenarios, the trajectory of the agent becomes impossible to achieve with a fixed-wing UAV. Figure 2 shows one such trajectory, where the agent is required to fly with a turn radius that is too small. It is desirable to ensure that the turn radius, R, is above some threshold as defined by the agent's operating parameters.

The equations of motion for an agent can be obtained from (3), for a single time interval $t \in [kT, (k+1)T)$, for any k (hence we drop the subscript i). For simplicity of analysis, assume $\phi = 0$. The radius of curvature is $R = \left|(\dot{x}^2 + \dot{y}^2)^{3/2}/\dot{x}\ddot{y} - \dot{y}\ddot{x}\right|$ where $\dot{x}, \dot{y}$ are, respectively, the first derivatives of $x(t)$ and $y(t)$ with respect to time, and $\ddot{x}, \ddot{y}$ are, respectively, the second derivatives of $x(t)$ and $y(t)$ with respect to time [31]. We obtain that for the parametric set (3)

$$R(t) = \left| \frac{\left[{v_x}^2 + {v_y}^2 + r^2\omega^2 + 2r\omega f(t)\right]^{\frac{3}{2}}}{r\omega^2\left[r\omega + f(t)\right]} \right| \tag{8}$$

where $f(t) = v_y \cos(\omega t) - v_x \sin(\omega t)$ and notice that the amplitude of $f(t)$ is $\sqrt{{v_x}^2 + {v_y}^2}$. It follows that $R(t) \geq Q$ for all t, where

$$Q = \frac{\left({v_x}^2 + {v_y}^2 + r^2\omega^2 - 2r\omega\sqrt{{v_x}^2 + {v_y}^2}\right)^{\frac{3}{2}}}{r\omega^2\left(r\omega - \sqrt{{v_x}^2 + {v_y}^2}\right)} \tag{9}$$

Suppose that v_x, v_y, ω are fixed and so Q is a function only of r. Then there exists two real roots, q_1, q_2, for Q, given as:

$$q_1 = \frac{1}{2}\left(Q + \sqrt{Q^2 + \frac{4Q\sqrt{{v_x}^2 + {v_y}^2}}{|\omega|} + \frac{2\sqrt{{v_x}^2 + {v_y}^2}}{|\omega|}}\right) \tag{10a}$$

$$q_2 = \frac{1}{2}\left(Q - \sqrt{Q^2 + \frac{4Q\sqrt{{v_x}^2 + {v_y}^2}}{|\omega|} + \frac{2\sqrt{{v_x}^2 + {v_y}^2}}{|\omega|}}\right) \tag{10b}$$

It follows that if $r > q_1$ or $r < q_2$ then $R(t) > Q$ for all t. This allows R to be controlled by changing r such that R never falls below some operational threshold. The above assumption that v_x, v_y, ω are fixed is reasonable. Consider that v_x, v_y are updated according to some consensus or formation control algorithm, such equation (6); then, for each time interval, the two velocities are fixed for equation (9). It is also reasonable to assume that ω is constant. The corresponding agent trajectories are shown in Fig. 2. It is clear that $r < q_2$ consumes the least amount of energy while $r > q_1$ provides the large motion which is useful for localization (particularly when $\theta(t)$ is noisy). Equation (10a) can be used during the initial localization process while (10b) is used once velocity consensus and/or formation shape control has been achieved.

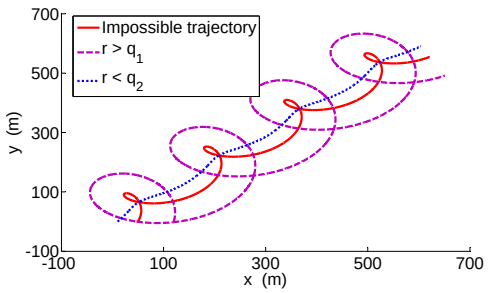


Figure 2. Trajectories for an agent with varying r.

Note that the term $|\omega|$ merely takes into account the direction of rotation. Furthermore, a 2-dimensional grid of ω and r can be used in conjunction with (9) to create a look-up table for pairs of ω, r which satisfy $R > Q$. This can give increased flexibility and the only constraint is to ensure that $\omega \in [l_i, u_i]$ as initially defined.

V. Simulations

In this section, results are presented for simulations of the proposed localization and formation control technique. Initially, simulations are presented for an ideal, noiseless environment. Simulations in the presence of noise are later presented in Section VI and strategies to mitigate the effect of noise are discussed. For velocity consensus and formation shape control, an undirected triangle formation is used. Furthermore, the values selected for simulation are intended to represented realistic UAV operating conditions and all values are in SI units. Distances are in meters, time in seconds and angle is measured in radians.

A. Localization

Recall the definitions of $X_{0,i}, Y_{0,i}$ and $V_{x,i}, V_{y,i}$ from Section II. Agent 1 is tasked with localizing Agent 2 in the following scenario, for the time interval $t \in [0, T)$, i.e. $k = 0$. Agent 1 has $X_{0,1} = 100$, $Y_{0,1} = -200$, $V_{x,1} = 5$, $V_{y,1} = 2$, $r_1 = 200$, $\omega_1 = 0.19$, $\phi_1 = \pi/6$. The parameters for Agent 2 are $X_{0,2} = 600$, $Y_{0,2} = 1000$, $V_{x,2} = 9$, $V_{y,2} = 3$, $r_2 = 80$, $\omega_2 = -0.2615$, $\phi_2 = -\pi/2$. Agent 1 knows that $\omega_2 \in [-0.6, -0.23]$ rads^{-1} and selected $\delta\omega = 0.001$ rads^{-1}. Set $n = 100$ equations and $f_s = 10$ Hz. The measured bearing angle in degrees is shown in Fig. 3, and only the first 10 seconds of $\theta(t)$ are used to create the overdetermined system of equations. Figure 4 shows the minimum of $\|\boldsymbol{u} - \boldsymbol{A}\boldsymbol{x}\|$ is reached when $\widehat{\omega}_{2,i} = -0.261$ rads^{-1}. The associated $\boldsymbol{x}^*$ gives $x_2 = 504$, $y_2 = 1198$, $r_2 = 80.3$, $v_{xr} = 3.9$, $v_{yr} = 0.95$, $\phi_2 = -1.58$. Recall from Section II-A that x_2, y_2, v_{xr}, v_{yr} are Agent 2's parameters relative to Agent 1's coordinate basis, e.g. $x_2 = X_{0,2} - X_{0,1}$, and so Agent 2 is successfully localized.

B. Accuracy and Performance of the Algorithm

The key factors which affect the accuracy and computational time of the proposed localization algorithm are $\delta\omega$ and n. The numerical nature of the method proposed makes

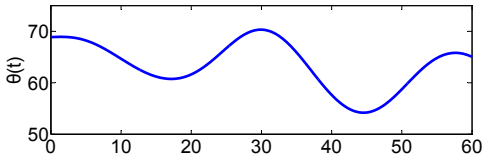


Figure 3. Plot of the measured bearing angle, $\theta(t)$ (in degrees) over time.

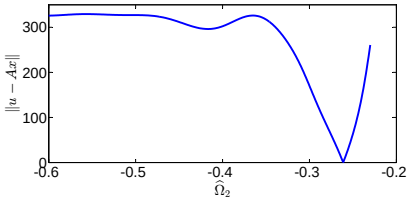


Figure 4. Plot of Least Squares Error, $\|\boldsymbol{u} - \boldsymbol{A}_i\boldsymbol{x}_i\|$ over the search grid, $\widehat{\boldsymbol{\Omega}}_2$

it harder to quantify the effects of these parameters on the estimation accuracy. In this subsection, we use Monte Carlo simulations to explore these effects. The simulations are of the same nature as outlined in Section V-A but the parameters are generated randomly. An error vector is defined $\boldsymbol{e} = [e_{y2}, e_{x2}, e_{r2}, e_{\omega_2}, e_{\phi_2}, e_{v_{xr}}, e_{v_{yr}}]^\top$ where each element is the absolute percentage error between the true value and value obtained by the localization algorithm. In Fig. 5, we hold $\delta\omega$ constant and vary n from $n = 10$ to $n = 510$. For each n, we randomly generated 50 scenarios and computed the average value of $\|e\|$ across the 50 scenarios. In Fig. 6, we hold n constant and vary $\delta\omega$ from $\delta\omega = 0.01$ to $\delta\omega = 0.0001$. Again, we randomly generated 50 scenarios and computed the average $\|e\|$ across the 50 scenarios. Note that at this stage, there has been no noise introduced; the errors if Fig. 5 and 6 are due to discretisation of the search grid $\widehat{\boldsymbol{\Omega}}_2$. These two sets of simulations show that the algorithm works well for multiple scenarios, as opposed to a single carefully selected scenario.

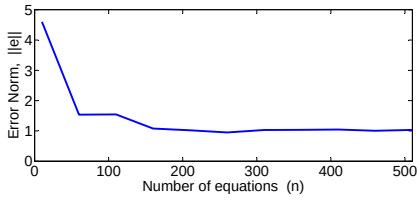


Figure 5. Localization error as the number of equations varies, with noiseless $\theta(t)$ and $\delta\omega = 0.001\,\text{rads}^{-1}$.

C. Velocity Consensus

An undirected triangle formation is used along with algorithm (5). The measuring period is $T = 60$ s, $f_s = 10$ Hz, $n = 100$ and $\delta\omega = 0.001$ rads^{-1}. A value of $\epsilon = 0.45$ was selected. The initial conditions from Section V-A for Agents 1 and 2 remain the same. Agent 3 is introduced: for $k = 0$ we have $X_{0,3} = 800$, $Y_{0,3} = -600$, $V_{x,3} = -8$, $V_{y,3} = -1$,

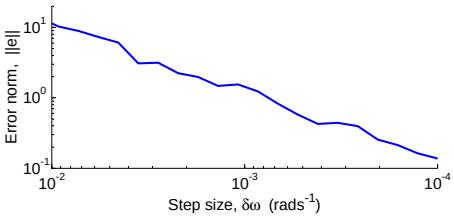


Figure 6. Localization error as the step size $\delta\omega$ varies, with noiseless $\theta(t)$ and $n = 300$.

$r_3 = 100$, $\omega_3 = -0.09$, $\phi_3 = \pi/4$. The information $\omega_1 \in [0.150.2]$, $\omega_2 \in [-0.6 - 0.2]$, $\omega_3 \in [-0.1 - 0.05]$ is supplied to the relevant agents. The radius of each agent is changed adaptively in the following manner. We solve for q_1 and q_2 in (10) by setting the minimum turn radius to $Q = 50$ m. For Agent i, we initially set $r_i > q_1$. When $\sum_{j\in\mathcal{N}_i} \|\boldsymbol{v}_j(kT) - \boldsymbol{v}_i(kT)\| < 0.05$ we update r_i at the end of the k^{th} period to $r_i < q_2$. This ensures that the agent motion is large while consensus has not been reached to aid in localization, and the agent changes to an energy saving motion once the relative velocities are reduced to approximately zero. The trajectories of the agents over the global X, Y coordinate frame are shown in Fig. 7 and the velocities of the circle centers of the agents are in Fig. 9.

D. Formation Shape Control and Velocity Consensus

Using the same undirected triangle formation, the same initial conditions and the same $f_s, n, \delta\omega$ as in Sections V-A and V-C, algorithm (7) is implemented. The measuring period is reduced to $T = 30$s. Suitable values were found as $\epsilon_1 = 0.3$ and $\epsilon_2 = 0.0025$. The desired formation is an equilateral triangle such that $d_{ij} = d_{ji} = 800,\ i, j = 1, 2, 3, i \neq j$. The same adaptive radii algorithm is used, with $Q = 50$ m and $r > q_1$ is maintained until the velocity consensus and formation shape control objectives are complete. The trajectories of the agents over the global X, Y coordinate frame are shown in Fig. 8 and the velocities of circle centers of the agents are presented in Fig. 10. Notice a sudden divergence in the circle center velocities at $t \approx 700$ s. This is when the radii of the agents change from $r > q_1$ to $r < q_2$. Since the circle center is a local, virtual point and the agent is a physical entity, a change in r results in the circle center shifting rather than the agent. Even while executing energy conserving motion, velocity consensus and the correct formation shape is regained.

VI. Noise

In practical implementation, there is some amount of noise when sensing or taking samples. The noise is introduced in $\theta(t)$, and is usually wide bandwidth noise. In the following simulations, noise with a bandwidth well above that of θ is added. In reality, the noise follows a von Mises distribution [32]. However in simulation, a zero mean Gaussian distribution is used. When the noise is small and concentrated about the true value of θ, the von Mises distribution is closely approximated by a Gaussian distribution. Internal communication with DSTG gave an indication of likely encountered levels of noise

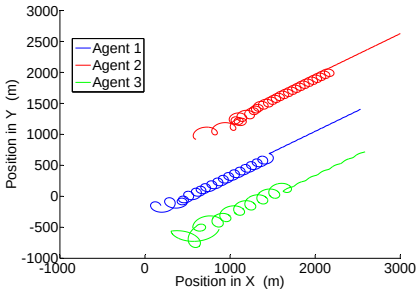


Figure 7. Velocity Consensus: The trajectories of the agents are depicted in the global coordinate basis.

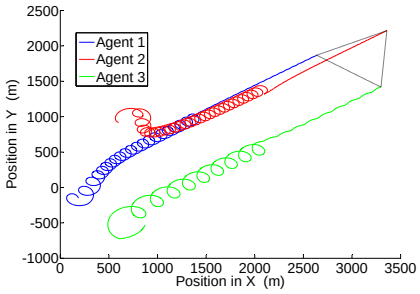


Figure 8. Formation Shape Control and Velocity Consensus: The trajectories of the agents are depicted on the global coordinate basis.

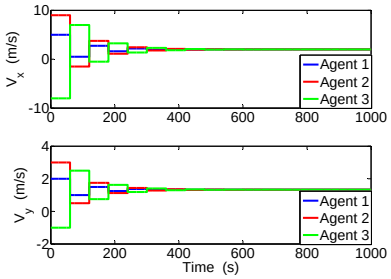


Figure 9. Velocity Consensus: Velocity of the agent circle centers in the global coordinate basis.

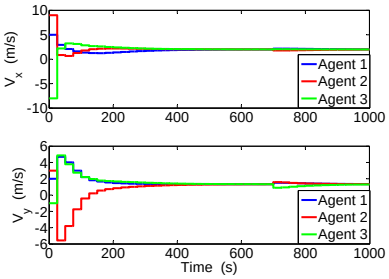


Figure 10. Formation Shape Control and Velocity Consensus: Velocity of the agent circle centers in the global coordinate basis.

with the optical cameras operating on their UAVs. Noise will be affected by the distance between the agent and the object being tracked; a larger distance means the object will occupy fewer pixels on the camera. However, to an approximation, the cameras have a standard deviation of 1 to 10 milliradians, or approximately 0.06 to 0.6° . We indicate noise level by $k = 3\sigma$ so that with a standard deviation of σ degrees, the noise is with probability 0.99 less than k degrees.

Reducing the effects of noise should be concentrated at the localization stage as the ability to achieve formation control tasks is dependent on accurate localization. The study of the effects of noise on velocity consensus and formation shape control are omitted from this paper.

A. Noisy Localization Simulations

In this section, Monte Carlo simulations are conducted to investigate the performance of the algorithm with varying levels of noise.

1) Simulation 1.1: In Simulation 1.1, we investigate the effect of increasing n from 10 to 510, for varying levels of noise. For each n, and for each k, we randomly generated 100 different scenarios. The grid step is a constant $\delta\omega = 0.001\,\text{rads}^{-1}$. We conducted 100 Monte Carlo simulations for each of the randomly generated scenarios. For each n and k, the value of $\|e\|$ was averaged over the 100 Monte Carlo simulations and 100 different scenarios. Figure 11 shows the average value of $\|e\|$ as n increases, for different k.

2) Simulation 1.2: In Simulation 1.2, we investigate the effect of decreasing $\delta\omega$ from $0.01\,\text{rads}^{-1}$ to $0.0001\,\text{rads}^{-1}$. For each $\delta\omega$, and for each k, we randomly generated 100 different scenarios. We conducted 100 Monte Carlo simulations for each of the randomly generated scenarios. The number of equations is a constant $n = 300$. Figure 12 shows that $\|e\|$ decreases with decreasing step size. Unsurprisingly, there is an asymptotic limit where decreasing $\delta\omega$ does not yield a performance increase. We also see that the performance is inversely proportional to the level of noise.

Note that e has been defined as the vector of absolute percentage error for all unknowns. This means that the absolute percentage error of each of the unknowns $y_2, x_2, r_2, \omega_2, \phi_2, v_{xr}, v_{yr}$, is less than $\|e\|$. It is clear from our extensive Monte Carlo simulations that a good performance from the algorithm is obtainable in the presence of noisy $\theta(t)$ if we have a large number of measurements (equivalently a large value for n), and a small $\delta\omega$.

Remark 2: It is well known in signal processing literature that a problem can occur from sampling a signal too frequently. If f_s is high compared to ω_i, e.g. if $f_s = 50$ Hz while ω_1, ω_2 are both below $1\ \text{rads}^{-1}$, then the values of $b(t), c_i(t), d_i(t)$ between successive points in time $t_1, t_2, ..., t_n$ change by a small amount. If the change is so small that no meaningful information can be extracted, then the noisy system of equations will fail to solve correctly. Thus, while Fig. 11 show the benefits of having a larger value for n, the reader is reminded that this comes from measuring $\theta(t)$ for longer periods, rather than sampling at a higher frequency.

B. Localization Bias

Estimation bias in localization problems can occur when there is additive zero mean noise in the measurements and the function mapping the measurements to the localization solution is nonlinear [33]. The simulations in this section are conducted to investigate this bias. Monte Carlo simulations were conducted using the geometric scenario in Section V-A.

1) Simulation 2.1: In Simulation 2.1, we set $n = 600$. *Different to all previous simulations*, we postulate that Agent 1 knows $\omega_2 = -0.2615\,\mathrm{rads}^{-1}$ *exactly*. The reason for this will become apparent in the sequel. For the scenario outlined in Section V-A, random white noise of varying k was added to $\theta(t)$ and the localization solution obtained. This was repeated for 1000 Monte Carlo iterations. Figure 13 shows the estimated probability density function of the localization solution value for y_2 as k varies, with the true y_2 value plotted as a vertical dotted black line. We observe that the bias increases proportionally to σ^2; this is consistent with what has been reported in recent literature [33].

2) Simulation 2.2: In Simulation 2.2, we set $n = 600$ and $k = 0.6°$. We vary $\delta\omega$, conduct 1000 Monte Carlo simulations and plot the estimated probability density function of y_2. Note that $\omega_2 = -0.2615\,\mathrm{rads}^{-1}$ as in Simulation 2.1 but Agent 1 knows only that $\omega_2 \in [-0.6, -0.23]\,\mathrm{rads}^{-1}$ as in Section V-A. In other words, we are investigating the relationship between localization bias and search grid quantization. Figure 14 shows that, when the grid size is sufficiently large such that ω_2 falls between two grid points ($\delta\omega = 0.01, 0.005, 0.001$), then there is no obvious relationship between $\delta\omega$ and the bias (as opposed to bias being proportional to $\delta\omega$). If $\delta\omega$ is sufficiently small such that ω_2 falls on a grid point ($\delta\omega = 0.0005, 0.0001$), then the bias is approximately equal for different $\delta\omega$.

The assumption that *Agent 1 knows ω_2 exactly* was applied in Simulation 2.1 to remove $\delta\omega$ as a factor. It is now apparent that the bias is dependent both on the level of noise and on the discretization of the grid $\widehat{\mathbf{\Omega}}_2$. Due to space limitations, we do not present the simulation plots for x_2, r_2, ϕ_2, v_{xr} or v_{yr} but they show similar results as those in Fig. 13 and 14.

Remark 3: The paper [33] provided a systematic way to compute and reduce localization/estimation bias. However, the method does not account for the bias arising from grid size, and the relationship between estimation bias and grid size is currently unclear. In order to meaningfully employ the bias reduction techniques in [33], $\delta\omega$ must be sufficiently small such that ω_2 falls on a grid point in $\widehat{\mathbf{\Omega}}_2$. The necessarily extensive details are omitted due to space limitations.

We may consider consecutively using two different $\delta\omega$. The first search grid has larger step sizes, and obtains a value for $\widehat{\omega}^*_{2,i}$ associated with $\boldsymbol{x}^*$. The second grid is centered around the value $\widehat{\omega}^*_{2,i}$ obtained from the first search and has a much smaller $\delta\omega$. This will increase accuracy without sacrificing computational time *and* increases the chance that ω_2 falls on a grid point of the second grid, eliminating bias due to grid discretization. Of course, the first grid must be sufficiently fine to ensure that the first grid search captures the convergence region of the global minimum; Fig. 12 indicates that $\delta\omega = 0.001\,\mathrm{rads}^{-1}$ is sufficient.

VII. Conclusion

A localization technique in the situation where agents in a formation have limited sensing capabilities has been presented. Specifically, we show that localization is possible, using bearing-only measurements recorded over a time interval, when the motions of all agents are restricted to be a linear combination of a circular orbit and translation of the circle center, and the agents have approximate knowledge of other agents' angular velocities. The localization solution involves linearization of a system of equations by a grid of angular velocity values so that in fact, a set of such systems is solved. The localization algorithm is then applied to an undirected triangle agent formation to achieve 1) velocity consensus and 2) velocity consensus combined with formation shape control. An adaptive algorithm is developed to ensure the agent's trajectory has a sufficiently large radius of curvature for realizable motion. Simulation results involving noiseless and noisy bearing measurements are provided for localization only, for velocity consensus and for velocity consensus combined with formation shape control. In particular, extensive Monte Carlo simulations are conducted which show the localization algorithm is robust to small measurement noise, with a bias.

Appendix A
Proof of linear dependence when $\omega_1 = \omega_2$

In the situation where $\omega_1 = \omega_2 = \omega$, equation (1) can be expressed as $\sin(\theta(t))/\cos(\theta(t)) = V(t)/U(t)$ where $V(t) = y_2 + v_{yr}t + a\sin(\omega t) + b\cos(\omega t)$, $U(t) = x_2 + v_{xr}t + a\cos(\omega t) - b\sin(\omega t)$. Note the substitutions $a = r_2\cos(\phi_2) - r_1\cos(\phi_1)$ and $b = r_2\sin(\phi_2) - r_1\sin(\phi_1)$ are constants. Construct the 6×6 submatrix $\widetilde{\boldsymbol{A}}$ by taking any six rows from $\boldsymbol{A}_i$ in equation (4) and letting these be the rows in $\widetilde{\boldsymbol{A}}$. Each row i of $\widetilde{\boldsymbol{A}}$ has an associated time value $\widetilde{t}_i$. Note that in general $\widetilde{t}_i \neq t_i$. Multiplying each element in the i^{th} row of $\widetilde{\boldsymbol{A}}$ by $U(\widetilde{t}_i)/\cos(\theta(\widetilde{t}_i))$, the i^{th} row of $\widetilde{\boldsymbol{A}}$ is given by

$$\widetilde{\boldsymbol{A}}(i) = \begin{bmatrix} U(\widetilde{t}_i) & V(\widetilde{t}_i) & \widetilde{t}_i U(\widetilde{t}_i) & \widetilde{t}_i V(\widetilde{t}_i) & C(\widetilde{t}_i) & D(\widetilde{t}_i) \end{bmatrix}$$

where $C(t_i) = U(t_i)\cos(\omega t_i) + V(t_i)\sin(\omega t_i)$ and $D(t_i) = U(t_i)\sin(\omega t_i) - V(t_i)\cos(\omega t_i)$. Note that, although $U(\widetilde{t}_i)/\cos(\theta(\widetilde{t}_i))$ could be infinite, it is appropriate here since it is merely a tool to analyse $\widetilde{\boldsymbol{A}}$. *Mathematica* delivers the conclusion that for any $\widetilde{t}_1, \widetilde{t}_2, ..., \widetilde{t}_6 \in \mathbb{R}_{>0}$, $\det(\widetilde{\boldsymbol{A}}) = 0$. It follows that $\widetilde{\boldsymbol{A}}$ is not full rank which implies that all 6×6 minors of $\boldsymbol{A}_i$ are zero. We therefore conclude that $\boldsymbol{A}_i$ does not have full rank and $\boldsymbol{u} = \boldsymbol{A}_i\boldsymbol{x}_i$ has no unique solution.

Acknowledgment

This work was supported by the Australian Research Council (ARC) under the grants DP-130103610 and DP-160104500, by the National Natural Science Foundation of China (grant 61375072), and by Data61-CSIRO (formerly NICTA).

References

[1] Y. Cao, W. Yu, W. Ren, and G. Chen, "An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, Feb 2013.

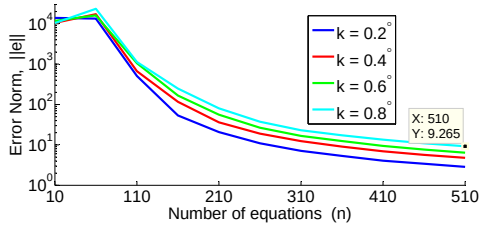


Figure 11. Localization error as the number of equations varies, with noisy $\theta(t)$ and $\delta\omega = 0.001\,\text{rads}^{-1}$.

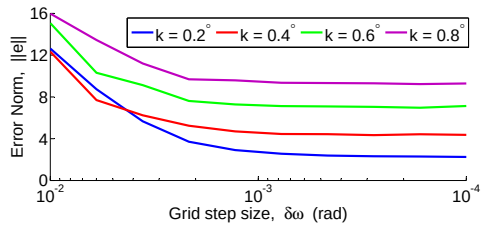


Figure 12. Localization error as the number of equations varies, with noisy $\theta(t)$ and $n = 300$.

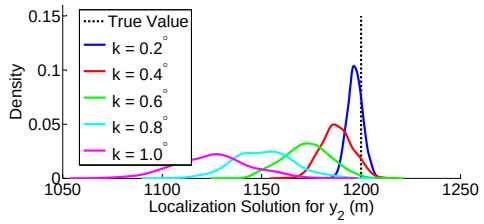


Figure 13. Plot of the localization solution for y_2 against density in a Monte Carlo simulation, varying the noise, k.

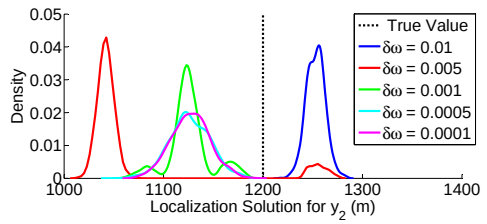


Figure 14. Plot of the localization solution for y_2 against density in a Monte Carlo simulation, varying step size, $\delta\omega$.

[2] P. B. Sujit and D. Ghose, "Search using multiple uavs with flight time constraints," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 40, no. 2, pp. 491–509, April 2004.

[3] Q. Zhang, L. Lapierre, and X. Xiang, "Distributed control of coordinated path tracking for networked nonholonomic mobile vehicles," *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 1, pp. 472–484, Feb 2013.

[4] W. Ren and Y. Cao, *Distributed Coordination of Multi-agent Networks*. Springer, 2011.

[5] B. D. O. Anderson and C. Yu, "Range-only sensing for formation shape control and easy sensor network localization," in *Control and Decision Conference (CCDC), 2011 Chinese*. IEEE, 2011, pp. 3310–3315.

[6] H. Bai, M. Arcak, and J. Wen, *Cooperative Control Design*. New York: Springer, 2011, vol. 89.

[7] R. Olfati-Saber and P. Jalalkamali, "Coupled distributed estimation and control for mobile sensor networks," *Automatic Control, IEEE Transactions on*, vol. 57, no. 10, pp. 2609–2614, Oct 2012.

[8] K. Doğançay, "Self-localization from landmark bearings using pseudolinear estimation techniques," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 50, no. 3, pp. 2361–2368, July 2014.

[9] I. Shames, B. Fidan, B. D. O. Anderson, and H. Hmam, "Cooperative self-localization of mobile agents," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 47, no. 3, pp. 1926–1947, July 2011.

[10] T. Eren, W. Whiteley, and P. N. Belhumeur, "Using Angle of Arrival (Bearing) Information in Network Localization," in *45th IEEE Conference on Decision and Control*. Citeseer, 2006.

[11] A. Bishop, B. D. O. Anderson, B. Fidan, P. Pathirana, and G. Mao, "Bearing-only localization using geometrically constrained optimization," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 45, no. 1, pp. 308–320, Jan 2009.

[12] R. Sharma, R. Beard, C. Taylor, and S. Quebe, "Graph-based observability analysis of bearing-only cooperative localization," *Robotics, IEEE Transactions on*, vol. 28, no. 2, pp. 522–529, April 2012.

[13] K. Doğançay, "Uav path planning for passive emitter localization," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 48, no. 2, pp. 1150–1166, APRIL 2012.

[14] G. P. Huang, K. X. Zhou, N. Trawny, and S. Roumeliotis, "Bearing-only target tracking using a bank of map estimators," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 4998–5005.

[15] L. Zhang, M. Ye, B. D. O. Anderson, P. Sarunic, and H. Hmam, "Cooperative Localisation of UAVs in a GPS-denied Environment Using Bearing-Only Measurements," in *to appear in IEEE 55th Annual Conference on Decision and Control*.

[16] Z. Wang, J.-A. Luo, and X.-P. Zhang, "A Novel Location-Penalized Maximum Likelihood Estimator for Bearing-Only Target Localization," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6166–6181, 2012.

[17] B. Ristic and M. S. Arulampalam, "Tracking a manoeuvring target using angle-only measurements: algorithms and performance," *Signal processing*, vol. 83, no. 6, pp. 1223–1238, 2003.

[18] D. Mušicki, "Bearings only single-sensor target tracking using Gaussian mixtures," *Automatica*, vol. 45, no. 9, pp. 2088–2092, 2009.

[19] M. E. Campbell and W. W. Whitacre, "Cooperative Tracking Using Vision Measurements on SeaScan UAVs," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 4, pp. 613–626, 2007.

[20] K. E. Bekris, M. Glick, and L. E. Kavraki, "Evaluation of Algorithms for Bearing-Only SLAM," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 1937–1943.

[21] M. Cao, C. Yu, and B. D. O. Anderson, "Formation control using range-only measurements," *Automatica*, vol. 47, no. 4, pp. 776–781, 2011.

[22] B. Jiang, M. Deghat, and B. D. O. Anderson, "Simultaneous Velocity and Position Estimation via Distance-only Measurements with Application to Multi-Agent System Control," *IEEE Transactions on Automatic Control*, vol. PP, no. 99, pp. 1–1, 2016.

[23] D. Zelazo, A. Franchi, H. H. Bülthoff, and P. R. Giordano, "Decentralized rigidity maintenance control with range measurements for multi-robot systems," *The International Journal of Robotics Research*, p. 0278364914546173, 2014.

[24] T. Eren, "Formation shape control based on bearing rigidity," *International Journal of Control*, vol. 85, no. 9, pp. 1361–1379, 2012.

[25] S. Zhao and D. Zelazo, "Bearing Rigidity and Almost Global Bearing-Only Formation Stabilization," *IEEE Transactions on Automatic Control*, vol. 61, no. 5, pp. 1255–1268, 2016.

[26] M. Ye, B. D. O. Anderson, and C. Yu, "Multiagent Self-Localization Using Bearing-only Measurements," in *Proceedings of IEEE 52nd Annual Conference on Decision and Control (CDC)*. IEEE, 2013, pp. 2157–2162.

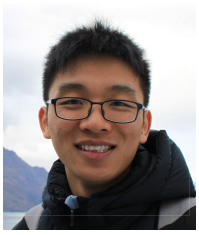
[27] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer, and C. J. Taylor, "A vision-based formation control framework," *Robotics and Automation, IEEE Transactions on*, vol. 18, no. 5, pp. 813–825, 2002.

[28] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan 2007.

[29] L. Krick, M. E. Broucke, and B. A. Francis, "Stabilisation of infinitesimally rigid formations of multi-robot networks," *International Journal of Control*, vol. 82, no. 3, pp. 423–439, 2009.

[30] B. D. O. Anderson, Z. Lin, and M. Deghat, "Combining distance-based formation shape control with formation translation," *Developments in Control Theory Towards Glocal Control*, pp. 121–130, 2012.

[31] T. Banchoff and S. Lovett, *Differential Geometry of Curves and Surfaces*. Natick, Massachusetts: A K Peters Ltd, 2010.
[32] C. Forbes, M. Evans, N. Hastings, and B. Peacock, *Statistical Distributions*. John Wiley & Sons, 2011.
[33] Y. Ji, C. Yu, J. Wei, and B. D. O. Anderson, "Localization bias reduction in wireless sensor networks," *Industrial Electronics, IEEE Transactions on*, vol. 62, no. 5, pp. 3004–3016, 2015.



Mengbin Ye was born in Guangzhou, China. He received the B.E. with First Class Honours in mechanical engineering. He is currently pursuing his PhD in control engineering at the Australian National University, Canberra.

He was a summer research scholar at the Australian National University in 2012 and 2013. He has been a visiting PhD student at Zhejiang University, Osaka City University and University of California, Riverside. His current research interests include consensus and synchronisation of multi-agent systems, opinion dynamics and social networks, and localisation using bearing measurements.



Changbin Yu received the B.Eng (Hon 1) degree from Nanyang Technological University, Singapore in 2004 and the Ph.D. degree from the Australian National University, Australia, in 2008. Since then he has been a faculty member at the Australian National University and subsequently holding various positions including a specially appointed professorship at Hangzhou Dianzi University.

He had won a competitive Australian Postdoctoral Fellowship (APD) in 2007 and a prestigious ARC Queen Elizabeth II Fellowship (QEII) in 2010. He was also a recipient of Australian Government Endeavour Asia Award (2005) and Endeavour Executive Award (2015), Chinese Government Outstanding Overseas Students Award (2006), Asian Journal of Control Best Paper Award (20062009), etc. His current research interests include control of autonomous aerial vehicles, multi-agent systems and humanrobot interactions. He is a Fellow of Institute of Engineers Australia, a Senior Member of IEEE and a member of IFAC Technical Committee on Networked Systems. He serves as a subject editor for International Journal of Robust and Nonlinear Control and was an associate editor for System & Control Letters and IET Control Theory & Applications.



Brian D.O. Anderson (M'66-SM'74-F'75-LF'07) was born in Sydney, Australia, and educated at Sydney University in mathematics and electrical engineering, with PhD in electrical engineering from Stanford University in 1966.

He is an Emeritus Professor at the Australian National University, and Distinguished Researcher in Data61-CSIRO (previously NICTA) and Distinguished Professor at Hangzhou Dianzi University. His awards include the IEEE Control Systems Award of 1997, the 2001 IEEE James H Mulligan, Jr Education Medal, and the Bode Prize of the IEEE Control System Society in 1992, as well as several IEEE and other best paper prizes. He is a Fellow of the Australian Academy of Science, the Australian Academy of Technological Sciences and Engineering, the Royal Society, and a foreign member of the US National Academy of Engineering. He holds honorary doctorates from a number of universities, including Université Catholique de Louvain, Belgium, and ETH, Zürich. He is a past president of the International Federation of Automatic Control and the Australian Academy of Science. His current research interests are in distributed control, sensor networks and econometric modelling.