Decentralized Cooperative Control of Heterogeneous Vehicle Groups

H. G. Tanner *, D. K. Christodoulakis

University of New Mexico, Department of Mechanical Engineering, Albuquerque, NM 87131-0001, USA

Abstract

We coordinate in discrete time the interaction of two heterogeneous groups of mobile agents: a group of ground vehicles (UGVs) and a group of aerial vehicles (UAVs). The ground agents interact with each other through time-invariant, nearest-neighbor rules. They synchronize their velocities through a specific communication protocol, and maintain cohesion and separation behavior by means of inter-agent potential forces. Ground vehicles estimate their formation's centroid using only locally available delayed information. That same information is transmitted to the aerial group, which orbits above the ground formation's centroid, while avoiding midair collisions. Stability of the ground group motion is established in a Lyapunov framework. A Lyapunov analysis is also used to ensure that UAVs track the ground group's centroid.

Key words: Cooperative control, Agents and autonomous systems, Autonomous robots

1 Introduction

This work is motivated by the need to automate cooperative intelligence surveillance and reconnaissance (ISR) missions, which involve unmanned vehicles (ground and aerial) operating autonomously in order to achieve a common objective.

Envision a scenario where a safe corridor needs to be established through hostile terrain. A group of unmanned ground vehicles (UGVs) is sent to clear a

* Corresponding author.
Email addresses: tanner@unm.edu (H. G. Tanner), dchristo@us.ibm.com (D. K. Christodoulakis).

path through the area. To provide aerial coverage for the ground vehicles, and to surveil the surrounding area for threats, a group of unmanned aerial vehicles (UAVs) is dispatched. To realize such a scenario, several control objectives must be met: UGVs must move in unison along collision free paths; UAVs need to establish stable orbits above and around the UGV group, while avoiding midair collisions.

Several research groups have developed UGV prototypes [1], [2]. Among them, is the Intelligent Systems and Robotics Center of the Sandia National Laboratories [3]. The typical control objectives in order to coordinate a group of UGVs to exhibit the type of flocking and schooling behaviors observed in biological systems, are cohesion and separation in terms of subsystem positions, as well as velocity alignment [4]. Ideally, UGVs need to achieve these objectives by exchanging local information without using too much communication bandwidth. Observing these two design constraints facilitates implementation of the coordination schemes, and enables the latter to scale better with the size of the group.

In the cases considered in recent literature (cf. [4–7]), information is assumed to be exchanged and processed instantaneously. But in a realistic scenario, neither communication nor computation can happen instantly. When nearest-neighbor interaction is subject to communication delays, performance is expected to suffer [8], [9]. In previous work [10] we showed that for a particular model of discrete time nearest-neighbor interaction with delayed information, synchronization of velocities to a common vector is still possible, irrespectively of the size of the delay. Parallel efforts [11], [12], [13] are in agreement with our results.

Cohesion and separation are set as control objectives because on one hand vehicles need to avoid collisions with each other, and on the other they need to stay physically close to give the appearance of a group. For collision avoidance, gyroscopic forces have been used as a control input component in UAV systems [14], [15]. In [14] the mobile agents react only to the nearest obstacle in front of them. In [15] there is formal proof that collision avoidance can always be achieved for the case of two agents.

Most of the work in UAV coordination focuses on single UAV missions, or cooperation within formations of multiple UAVs. Visual servoing is proposed in [16], and in [17], involving remote guidance and collision avoidance respectively. Obstacle avoidance techniques, in cases where agents are equipped with infrared and visual sensors, are implemented in [18]. Laser range finders are used in [19] for static and dynamic obstacle avoidance aided by geometric algorithms, while object recognition for camera-based control is suggested in [20].

Reported work on interaction of heterogeneous teams of multiple agents is

fairly limited. Initial efforts in this direction include cooperation between one UGV and one UAV [1], [16]. Most of the work deals exclusively with either ground or aerial formations. For the type of ISR missions envisioned in the motivating scenario discussed above, cooperation between teams of mobile agents of different modalities is required. Depending on the specific vehicle capabilities, teams can have different team goals, but all should support the common objective.

This paper extends the work in [10], by enabling the UGV group to not only synchronize its velocity vectors to a common one, but stabilize the distances between communicating agents as well. This is done assuming that vehicles communicate over a TDMA-type communication scheme, where each broadcasts its state information to neighbors in turns. As a result, the neighbor information each vehicle uses in its control loop is delayed. A Lyapunov analysis establishes that the cohesion and separation forces introduced do not affect the convergence of velocity vectors. We develop an algorithm where each UGV estimates the ground group's centroid using only local delayed information. Then we establish a control interconnection between the UAV and UGV groups, by means of a unidirectional (ground to air) transmission of (local) information. UAVs achieve their objective by "listening" to the broadcast of the closest ground vehicles. This objective is formalized as stabilization to circular orbits of different radii over the (moving) centroid of the ground group.

The rest of the paper is organized as follows: in Section 2 we motivate our control design and outline the control specifications that we meet. We describe the dynamics of the two distinct groups in Section 3, we refine the control specifications in view of the equations of motion, and we state the underlying assumptions. Section 4 presents our control design approach for meeting the control specifications of the UGV team. We prove that ground vehicle velocities will be synchronized in finite time (due to the discrete-time nature of the closed loop implementation) and that inter-vehicle forces will vanish at steady state. In addition, we demonstrate how ground vehicles are able to estimate their group's centroid without using global information. Section 5 describes how UAVs stabilize to circular orbits and track the ground group's centroid. In Section 6 we present simulation results that illustrate how the two groups cooperate and verify the convergence properties of our controllers. Section 7 concludes the paper by summarizing the results presented.

2 Motivation and Problem Statement

In the scenario we envision, UGVs must establish a secure corridor by passing over a segment of an area, supported by air from a UAV group which scans the area around the ground group in search for possible threats.

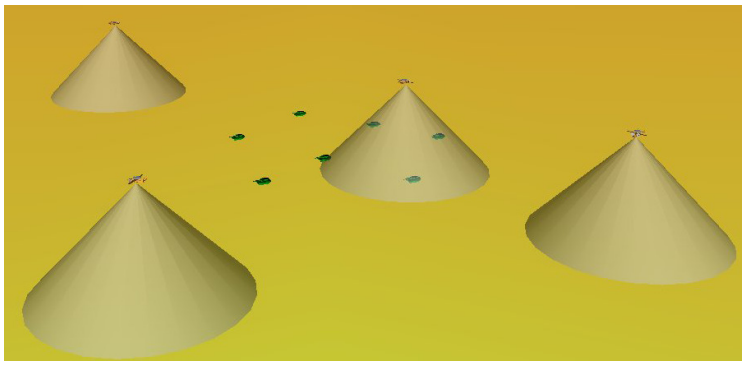

Fig. 1. Representation of the combined intelligence-surveillance- reconnaissance mission: ground vehicles move in formation while aerial vehicles provide aerial coverage by scanning the ground using their sensors. The aerial team provides early warning against threats on the ground. The image is a VR animation of the simulation scenario realized by the proposed closed loop controllers.

One set of action items in order to realize such a scenario, is the following:

(1) UGVs synchronize their velocities to a common vector and move in unison.
(2) UGVs spread out, regulating inter-vehicle distances between communicating vehicles.
(3) UGVs estimate the centroid of their formation.
(4) UAVs establish stable orbits around the UGV team centroid.
(5) UAVs avoid midair collisions.

The control design approach described in this paper aims at meeting the aforementioned specifications, without discussing in detail how a particular mission scenario can be be implemented.

3 Overview of Approach

We assume the existence of a communication network over the vehicles, the topology of which is fixed. The dynamics of the ground vehicles are described by double integrators, on the premises that UGVs are more maneuverable compared to aerial vehicles.

$$\dot{x}_i = v_i,$$
$$\dot{v}_i = u_i.$$

In discrete-time these equations take the form

$$r_i(k+1) = r_i(k) + u_i(k)T \tag{1}$$
$$u_i(k+1) = u_i(k) + \alpha_i(k)T, \tag{2}$$

where T is the sampling period and $i = 1, \ldots, N$ is the index of the particular UGV. To eliminate communication delays due to packet collision when network-neighboring vehicles transmit simultaneously, we adopt a TDMA-type protocol, where each node is assigned to a specific time slot. All possible communication links are described by an undirected graph:

Definition 1 *The interconnection graph, $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, is an undirected graph with:*

- *a set of nodes $\mathcal{V} = \{1, \ldots, N\}$, indexed by the agents in the group, and*
- *a set of edges $\mathcal{E} = \{(i, j) \in \mathcal{V} \times \mathcal{V}\}$, which contains unordered pairs of nodes corresponding to agents that can communicate.*

Whenever $(i, j) \in \mathcal{E}$, we can write $i \sim j$. The set of all nodes adjacent to i is denoted $\mathcal{N}_i$. Vehicles can communicate only with a fixed subset of their group mates and communication links need not be "live" at all times. Since each vehicle broadcasts its state information at a particular time step, its neighbors have to use their most recent information about it until it retransmits. Thus, information used for control is necessarily delayed.

We design UGV controllers that generate flocking behavior, by steering each vehicle velocity to the average of its neighbors'. The latter, however, are not current since they have been communicated to the vehicle some time steps back. During this velocity alignment process, we constrain the UGVs to use their own current state *only after* they have broadcast them to their neighbors, even if it could be readily available. Thus, each vehicle introduces time delays in its own state information being used by its controller. This is done to enforce consistency in the use of communicated information, allowing neighboring vehicles to use the same (delayed) information about each other. Such consistency is critical for the cohesion/separation forces we introduce in the control laws of this paper to remain symmetric between interacting vehicles.

In [10] we showed that when vehicles update their velocities to the (delayed) average of their neighbors', and communicate according to the aforementioned communication protocol, the velocities of all vehicles still align exponentially fast. For completeness, the method we proposed in [10] is briefly described in Section 4.1.

To regulate the distance between communicating UGVs we introduce a quadratic potential field. In Section 4.2 we build upon the velocity synchronization controller of [10] by including this potential field in the control law, and we subsequently analyse the stability of the closed loop system in a Lyapunov framework.

We show that the centroid of the ground formation can be estimated in a distributed fashion. Similarly to velocity synchronization, each vehicle averages

the estimates of its neighbors, and propagates this estimate through the quasi-steady state group dynamics. Collision avoidance for the UGVs is performed using communicated information exclusively; non-communicating agents are not aware of each other's presence. Centroid estimation is subject to the same communication delays. UAVs are assumed to pick up the centroid broadcasts of ground vehicles, and use it to update their orbiting controller. At steady state, all ground vehicles have the same centroid estimate, so it does not matter where the centroid broadcast is originating (Section 4.3).

Every UAV orbits over the centroid estimate. The orbit is designed to be circular at a common altitude, with desired radius d_i and speed V. As the ground formation moves, these orbits become spirals. We use unicycle dynamics to represent the motion of the UAVs [21], since the turning radius limitations in aerial flight cannot be ignored, even at a first approximation.

$$\dot{x}_i = V \cos\theta_i \tag{3a}$$
$$\dot{y}_i = V \sin\theta_i \tag{3b}$$
$$\dot{\theta}_i = \omega_i, \tag{3c}$$

where $i = 1, \ldots, M$ is the index of the particular UAV. The control input to UAV i is the angular velocity ω_i. In Section 5.1 we design angular velocity control laws that establish stable circular orbits for the aerial vehicles.

Every UAV has a sensing range of radius R_c, within which any other aerial vehicle is detected and treated as a collision threat. All UAVs within distance R_c from each other perform collision avoidance maneuvers using the maximum control effort available, ω_{max}. In Section 5.2 we specify a collision avoidance direction for each UAV, by weighing the effect of each possible collision threat according to proximity. In Section 5.3 we combine the orbiting and collision avoidance controllers into a single discrete-time switching control scheme for each UAV.

4 Decentralized UGV control

In Section 4.1 we show how the velocities of UGVs are synchronized by interlacing the control update with a specific communication protocol. For the augmented system that includes the delayed states, we show that a certain state transition matrix is ergodic. Under the assumptions made on inter-vehicle communication, we ensure that vehicles use the same information regarding each other and thus, inter-vehicle forces are symmetric. The symmetry allows us to show in Section 4.2, that the convergence of the velocity vectors is not affected. The same update mechanism that was shown to synchronize the velocities is used to estimate, in a distributed fashion, the group's centroid in

Section 4.3.

4.1 UGV velocity synchronization

In discrete time, UGV positions evolve according to (1)-(2). If we include the time period T, the control law of [22] takes the form

$$\alpha_i(k) = \left\{ \frac{1}{1+|\mathcal{N}_i|} \left(u_i(k) + \sum_{j \in \mathcal{N}_i} u_j(k) \right) \right\} \frac{1}{T}.$$

Stacking the current velocity vectors u_i in vector u, and assuming for the moment that information is communicated instantly, the discrete-time dynamics of the velocities of the UGV group in state-space form become

$$u(k+1) = \{[(I+D)^{-1}(A+I)] \otimes I\} u(k), \tag{4}$$

where $\otimes$ denotes the Kronecker product, and A, D are the adjacency and valency matrices of $\mathcal{G}$, respectively [23].

We impose a set of communication rules on (4), which allow each agent to broadcast only one at a time step. To keep track of ordered transmissions we use a binary matrix S (henceforth called the *communication matrix*), with rows indexed by the agents and columns indexed by time steps. Each column of S, denoted S_j, has only one nonzero element $s_{ij} = 1$, indicating that vehicle i is transmitting at step j. The columns of S are ordered, with the leftmost column indicating the current time step and the rightmost referring to $N-1$ steps in the past. According to the order of transmission, agent i for which $s_{iN} \neq 0$ is transmitting in the following time step . Thus, as time evolves, the columns of S shift from left to right, with the rightmost column being recycled to the left. For example, consider a group of four agents, where at time step k agent 1 broadcasts. Before agent 1 agent 4 did so, following agent 3 and agent 2. The communication matrices for steps k and $k+1$ are

$$S(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \qquad S(k+1) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{5}$$

This construction bears much resemblance to the circulant matrices used in [24], but unfortunately, these matrices do not satisfy the defining properties of circulant matrices. (No obvious modification of our protocol was able to utilize this construction.)

To enforce these communication rules on (4), augment the velocity vector with the delayed information as

$$U(k) \triangleq \begin{bmatrix} u(k)^T \; u(k-1)^T \; \cdots \; u(k-N+1)^T \end{bmatrix}^T,$$

where $U(k)$ is the stack vector of $u(k)$ through $u(k-N+1)$. The state transition equations for U are

$$U(k+1) = \begin{bmatrix} \{[(D+I)^{-1}(I+A)]\otimes I\}\bar{S} \\ \begin{smallmatrix} I & 0 & \dots & \dots & 0 & 0 \\ 0 & I & \dots & \dots & 0 & 0 \\ 0 & 0 & I & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & \dots & I & 0 \end{smallmatrix} \end{bmatrix} U(k) \triangleq H(k)U(k), \tag{6}$$

where $\bar{S}$ is constructed by taking Kronecker products of its columns with their transposes

$$\bar{S} = [S_1 \otimes S_1^T \cdots S_N \otimes S_N^T].$$

Multiplying the augmented velocity vector with $\bar{S}$ simply selects the delayed velocities that are to be used in the update equation at that particular time step.

Without loss of generality, we order the transmissions according to the indexing of the agents. For the example of four agents where $S(k)$ is given by (5), matrix $\bar{S}$ has the form

$$\bar{S} = \begin{bmatrix} 1&0&0&0 & 0&0&0&0 & 0&0&0&0 & 0&0&0&0 \\ 0&0&0&0 & 0&0&0&0 & 0&0&0&0 & 0&1&0&0 \\ 0&0&0&0 & 0&0&0&0 & 0&0&1&0 & 0&0&0&0 \\ 0&0&0&0 & 0&0&0&1 & 0&0&0&0 & 0&0&0&0 \end{bmatrix}.$$

Note that $S(k)$ (and consequently $\bar{S}$) is N-periodic. We call this period the *communication cycle*. Due to the time dependence, and despite the fact that the nominal system without delays is time invariant, (6) is time varying, and H is changing between time steps due to the columns of S shifting. In what follows, we express the first row of blocks of H as

$$F\bar{S} = \begin{bmatrix} f_1 \; f_2 \; \cdots \; f_N \end{bmatrix}.$$

Matrix H inherits some of the properties of F: it is row stochastic, and is non-negative. However, it does not have nonzero diagonal elements, and it is reducible. Without the latter two properties, the results of [22] do not apply, and stability of (6) cannot be directly established in the same way.

At initial time, $k = 0$, vehicles do not have any knowledge of their neighbors' states. We must allow one communication cycle to elapse for all vehicles to acquire local information over the network. Our asymptotic analysis therefore begins from time step $k = N$. The following theorem summarizes the result of Section 4.1:

Theorem 2 (Synchronization with communication delays) *Consider a multi-agent system with a time invariant connected interconnection graph $\mathcal{G}$, and discrete time agent dynamics described by*

$$r_i(k+1) = r_i(k) + u_i(k)T$$
$$u_i(k+1) = u_i(k) + \alpha_i(k)T.$$

If only one agent is allowed to broadcast its state information to its network neighbors at each time step, and agents use their last transmitted own state in their control laws, then in the closed loop system all velocity vectors will converge to a common vector.

PROOF. Note that H changes from time step k to $k+1$ due to the permutation of the first block of rows, $[f_1 \ \cdots f_N]$. Matrix H is periodic with a period of one communication cycle, that is, $H(t+N) = H(t)$. Sampling the trajectory at multiples of the communication cycle we obtain a sequence of points in state space. This sequence coincides with the evolution, from the same initial condition, of the time invariant discrete-time system having the state transition matrix for one step given by

$$M \triangleq H(N)H(N-1)\cdots H(1).$$

Since $H(k)$ is row stochastic and non-negative for every k, M will also be (row) stochastic and non-negative. For multiples of the communication cycle, the state transition matrix of (6), is written as

$$\Phi(kN, 1) = M^k.$$

A direct calculation verifies that:

$$M = \begin{bmatrix} G_1(N) & G_2(N) & \cdots & G_N(N) \\ G_1(N-1) & G_2(N-1) & \cdots & G_N(N-1) \\ \vdots & \vdots & & \vdots \\ G_1(1) & G_2(1) & \cdots & G_N(1) \end{bmatrix},$$

where $G_i(1) = f_i$, and the block elements of each column are given by the recursive formula

$$G_i(k) = \sum_{j=1} f_j(k) G_i(k-j), \tag{7}$$

where $G_i(r)$ for $r < 1$ are read from the blocks of $H(1)$ below $f_i(1)$ (For elements generated during the first communication cycle). Using (7), and given that due to the permutation of S,

$$f_{\mathrm{mod}(i,N,1)}\Big(\mathrm{mod}(j,N,1)\Big) = f_{\mathrm{mod}(i-1,N,1)}\Big(\mathrm{mod}(j-1,N,1)\Big), \tag{8}$$

in which $\mathrm{mod}(\cdot, N, 1)$ is the modulo function with offset 1, we find that in all lower diagonal matrix blocks of M, including $G_1(N), G_2(N-1), \ldots, G_N(1)$, we have $f_i(1)$ appearing in the block column i as the last term in the sum $\sum_{j=1} f_j(k)G_i(k-j)$. All terms in this sum are right multiplied by $f_i(1)$ and involve products of the form $f_j(1)f_i(1)$. Based on the special structure of $f_i(1)$ every such multiplication will result in either:

- A zero matrix, if the node corresponding to the nonzero column of $f_j(1)$ is not connected to the node of the nonzero column of $f_i(1)$ (meaning $f_{ji} = 0$) or if the nodes are connected,
- in a matrix having nonzero elements in every row of the *nonzero column of* $f_i(1)$ for which $f_j(1)$ has nonzero elements.

Putting it formally,

$$f_j f_i = \begin{bmatrix} 0 \ldots 0 & f_{1j} & 0 \ldots 0 \\ 0 \ldots 0 & f_{2j} & 0 \ldots 0 \\ \vdots \ \vdots & \vdots & \vdots \\ \underbrace{0 \ldots 0}_{j-1 \text{ times}} & f_{Nj} & 0 \ldots 0 \end{bmatrix} \begin{bmatrix} 0 \ldots 0 & f_{1i} & 0 \ldots 0 \\ 0 \ldots 0 & f_{2i} & 0 \ldots 0 \\ \vdots \ \vdots & \vdots & \vdots \\ \underbrace{0 \ldots 0}_{i-1 \text{ times}} & f_{Ni} & 0 \ldots 0 \end{bmatrix} = \begin{bmatrix} 0 \ldots 0 & f_{1j}f_{ji} & 0 \ldots 0 \\ 0 \ldots 0 & f_{2j}f_{ji} & 0 \ldots 0 \\ \vdots \ \vdots & \vdots & \vdots \\ \underbrace{0 \ldots 0}_{i-1 \text{ times}} & f_{Nj}f_{ji} & 0 \ldots 0 \end{bmatrix}$$

In other words, if the corresponding nodes are connected, the product $f_j(1)f_i(1)$ will inherit nonzero elements from the nonzero column of $f_j(1)$ and place them in the nonzero column of $f_i(1)$.

Therefore, the remaining terms in each sum $\sum_{j=1} f_j(k)G_i(k-j)$ of block column i below the block diagonal, can only add positive elements to the nonzero column of $f_i(1)$. Using (8), we can verify the existence of a term of the form $f_i(1)^2$ in the G_i blocks above the diagonal of M. This happens because the $f_j(k)$ terms in $\sum_{j=1} f_j(k)G_i(k-j)$ are equal to the corresponding blocks $f_r(1)$ due to (8), with r taking values in $\{N-(k-2), \ldots, N, 1\}$. Given now that the diagonal block is located in block-row $N-(k-1)$ some $f_j(k)$ are equal to $f_i(1)$.

Since the diagonal element of the nonzero column of $f_i(1)$ is always positive ($f_i(1)$ inherits this property from F), $f_i(1)^2$ maintains the nonzero elements of $f_i(1)$. Thus, for every $j = 1, \ldots, N$, we have a nonzero column of M, denoted

m_i, with the following structure

$$m_i = \Delta \begin{bmatrix} f_j(1) \\ \vdots \\ f_j(1) \end{bmatrix} + \lambda_i, \tag{9}$$

where Δ is a $N^2 \times N^2$ diagonal positive definite matrix, f_j is a column of F, and λ_i a nonnegative $N^2 \times 1$ vector.

The zero columns of M indicate that the corresponding delayed velocities do not contribute to the overall dynamics; they have some dynamics of their own which is decoupled from (6). One can obtain a reduced model by removing those states from the state model. The reduced system, viewed over multiples of the communication cycle, has the form

$$\bar{U}(k+1) = \bar{M}\bar{U}(k).$$

After removing the zero columns from M, one column corresponding to a (possibly delayed) state of different agent is always kept. Based on (9), $\bar{M}$ has the following form

$$\bar{M} = \Delta_1 \circ (FS) + \Lambda,$$

where Δ_1 is an $N \times N$ matrix with positive elements, $\circ$ denotes the Hadamard product, Λ is a nonnegative $N \times N$ matrix, and the product of F with the communication matrix S results in a permutation of the columns of F. With F being an irreducible matrix, since the underlying interaction graph of the nominal system is (strongly) connected, $\bar{M}$ will be too: Λ inserts additional edges in the underlying graph. With F having all diagonal elements nonzero, $\bar{M}$ is primitive [25]. In addition, it is (row) stochastic, because the removal of zero columns from M does not affect the sum of the row elements. A primitive and stochastic matrix is ergodic [22] which means that by definition

$$\lim_{k\to\infty} \bar{M}^k = \mathbf{1}c^T.$$

This establishes the convergence of the reduced state vector to $\xi\mathbf{1}$, for some $\xi \in \mathbb{R}$. If all current and some delayed states of agents converge to a common value, all current states also converge.

□

4.2 Group cohesion and collision avoidance

We build on the UGV velocity synchronization controller by adding a component that generates attractive and repulsive interaction between vehicles. For

this to be implemented we have to assume that vehicles communicate their position along with their velocity. With the introduction of a cohesion/separation component, f, we transfer the stability analysis to a Lyapunov framework, and the group dynamics are now given as

$$\begin{aligned} R(k+1) &= R(k) + U(k)T \\ U(k+1) &= \begin{bmatrix} \{[(D+I)^{-1}(I+A)]\otimes I\}\bar{S} \\ \begin{matrix} I & 0 & \dots & \dots & 0 & 0 \\ 0 & I & \dots & \dots & 0 & 0 \\ 0 & 0 & I & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & \dots & I & 0 \end{matrix} \end{bmatrix} U(k) + \begin{bmatrix} f(k) \\ 0 \\ \\ \vdots \\ \\ 0 \end{bmatrix}, \end{aligned} \tag{10}$$

where R is the stack vector of current and delayed (up to $N-1$ steps back) agent positions.

The stability analysis is performed by investigating differences of different quantities between consecutive time steps, expressed in the form $\Delta f(x) = f(x)\Big|_k - f(x)\Big|_{k-1}$. Using Taylor expansion, the total difference between time steps is expressed as

$$\Delta f(x) = \frac{\partial f}{\partial x}\delta x + \frac{\partial^2 f}{\partial x^2}\delta x^2 + \dots + \frac{\partial^n f}{\partial x^n}\delta x^n \dots,$$

where we set $\delta g\Big(x(k)\Big) = \frac{\partial g(x)}{\partial x}\Big|_{k-1} \delta x(k)$.

Conceptually, the interaction component of the input is be related to the gradient of a quadratic function of the distance $\|r_{ij}\|_2$ between agents i and j. This quadratic potential function V_{ij} is calculated based on the *delayed* information the agents have about each other. The stack vector of (delayed) relative positions and velocities used in such a calculation is given by

$$\begin{aligned} R_{ij} &\triangleq (B\otimes I)SR \\ U_{ij} &\triangleq (B\otimes I)SU, \end{aligned}$$

where B is the incidence matrix of $\mathcal{G}$. With r_{ij} being the component of R_{ij} that corresponds to the delayed relative vector between vehicles i and j, we define the quadratic potential function as

$$V_{ij} \triangleq a(\|r_{ij}\|_2 - r_d)^2, \quad a > 0,$$

where r_d is the desired distance which any vehicle should ideally maintain from its neighbors. In the sequel, unless explicitly specified, a vector norm is to be taken as the Euclidean norm. The difference $\Delta V_{ij}(k)$ between two time

steps is

$$\Delta V_{ij}(k) = V_{ij}(k) - V_{ij}(k-1) = \frac{\partial V_{ij}}{\partial \|r_{ij}\|}\Big|_{k-1} \delta\|r_{ij}\|(k) + \frac{\partial^2 V_{ij}}{\partial \|r_{ij}\|^2}\Big|_{k-1} \delta\|r_{ij}\|^2(k)$$

$$= 2a(\|r_{ij}(k-1)\| - r_d)\left(\frac{\partial\|r_{ij}\|}{\partial r_{ij}}\Big|_{k-1}\right)^T \delta r_{ij}(k) + 2a\left\{\left(\frac{\partial\|r_{ij}\|}{\partial r_{ij}}\Big|_{k-1}\right)^T \delta r_{ij}(k)\right\}^2$$

$$= 2a(\|r_{ij}(k-1)\| - r_d)\hat{r}_{ij}(k-1)\delta r_{ij}(k) + 2a\Big(\hat{r}_{ij}^T(k-1)\delta r_{ij}(k)\Big)^2$$

$$= 2a\Big[\,\|r_{ij}(k-1)\| - r_d + \hat{r}_{ij}(k-1)\delta r_{ij}(k)\Big] \cdot \hat{r}_{ij}^T(k-1)\delta r_{ij}(k),$$

where $\hat{r}_{ij}$ is the unit vector in the direction of r_{ij}. Our assumption that agents do not use their own state before they broadcast it to their neighbors, ensures that $\Delta V_{ij} = -\Delta V_{ji}$. Let us set

$$f_{ij} \triangleq 2a\Big[\,\|r_{ij}(k-1)\| - r_d + \hat{r}_{ij}^T(k-1)\delta r_{ij}(k)\Big]\hat{r}_{ij}^T(k-1)$$

$$= 2a\Big[\,\|r_{ij}\| - r_d + \hat{r}_{ij}^T \Delta u_{ij} T\Big]_{k-1} \hat{r}_{ij}^T(k-1),$$

and define the group's total potential as the sum of all interconnected agent potentials

$$\sum_{i\sim j} \Delta V_{ij}(k) = \frac{1}{2}\sum_{i=1}^{n}\sum_{i\sim j} f_{ij}^T\ \delta r_{ij}(k).$$

If we define $\text{sign}(U) \triangleq \Big(\text{sign}(U_1) \cdots \text{sign}(U_n)\Big)^T$, then the vector of cohesion/separation interactions is written as

$$f(k-1) \triangleq -\text{sign}\Big(U(k-1)\Big) \circ U(k-1) \circ T \sum_{i=1}^{n}\sum_{i\sim j} f_{ij}(k-1). \qquad (11)$$

Then we can state the main result of this section:

Theorem 3 (Cohesion and separation with communication delays) *The discrete-time system* (10)*, where f is defined in* (11)*, converges asymptotically to the set where $U(k-1) = \mathbf{1}c$ and $f = \mathbf{1}c'$ with $c, c' \in \mathbb{R}$.*

PROOF. Consider the Lyapunov function candidate

$$W(k) = \sum_{i=1}^{n}\sum_{i\sim j} V_{ij}(k) + \|U(k)\|_1 .$$

Use the fact that $\Delta\|U(k)\|_1 = \frac{\partial\|U\|_1}{\partial U}\Big|_{k-1} \delta U(k)$, and express the total difference

in the velocity term one-norm as

$$\Delta\|U(k)\|_1 = \operatorname{sign}\Big(U(k-1)\Big)^T H(k-1)U(k-1) \\ - \|U(k-1)\|_1 + \operatorname{sign}\Big(U(k-1)\Big)^T f(k-1).$$

By definition of the force vector, and since $U(k-1)T = \delta r_{ij}(k)$,

$$\operatorname{sign}(U(k-1))^T f(k-1) = -\sum_{i=1}^{n}\sum_{i\sim j}\Big(\frac{\partial V_{ij}}{\partial\|r_{ij}\|}\hat{r}_{ij}\Big)^T\Big|_{k-1}\delta r_{ij}(k).$$

Write the total difference in the Lyapunov function candidate between consecutive time steps as

$$\Delta W(k) = 2\sum_{i\sim j}\Delta V_{ij}(k) + \Delta\|U(k)\|_1 \\ = \operatorname{sign}\Big(U(k-1)\Big)^T H(k-1)U(k-1) - \|U(k-1)\|_1,$$

and note that since

$$\|H(k-1)U(k-1)\|_1 = \Big(\sum_{i=1}^{n}|h_{1i}u_i|+,\cdots+\sum_{i=1}^{n}|h_{ni}u_i|\Big)^T \\ \geq \operatorname{sign}\Big(U(k-1)\Big)H(k-1)U(k-1),$$

and $H(k-1)U(k-1)$ is a contraction, as shown in section 4.1, one has

$$0 > \|H(k-1)U(k-1)\|_1 - \|U(k-1)\|_1 \geq \\ \geq \operatorname{sign}\Big(U(k-1)\Big)H(k-1)U(k-1) - \|U(k-1)\|_1,$$

which shows that $\Delta W(k)$ is nonpositive.

The level sets of W define compact sets in the (R_{ij}, U_{ij}) space, due to connectivity of $\mathcal{G}$. The non-positiveness of $\Delta W(k)$ ensures that they are also invariant. By the discrete version of the invariance principle, (R_{ij}, U_{ij}) converges with time to the set of fixed points of $W(k)$, that is the fixed point of the map $H(k-1)U(k-1)$. Therefore,

$$U(k-1) = \mathbf{1}w, \qquad (10) \Rightarrow f = \mathbf{1}w'. \qquad \square$$

4.3 UGV group centroid estimation

The UAV team tracks the motion of the ground formation by following the latter's centroid. The UGV group's centroid, c, is generally expressed using information from all UGVs: $c = \frac{1}{N_g}\sum_{i=1}^{N_g} r_i$. To avoid collecting and processing

global information, we design a cooperative estimation scheme that allows each UGV to approximate the group's centroid coordinates using local information. Due to the use of delayed information, this estimate does not converge to the actual centroid, but the error is bounded.

The estimate of the centroid coordinates by vehicle i, denoted c_i, is updated in discrete time as follows

$$c_i(k+1) = \frac{1}{1+|\mathcal{N}_i|}\left(c_i(k) + \sum_{j\in\mathcal{N}_i} c_j(k)\right) + u_i T. \tag{12}$$

Once UGV i computes c_i according to (12), it broadcasts this estimate, together with its velocity and position. During the initial communication cycle, vehicles use the average position of their neighboring vehicles as an centroid estimate.

The centroid estimates of all UGVs are stacked to form the augmented centroid position estimation vector C. Collecting (12) for all vehicles we express the centroid update rule as

$$C(k+1) = H(k)C(k) + U(k)T,$$

where $H(k)$ is the same matrix used in (6). At steady state, we have $U(k) = \mathbf{1}w$ with $w \in \mathbb{R}$ (due to Theorem 3), so the dynamics of C are reduced to

$$C(k+1) = H(k)C(k) + \mathbf{1}wT. \tag{13}$$

At yet another time step,

$$\begin{aligned} C(k+2) &= H(k+1)C(k+1) + \mathbf{1}wT \\ &= H(k+1)H(k)C(k) + H(k+1)\mathbf{1}wT + \mathbf{1}wT \\ &= H(k+1)H(k)C(k) + 2Tw\mathbf{1}, \end{aligned}$$

so by induction we have

$$C(k+N) = H(k+N-1)\dots H(k)C(k) + (N-k+1)\mathbf{1}wT.$$

Exploiting the convergence properties of M which are established in Section 4.1 we have that $H(k+N-1)\dots H(k)C(k) = \mathbf{1}\lambda$, and since $\mathbf{1}wT \in \text{span}\{\mathbf{1}\}$ at steady state

$$\lim_{n\to\infty} C(n) \in \text{span}\{\mathbf{1}\}.$$

Thus, centroid estimates of all vehicles converge to a common vector. However, this vector cannot coincide with the actual group centroid, due to the use of delayed state information in the update rule (13). Nevertheless, this estimate tracks the true centroid with a bounded error, which ultimately depends on the size of the group, N_g.

5 Decentralized UAV control

UAVs fly over the UGV formation, providing aerial coverage and early warning of ground threats Each UAV covers a section of surface using their sensors (cameras, radar, etc) as shown in Figure 1.

For modeling purposes UAVs are considered point-mass vehicles. They all fly at a constant common altitude, and can measure the distance between them should they find themselves within a certain radius R_c. The translational speed of each UAV is fixed to V, to ensure that aerial vehicles do not stall.

To cover the UGV team from the air, the motion controller of UAV i is designed to make it follow circular orbits of radius d_i. Each UAV has a different d_i, and all orbits form co-centric circles, that are apart by a certain distance no smaller than R_c and not larger than twice the radius of the footprint of a UAV ground sensors (Figure 1).

In discrete-time, assuming that the control input ω_i remains constant between time steps, (3a), (3b) and (3c) become

$$x_i(k+1) = x_i(k) + \frac{\mathrm{V}\Big\{ \sin\Big(\theta_i(k) + \omega_i(k)T\Big) - \sin\theta_i(k)\Big\}}{\omega_i(k)} \tag{14}$$

$$y_i(k+1) = y_i(k) + \frac{\mathrm{V}\Big\{ -\cos\Big(\theta_i(k) + \omega_i(k)T\Big) + \cos\theta_i(k)\Big\}}{\omega_i(k)} \tag{15}$$

$$\theta_i(k+1) = \theta_i(k) + \omega_i(k)T \tag{16}$$

where ω_i is the control input for UAV i, assumed bounded.

5.1 Tracking the UGV group centroid

Let p_i denote the planar position vector of UAV i, c_i the position vector of the centroid estimate by UAV i, δ_i the current planar distance between UAV i and c_i, and let η_i be a unit vector normal to the velocity of UAV i. These quantities are expressed as follows

$$p_i = [x_i\ y_i]^T, \quad c_i = [c_{xi}\ c_{yi}]^T, \quad \delta_i = p_i - c_i, \quad \eta_i = [-\sin\theta_i\ \cos\theta_i]^T,$$
$$r_i = \frac{1}{2}\|\delta_i\|^2 = \frac{1}{2}\Big\{(x_i - c_{xi})^2 + (y_i - c_{yi})^2\Big\}.$$

The derivatives of r_i are expressed as

$$\begin{aligned}\dot{r}_i &= (c_i - p_i)^T(\dot{c}_i - \dot{p}_i)\\ \ddot{r}_i &= \mathrm{V}^2 + \mathrm{V}\delta_i^T\eta_i\theta_i - 2\dot{p}_i^T\dot{c}_i + \dot{c}_i^2 - 2\delta_i^T\ddot{c}_i.\end{aligned} \tag{17}$$

Note that the first two terms in (17) do not depend on derivatives of c_i; the last three terms do.

The orbiting behavior of UAV i around c_i is produced by setting

$$\omega_i = \frac{1}{\mathrm{V}\delta_i^T\eta_i}\Big\{ - \mathrm{V}^2 - q^2(r_i - \frac{1}{2}d_i^2) - 2q\dot{r}_i\Big\}, \tag{18}$$

where $q > 0$ is a control gain. Then the closed loop dynamics of the distance between UAV i and c_i is

$$\ddot{r}_i + 2q\dot{r}_i + q^2(r_i - \frac{1}{2}d_i^2) = g_i(r_i, \dot{c}_i, \ddot{c}_i), \tag{19}$$

where g_i combines all the terms that depend on derivatives of c_i:

$$g_i = -2\dot{p}_i^T\dot{c}_i + \dot{c}_i^2 - 2\delta_i^T\ddot{c}_i. \tag{20}$$

For analysis purposes, g_i is treated as a disturbance. Thus (19) describes a critically damped, second order linear system, which is perturbed by g_i. At steady state, the velocities of UGVs are synchronized, implying $\ddot{c}_i = 0$. At steady state, therefore, g_i is bounded as

$$\limsup_{t\to\infty} \|g\| \leq 2\mathrm{V}\,\|\dot{c}_i\| + \|\dot{c}_i\|^2 = 2\mathrm{V}\bar{C} + \bar{C}^2 = \bar{C}\Big(2\mathrm{V} + \bar{C}\Big), \tag{21}$$

where $\bar{C}$ denotes the upper bound of the centroid speed (smaller than the largest UGV initial speed [22]). In state space, (19) reads

$$\dot{z}_{1i} = z_{2i} \tag{22a}$$

$$\dot{z}_{2i} = -2qz_{2i} - q^2 z_{1i} + g_i(z_i, \dot{c}_i, \ddot{c}_i), \tag{22b}$$

where $z_{1i} = r_i - d_i^2$.

Consider the following Lyapunov function candidate

$$W = z_i^T M z_i = \begin{bmatrix} z_{1i} & z_{2i}\end{bmatrix}\begin{bmatrix} m_1 & m_2 \\ m_2 & m_3\end{bmatrix}\begin{bmatrix} z_{1i} \\ z_{2i}\end{bmatrix}, \tag{23}$$

where M is a positive definite matrix. The time derivative of (23) along the flows of (22) is

$$\begin{aligned}\dot{W} = z_{1i}z_{2i}(2m_1 - 4qm_2 - 2q^2m_3) - 2q^2z_{1i}^2 \\ + (2m_2 - 4qm_3)z_{2i}^2 + (2m_2z_{1i} + 2m_3z_{2i})g_i.\end{aligned}$$

If we select $m_1 = \frac{q}{2},\ m_2 = -\frac{1}{4}$ and $m_3 = \frac{1}{q}$ we have

$$\begin{aligned}\dot{W} &= -2q^2 z_{1i}^2 - \frac{9}{2} z_{2i}^2 + \left(-\frac{1}{2} z_{1i} + \frac{2}{q} z_{2i}\right) g_i \\ &\leq -\min\left\{2q^2,\ \tfrac{9}{2}\right\} (z_{1i}^2 + z_{2i}^2) + \max\left\{\tfrac{1}{2},\ \tfrac{2}{q}\right\} (\|z_{1i}\| + \|z_{2i}\|) \max \|g_i\| .\end{aligned}$$

The derivative of the Lyapunov function candidate is further bounded by

$$\begin{aligned}\dot{W} &\leq -\min\left\{2q^2,\ \tfrac{9}{2}\right\} \|z_i\|^2 + \max\left\{\tfrac{1}{2},\ \tfrac{2}{q}\right\} \sqrt{2}\ \|z_i\| \max \|g_i\| \\ &= -\|z_i\| \left(\min\left\{2q^2,\ \tfrac{9}{2}\right\} \|z_i\| - \max\left\{\tfrac{1}{2},\ \tfrac{2}{q}\right\} \sqrt{2} \max \|g_i\|\right).\end{aligned}$$

For $\dot{W} < 0$ we need

$$0 < \min\left\{2q^2,\ \tfrac{9}{2}\right\} \|z_i\| - \max\left\{\tfrac{1}{2},\ \tfrac{2}{q}\right\} \sqrt{2} \max \|g_i\| ,\ \Rightarrow$$

$$\|z_i\| > \frac{\max\left\{\frac{1}{2},\ \frac{2}{q}\right\} \sqrt{2} \max \|g_i\|}{\min\left\{2q^2,\ \frac{9}{2}\right\}}.$$

For sufficiently large q,

$$\|z_i\| > \frac{\sqrt{2} \max \|g_i\|}{9} = \frac{\sqrt{2}}{9} \bar{C}(2\mathrm{V} + \bar{C}). \tag{24}$$

Condition (24) ensures that z_{1i} is ultimately uniformly bounded, which implies that at steady state, UAVs track the UGVs' centroid. The spiral UAV orbits have therefore radii that close to the desired ones within a bounded error, and the slower the UGVs and UAVs move (expressed by $\bar{C}$ and V, respectively), the smaller this error is.

UAVs obtain velocity information from broadcasting UGVs, and they can therefore use it as an estimate for the centroid velocity. Let $\hat{\dot{c}}_i$ denote this centroid velocity estimate that UAV i has. Then the UAV control input (18) can be modified as follows

$$\omega_i = \frac{-\mathrm{V}^2 - q^2 \left(r_i - \frac{1}{2} d_i^2\right) - 2q\dot{r}_i - \hat{\beta}_i}{\mathrm{V} \delta_i^T \eta_i}, \tag{25}$$

where $\hat{\beta}_i \triangleq -2\dot{p}_i^T \hat{\dot{c}}_i + \hat{\dot{c}}_i^2$ is used to denote the estimate of $\beta_i \triangleq -2\dot{p}_i^T \dot{c}_i + \dot{c}_i^2$. By defining $\gamma_i \triangleq -2\delta_i^T \ddot{c}_i$, the closed loop dynamics (19) is rewritten as

$$\ddot{r}_i + 2q\dot{r}_i + q^2(r_i - \frac{1}{2} d_i^2) + (\hat{\beta}_i - \beta_i) - \gamma_i = 0. \tag{26}$$

Being a linear (nominally) stable system, (26) is bounded-input- bounded-output with respect to input $(\hat{\beta}_i - \beta_i) - \gamma_i$ (or ISS if viewed in a nonlinear

context) . Note, in addition that at steady state: $\limsup_{t\to\infty} \|(\hat{\beta}_i - \beta_i) - \gamma_i\| = 0$. Thus, Lemma 4.7 of [26] ensures that $r_i \to \frac{1}{2}\delta_i^2$, which means that the center of every UAV's circular orbit converges to its estimate of the UGV group centroid exponentially.

In discrete time, the stabilizing control law (25) is written

$$\Omega_i(k+1) = \frac{-\mathrm{V}^2 - q^2\Big(r_i(k) - \frac{1}{2}d_i(k)^2\Big) - 2q\left(\frac{r_i(k)-r_i(k-1)}{T}\right) - \hat{\beta}_i(k)}{\mathrm{V}\delta_i(k)^T\eta_i(k)}, \qquad (27)$$

where

$$\hat{\beta}_i(k) = -\frac{2\left(p_i(k) - p_i(k-1)\right)^T\left(\hat{c}_i(k) - \hat{c}_i(k-1)\right)}{T^2} + \left(\frac{\hat{c}_i(k) - \hat{c}_i(k-1)}{T}\right)^2.$$

5.2 Aerial collision avoidance

Our approach is similar in spirit to that of [14] and [15]. If at time step k two or more UAVs in risk of colliding, they modify their angular velocities ω_i at time step $k+1$ as follows

$$\omega_i(k+1) = \omega_{max}\ \mathrm{sign}\left\{\sum_{j:\|\mu_{ij}\|<R_c} \frac{\xi^T\Big(\mu_{ij}(k)\times u_i(k)\Big)}{\|\mu_{ij}(k)\|^2}\right\}, \qquad (28)$$

where $\mu_{ij} = r_i - r_j$ denotes the relative position vector of UAV i with respect to UAV j, and ξ is the unit vector in the vertical direction. Control input (28) differs from the one of [14] in that

- ω_i assumes only two values $-\omega_{max}$, ω_{max} during evasive maneuvers.
- Every UAV within range is taken into consideration, by averaging over all UAVs j within R_c distance to UAV i.
- The cross product in (28) weighs each neighbor's contribution according to proximity.

We note that no formal proof for collision avoidance involving multiple agents is currently available for this type of gyroscopic inputs, except for the analysis in [15] for the two-agent case.

Let us illustrate how (28) performs in the following examples where, for simplicity of representation, only two agents are involved in a near collision event. Near collision configurations between any two agents can be categorized into twelve different classes, according to relative positions and angular velocity. We will only discuss a small subset of these cases here.

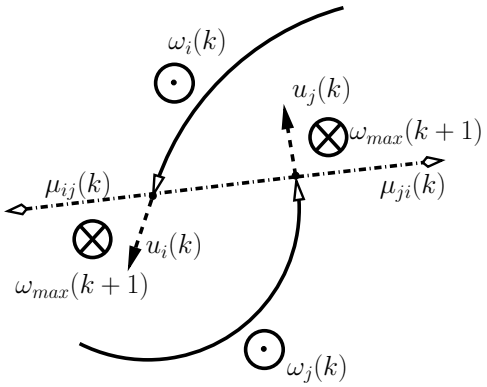

Fig. 2. Adjustment of angular velocities for collision avoidance in the case of two approaching vehicles which have angular velocities along same directions. Symbol $\odot$ denotes directions of angular velocities pointing out of the figure plane, whereas $\otimes$ denotes directions into the figure plane.

Figure 2 shows UAVs i and j approaching each other at time step k while having aligned angular velocities in the same direction. Their distance satisfies $\|\mu_{ij}(k)\| = \|\mu_{ji}(k)\| \leq R_c$. To avoid collision, UAVs adjust their angular velocities to

$$\omega_i(k+1) = \omega_{max}\ \text{sign}\left\{\frac{\xi^T\Big(\mu_{ij}(k) \times u_i(k)\Big)}{\|\mu_{ij}(k)\|^2}\right\} = -\omega_{max},$$

$$\omega_j(k+1) = \omega_{max}\ \text{sign}\left\{\frac{\xi^T\Big(\mu_{ji}(k) \times u_j(k)\Big)}{\|\mu_{ji}(k)\|^2}\right\} = -\omega_{max}.$$

The resulting angular velocities will both have direction into the figure plane, steering the two UAVs away from each other.

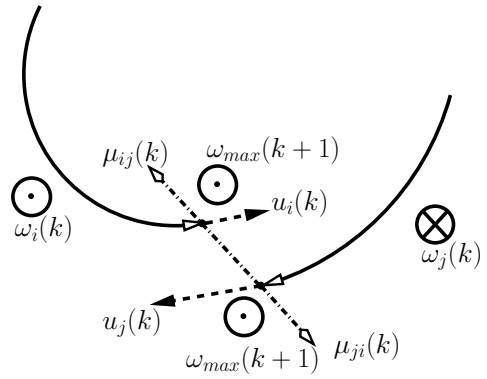

Fig. 3. Adjustment of angular velocities for collision avoidance in the case of two approaching vehicles which have angular velocities along different directions. Symbol $\odot$ denotes directions of angular velocities pointing out of the figure plane, whereas $\otimes$ denotes directions into the figure plane.

Figure 3 shows UAVs i and j approaching each other at time step k with opposite angular velocities. Their distance is $\|\mu_{ij}(k)\| = \|\mu_{ji}(k)\| \leq R_c$. To

avoid collision the UAVs adjust their angular velocities to

$$\omega_i(k+1) = \omega_{max}\ \text{sign}\left\{\frac{\xi^T\Big(\mu_{ij}(k)\times u_i(k)\Big)}{\|\mu_{ij}(k)\|^2}\right\} = \omega_{max},$$
$$\omega_j(k+1) = \omega_{max}\text{sign}\left\{\frac{\xi^T\Big(\mu_{ji}(k)\times u_j(k)\Big)}{\|\mu_{ji}(k)\|^2}\right\} = \omega_{max}.$$

The resulting angular velocities both have directions out of the figure plane, suggesting that the UAVs start to increase the distance that separates them.

Out of the total twelve possible one-on-one collision configurations, there are two where the cross product appearing in (28) is zero. In these cases UAVs assume maximum angular velocities of the same direction, by default.

5.3 Combined control law

The two control law expressions given in (27) and (28) are combined into a switching control scheme. In discrete-time, the input of UAV i is adjusted as

- If $\|\mu_{ij}\| > R_c,\ \forall j = 1,\ldots,M$, and no collision is imminent,

$$\omega_i = \begin{cases}\max\{\Omega_i, -\omega_{max}\} & \text{if } \Omega_i \le 0\\ \min\{\Omega_i, \omega_{max}\} & \text{if } \Omega_i > 0\end{cases} \tag{29}$$

- If $j : \|\mu_{ij}\| < R_c$ and a collision might occur,

$$\omega_i = \omega_{max}\text{sign}\left[\sum_{j:\|\mu_{ij}\|<R_c}\frac{\xi^T(\mu_{ij}\times u_i)}{\|\mu_{ij}\|^2}\right]. \tag{30}$$

Since the execution of (27) can be interrupted by a (finite time of) collision avoidance maneuvers, the exponential stability results of Section 5.1 do not hold, and we settle with uniform ultimate boundedness for the tracking errors.

6 Simulation Results and Discussion

We test the proposed controllers in a number of simulation examples. In the simulation examples, (1) and (10) represent the position and velocity dynamics of UGVs, respectively. In the examples, the UGV group consists of seven vehicles, starting at random initial conditions in a $[-1.7, 1.7]$ m interval for longitudinal and latitudinal position, and $[-1, 1]$ m/s for speed along random

directions. We select a desired inter-vehicle distance $R_d = 1$ m. To improve performance and decrease the risk of collision (recall that ground vehicles use delayed position information), agents estimate the current position of their neighbors using the delayed state, propagating it through the dynamics for the period of the delay. In the first simulation test, the UGV group communicates over a complete communication graph.

UAV dynamics is represented by (29) and (30). The UAV group consists of four vehicles (Figures 4 to 6). UAVs start with random initial conditions in a $[-4.89, 2.89]$ m interval for their position, and $V = 50$ m/s for their speeds with random velocity orientation. For UAVs 1, 2, 3, 4 the desired radius d is set to 0.2432, 0.4865, 0.7297, 0.9730 m respectively. The common sensing range for collision avoidance is $R_c = 1.7513$ m.

Figures 4 through 10 represent an abstract realization of the scenario described in Section 1. Figure 4 shows an overhead snapshot of the initial configuration of the seven-vehicle UGV group communicating over a complete interconnection graph, while four UAVs fly above them. There are no communication links between the UAVs.

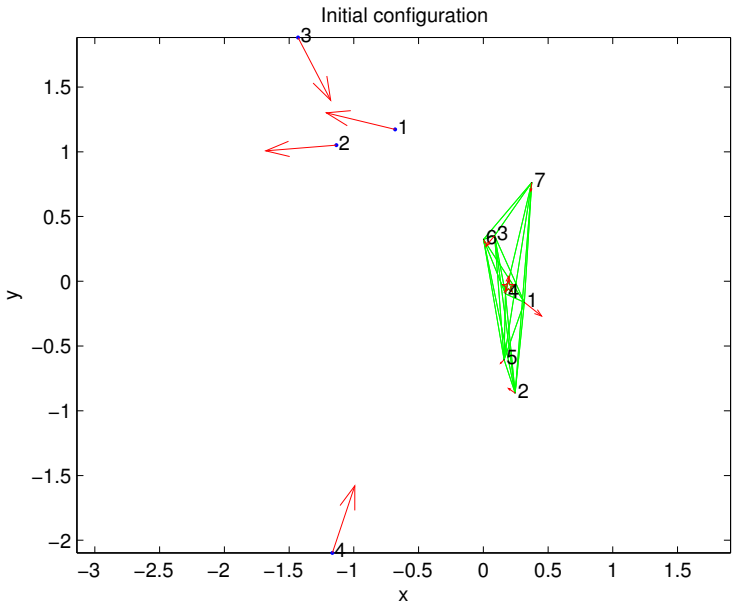

Fig. 4. Initial configurations of UAV and UGV groups. Both vehicles are assumed as point masses. The disconnected nodes and arrows (labeled 1 - 4) denote random initial positions and orientations, respectively, for the UAVs. UGVs 1 through 7, start at random initial positions and velocities in the neighborhood of the coordinate frame origin, and are all connected to each other by communication links, shown in the figure as line segments. The x and y coordinates are in meters.

In Figure 5, we see UAVs 2, 3 and 4 having successfully avoided collision. Their

tracks (shown as dotted curves), reveal that UAVs 2 and 4 have performed collision avoidance maneuvers. Right after UAV 4 avoided UAV 2, it came close to UAV 3, which triggered a subsequent collision avoidance maneuver in the opposite direction on the part of UAV 4.

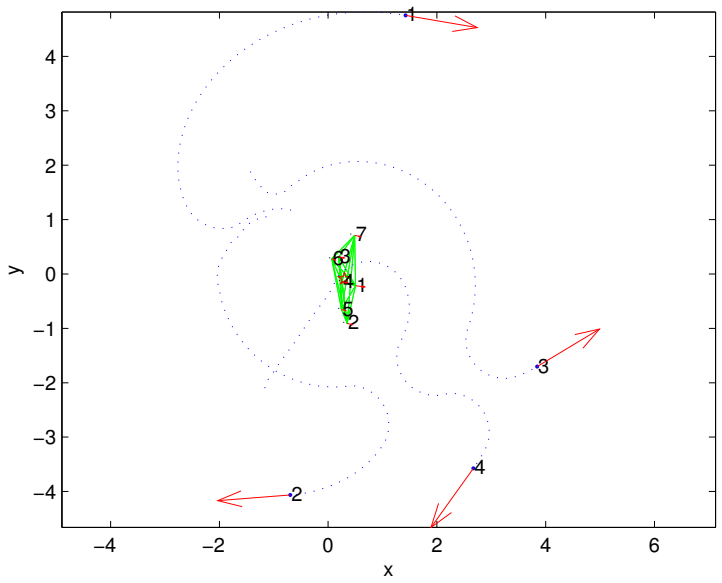

Fig. 5. Collision avoidance maneuvers of UAVs. Vehicle paths are shown as dotted curves. At initial time UAVs 1, 2, and 3 engage in obstacle avoidance maneuvers. Then, after a time period, 2 and 4 avoid each other. Immediately after, UAV 4 needs to avoid collision with UAV 3.

The configuration resulting after executing the simulation test for 3000 time steps is shown in Figure 6. Figure 8 shows that after this time, the UGVs' relative positions are stabilized. From Figure 10 it can be seen that right before the 3000th time step, UAVs 2 and 4 just tried to avoid collision but still came very close, as revealed by the sudden depression in the value of inter-vehicle distance appearing around that time.

By the 3000th time step, the velocities components of the UGVs have converged to common values (Figure 7). Figure 9 shows that interaction forces have converged to zero, indicating that the agents' potentials have been locally minimized.

Figures 11 through 15 illustrate a second simulation scenario, where UGVs are not all connected to each other by means of communication links. Figure 11 shows the initial configuration of a UGV group, where the interconnections between vehicles (communication graph edges) are represented by line segments. The communication graph is connected but not complete. Note the distribution of initial, individual UGV centroid estimates around the actual centroid in the vicinity of point $(-0.13, 0.25)$.

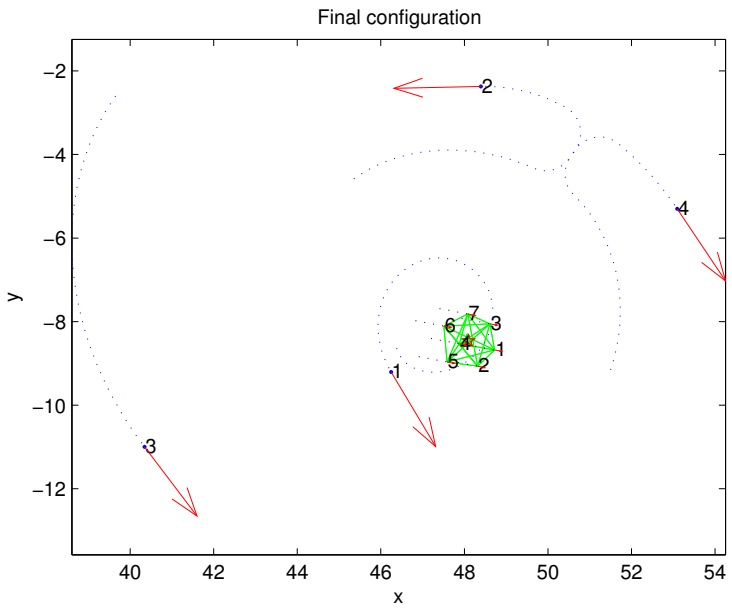

Fig. 6. Configuration of the UGV and UAV groups after 3000 time steps. Ground vehicles have formed a symmetric polygonal formation, and aerial vehicles complete spirals tracking the centroid of the ground group. Before the 3000th time step, UAVs 2 and 4 come close to collision (compare with Figure 10).

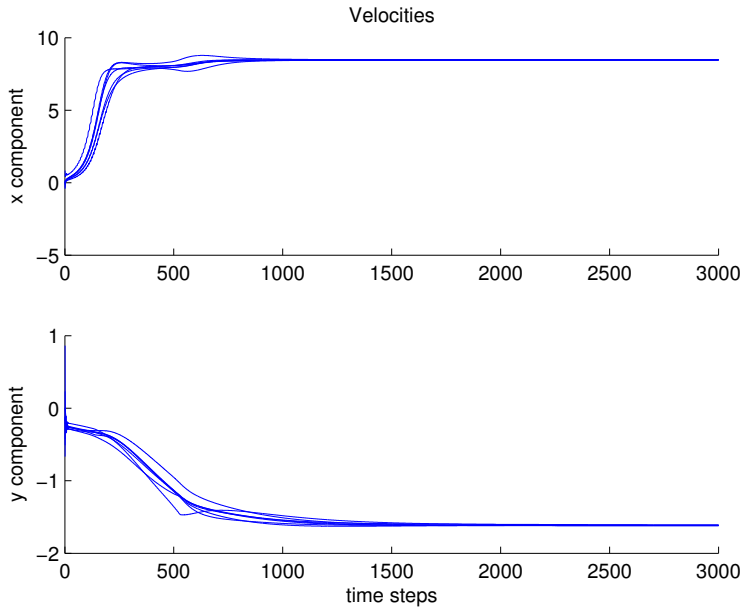

Fig. 7. Evolution of velocity components of the vehicles in the UGV group, along the x and y direction. Different curves correspond to different vehicles. Convergence for both x and y components is achieved after 1500th time step.

A few time steps later, these individual UGV centroid estimates converge (Figure 12).

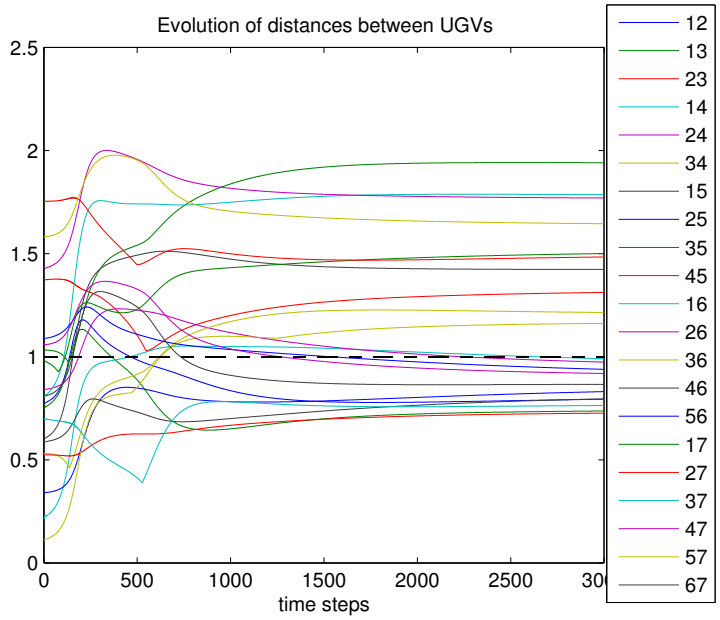

Fig. 8. Evolution of inter-vehicle distances between communicating UGVs. No distance among the UGVs comes close to zero, verifying collision avoidance. The ground group reaches steady state formation with relative positions stabilized.

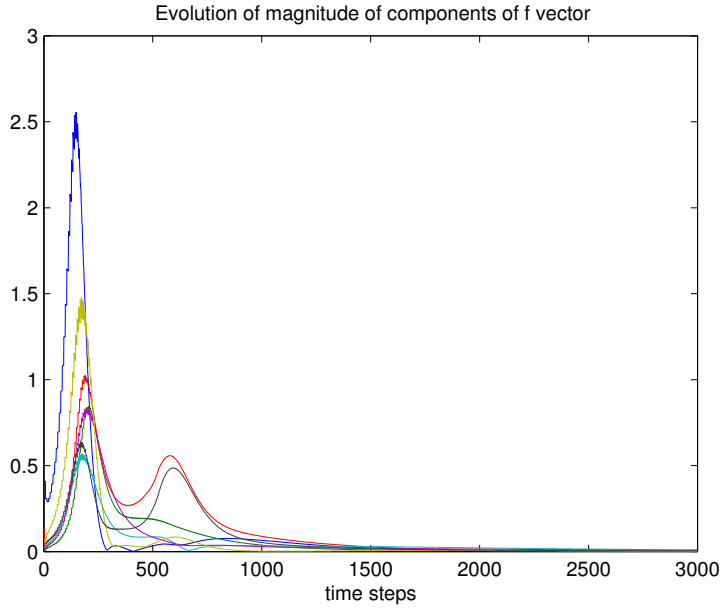

Fig. 9. Evolution of forces between communicating UGVs. Vehicle artificial potentials are locally minimized, causing the associated potential field terms to converge to zero.

The purpose of Figure 13 is to show that under the current scheme where no proximity sensing capabilities are assumed for the UGVs, unless vehicles communicate explicitly they cannot ensure that collision between them is avoided. These vehicles are simply not aware of each other's existence. In Figure 13,

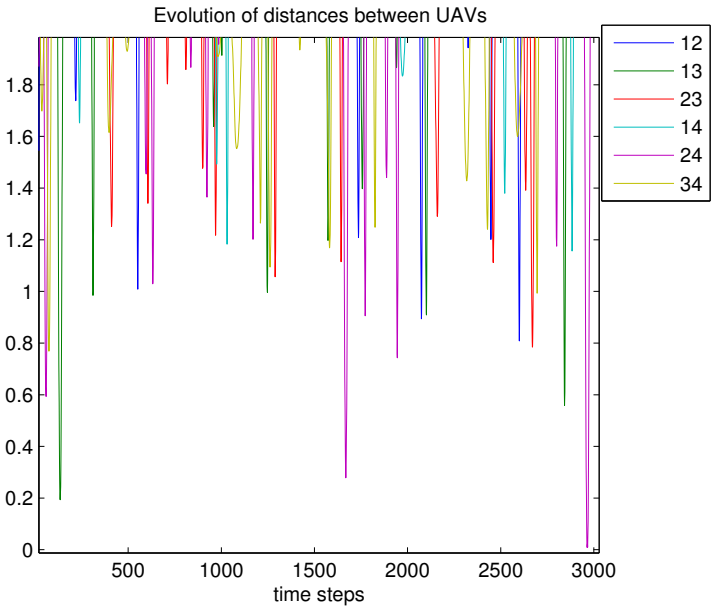

Fig. 10. Evolution of distances between UAVs. Note that the distance between UAVs 2 and 4 comes close to zero just before the 3000th time step. This behavior is indicative of the fact that gyroscopic inputs can not guarantee collision avoidance in a multi-vehicle setting.

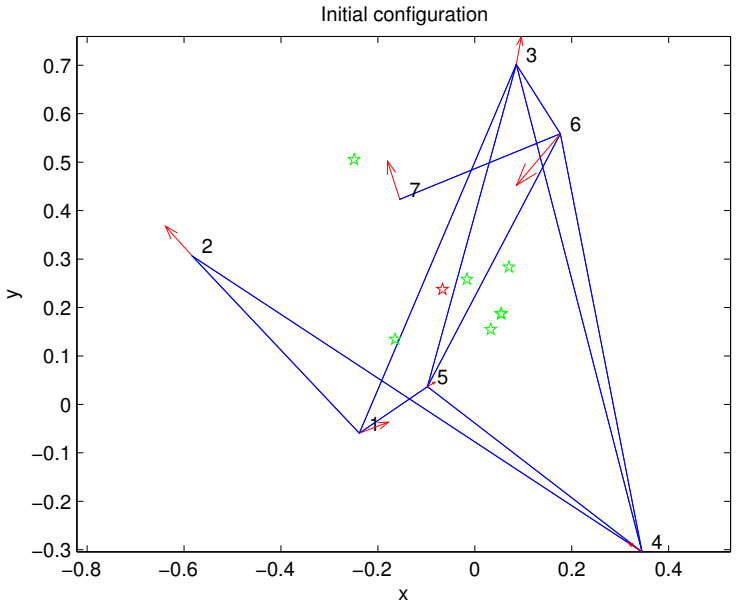

Fig. 11. Initial configuration of the UGV group, in the scenario where the communication graph is not complete. The ★ marker in the vicinity of point $(-0.13, 0.25)$ marks the location of the actual UGV group centroid. The remaining ★ markers show the distribution of the vehicles' initial estimates of this centroid.

UGVs 2 and 7 do not communicate; there is no graph edge incident to both these two nodes. Thus, it is quite possible that they find themselves "dan-

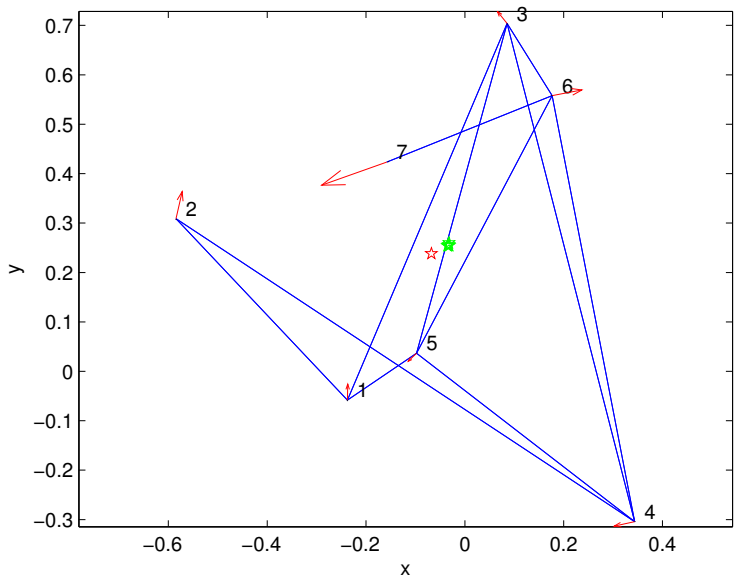

Fig. 12. A few time steps after initialization, individual UGV centroid estimates converge to a common point, indicated by the overlapping ★ markers around $(-0.07, 0.27)$.

gerously" close to each other. Such a phenomenon is possible not only at the steady state but also during the transient period. With the introduction of proximity sensing capabilities, one can consider adjacent nodes in (11) ($i \sim j$) not only those that correspond to communicating vehicles, but also those that are associated with vehicles within a certain distance from each other. This eliminate the phenomenon observed in Figure 13, but introduces time-varying inter-vehicle network dynamics and brings about interesting new problems that are, nevertheless, beyond the scope of this work.

After 3000 time steps, the UGV team is at the steady state configuration of Figure 14. The vehicle centroid estimates are very close to the actual centroid. Figure 15 illustrates how the velocity components of UAVs converge to common values along the x and the y direction.

7 Conclusions

In this paper we present a methodology for coordinating the motion of a group of UGVs and a group of UAVs, by means of decentralized controllers that use partial and delayed state information. The control objectives achieved by these controllers include the synchronization of ground vehicle velocity vectors, regulation of inter-vehicle distances between communicating UGVs, distributed estimation of the ground team's centroid, asymptotic tracking with

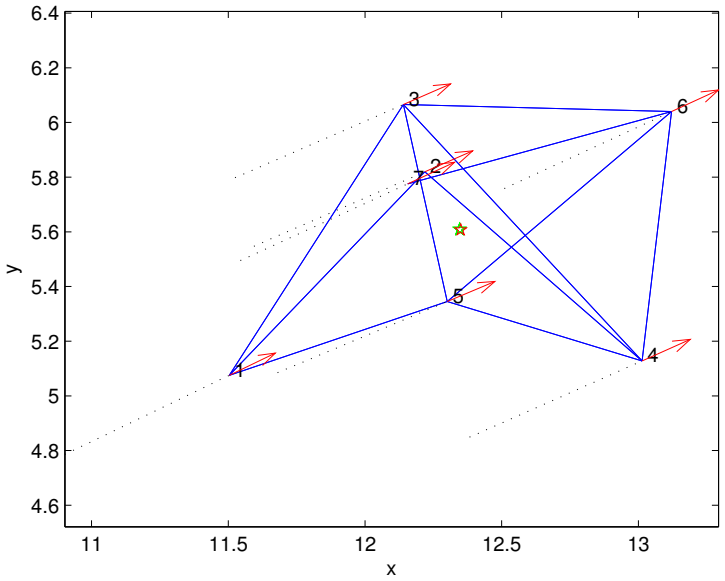

Fig. 13. A snapshot of the configuration of the UGV group in the scenario where the communication graph is not complete, during the transient period. Vehicle velocities are already aligned and the centroid estimates are very close to the actual group centroid near $(12.3, 5.6)$. Vehicles 2 and 7 do not communicate and are not aware of each other's position. They do not generate an inter-vehicle potential force that could maintain a distance between them. .

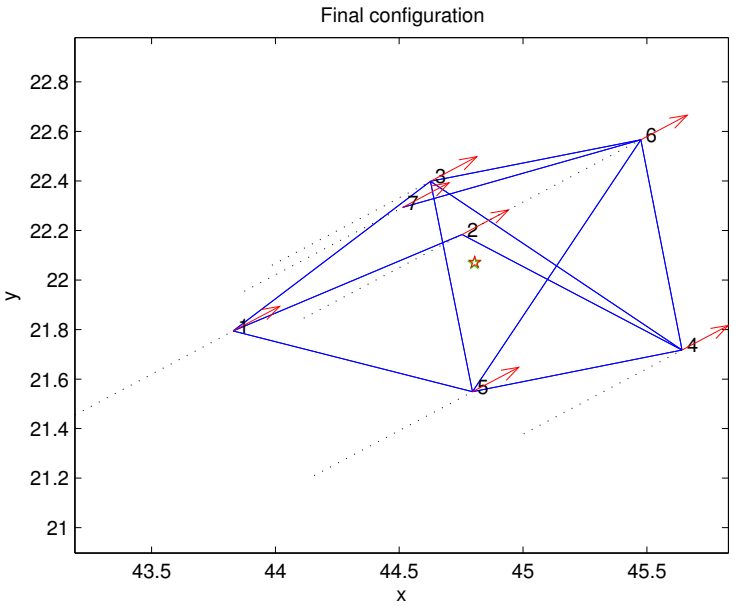

Fig. 14. Steady state configuration of the UGV team in the scenario where the communication graph is not complete. Centroid estimates and actual centroid coordinates (around $(44.7, 22.1)$) are very close. Velocity vectors are synchronized.

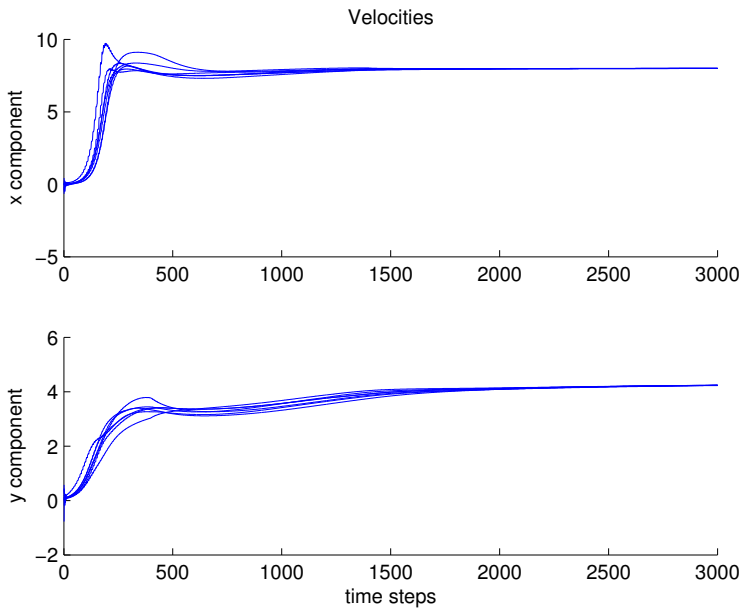

Fig. 15. Convergence of velocity components along the x and y directions for the case of a communication graph that is not complete.

bounded errors of the UGV centroid by aerial vehicles, and establishment of stable orbiting behaviors for UAVs with collision avoidance capabilities. The latter aspect of UAV motion is not guaranteed under the current scheme of gyroscopic-based collision avoidance maneuvering.

Compared to our earlier work [10], in this paper we establish a controlled interaction between heterogeneous groups, we introduce potential-based cohesion and separation forces to regulate inter-vehicle distances, and we establish the stability of the closed loop dynamics, that ensure coordinated motion, for each group individually. A communication protocol which accommodates time delays, inter-vehicle cohesion and separation forces, and a velocity synchronization scheme are combined into a switched control and communication strategy, that is shown to yield asymptotic stability within a Lyapunov framework.

Acknowledgements

The authors would like to thank the Intelligent Systems & Robotics Center of the Sandia National Labs and especially J. Feddema, and K. Groom for their support under SURP grants # 342079 and # 480480.

References

[1] T. Stentz, P. Rander, Integrated air/ground vehicle system for semi-autonomous off-road navigation, in: AUVSI Symposium, Orlando, Florida, 2002.

[2] D. C. Mackenzie, R. C. Arkin, J. M. Cameron, Multiagent Mission Specification and Execution, Kluwer Academic Publishers, Boston, MA, 2003.

[3] M. C. D. Sandia National Laboratories, Cost-effective flexibility; ISRC technologies meet the challenge, Sandia Technology, vol. 2, no. 2, pp 5-11, Summer 2000.

[4] H. G. Tanner, A. Jadbabaie, G. J. Pappas, Stable flocking of mobile agents, Part I: Fixed topology, in: Proceedings of the IEEE Conference on Decision and Control, Maui, Hawaii, 2003, pp. 2010–2015.

[5] Z. Lin, M. Broucke, B. Francis, Local control strategies for groups of mobile autonomous agents, IEEE Transactions on Automatic Control 49 (4) (2004) 622–629.

[6] J. Cortes, S. Martinez, T. Karatas, F. Bullo, Coverage control for mobile sensing networks, IEEE Transactions on Robotics and Automation 20 (2) (2004) 243–255.

[7] Y. Liu, K. M. Passino, Stable social foraging swarms in a noisy environment, IEEE Transactions on Automatic Control 49 (1) (2004) 30–44.

[8] R. Sandoval-Rodriguez, C. T. Abdallah, P. F. Hokayem, Internet-like protocols for the control and coordination of multiple agents with time delay, in: IEEE International Symposium on Intelligent Control, Houston, TX, 2003, pp. 420–425.

[9] R. Olfati-Saber, R. M. Murray, Consensus problems in networks of agents with switching topology and time-delays, in: IEEE Transactions on Automatic Control, Vol. 49, No. 9, 2004, pp. 1520–1533.

[10] H. G. Tanner, D. K. Christodoulakis, State synchronization in local-interaction networks is robust with respect to time delays, in: Proceedings of the 44th IEEE Conference on Decision and Control, Seville, Spain, 2005, pp. 4945–4950.

[11] D. Angeli, P.-A. Bliman, Stability of leaderless multi-agent systems. extension of a result by moreau, math.OC/0411338, (http://arxiv.org/list/math.oc/0411).

[12] A. Morse, Lecture Notes on Logically Switched Dynamical Systems, Springer-Verlag, 2005, (in print).

[13] Y. Liu, K. M. Passino, Cohesive behaviors of multiagent systems with information flow constraints, IEEE Transactions on Automatic Control 51 (11), 1734–1748.

[14] D. E. Chang, S. C. Shadden, J. E. Marsden, R. Olfati-Saber, Collision avoidance for multiple agent systems, in: Proc. IEEE 42st Conf. Decision and Control, Maui, Hawaii, 2003, pp. 539–543.

[15] E. W. Justh, P. S. Krishnaprasad, A simple control law for uav formation flying, technical research report, Institute For Systems Research, University of Maryland, 2002.

[16] R. Rao, V. Kumar, C. Taylor, Visual servoing of a ugv from a uav using differential flatness, in: Intl. Conference on Intelligent Robots and Systems, Las Vegas, Nevada, 2003, pp. 743–748.

[17] Y. Watanabe, E. N. Johnson, A. J. Calise, Vision-based approach to obstacle avoidance, in: AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco, California, 2005, pp. AIAA–2005–6092(1–10).

[18] J. A. Miller, P. D. Minear, J. Albert F. Niessner, A. M. DeLullo, B. R. Geiger, L. N. Long, J. F. Horn, Intelligent unmanned air vehicle flight systems, in: AIAA Infotech@Aerospace Conference, Arlington, VA, 2005, pp. AIAA–2005–7081.

[19] J. B. Saunders, B. Call, A. Curtis, R. W. Beard, T. W. McLain, Static and dynamic obstacle avoidance in miniature air vehicles, in: AIAA Infotech@Aerospace, Arlington, VA, 2005, pp. AIAA–2005–6950(1–14).

[20] S. Todorovic, M. C. Nechyba, A vision system for intelligent mission profiles of micro air vehicles, in: IEEE Transactions on Vehicular Technology, Vol. 53, No. 6, 2004, pp. 1713–1725.

[21] S. Bayraktar, G. E. Fainekos, G. J. Pappas, Hybrid modelling and experimental cooperative control of multiple unmanned aerial vehicles, Tech. Rep. MS-CIS-04-32, University of Pennsylvania (December 2004).

[22] A. Jadbabaie, J. Lin, A. S. Morse, Coordination of mobile autonomous agents using nearest neighbor rules, IEEE Transactions on Automatic Control (2004) 988–1001.

[23] C. Godsil, G. Royle, Algebraic Graph Theory, Springer-Verlag, New York, NY, 2001.

[24] J. A. Marshall, M. E. Broucke, B. A. Francis, Formations of vehicles in cyclic pursuit, IEEE Transactions on Automatic Control 49 (11) (2004) 1963–1974.

[25] R. A. Horn, C. R. Johnson, Matrix Analysis, Cambridge University Press, Cambridge, UK, 1999.

[26] H. K. Khalil, Nonlinear systems 3rd edition, Prentice Hall, Englewood Cliffs, NJ, 2003.