2014 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS 2014)
September 14-18, 2014, Chicago, IL, USA

A Cooperative Formation Control Strategy Maintaining Connectivity of A Multi-agent System

Rajdeep Dutta[1], Liang Sun[1], Mangal Kothari[2], Rajnikant Sharma[3] and Daniel Pack[1]

***Abstract*— In this paper, we present a cooperative target-centric formation control strategy that maintains the dynamic graph connectivity for a system of unmanned aerial vehicles. The connectivity of a graph plays a critical role in dynamic networks of multiple agents since it represents a level of information sharing capability of a system. The connectivity of a network of unmanned systems changes as the state dependent graph evolves over time, revealing the risk of the system being uncontrollable during null connectivity. We show that the existing formation controller fails to make the formation when the network connectivity is lost. We propose a novel formation control strategy which keeps the multi-agent graph connected throughout the system dynamic process. The convergence of the formation controller is shown with the help of the Lyapunov theory. Simulation results validate the effectiveness of the proposed control law.**

I. Introduction

Cooperative control strategies [1]–[5] offer efficient ways to explore various capabilities of a multi-agent system to accomplish a common task, such as surveillance, searching or target tracking. In recent years, there has been considerable research [3]–[5], [8] for multi-agent target tracking. The formation control strategies [4] play a significant role in tracking a smart hostile target by a team of unmanned aerial vehicles. Recently, Sharma et al. [1], [2] proposed a control law for multiple UAVs to make a formation around a moving target. The proposed control law was able to maintain a formation even with incomplete target information. But, the earlier works assumed a network consisting of all the agents (the target and the UAVs) is connected, represented as having a graph with at least one spanning tree; an assumption that is not typically met in reality.

Previous studies [10], [12]–[14] show extensive research work performed in controlling connectivity of a graph. The problem of enhancing robustness of a transportation network by optimizing the weighted algebraic connectivity was addressed in [11]. A decentralized algorithm has been developed in [10] to maximize the algebraic connectivity of a proximity graph consisting of multiple agents. Zavlanos et al. [12] proposed centralized and distributed algorithms to maintain, increase and control connectivity of mobile robot networks by working with a potential field. Authors in [13], [14] made significant contributions toward finding methods to preserve, control, and optimize connectivity of dynamic graphs.

A network of UAVs and targets (multiple agents) can be described as a graph, where agents are the nodes of the graph and an edge between two nodes represents an ability to communicate. In a dynamic network, all nodes move and the distances between them vary with time. A graph edge between two nodes forms or breaks depending on the distance between them since the communication becomes weaker as the relative distance increases and diminishes to null beyond a limit [10]. Such multi-agent system corresponds to a time-varying information exchange topology, and the related graph involves fixed number of nodes but variable number of edges between them. A connected graph implies a network where every two nodes can exchange information through an existing path in that graph. A graph can be analyzed mathematically by investigating the Laplacian matrix [9] associated with it. The graph connectivity is related with the second smallest eigenvalue of its Laplacian matrix, which is also known as the algebraic connectivity [9]. The algebraic connectivity quantifies the strength of connection among the agents in a multi-agent network. The greater its value is above zero, the stronger is the connection among agents.

In recent years, several researchers [1], [3], [5] investigated the problem of designing formation control laws for connected networks. These control laws work effectively as long as the graph is connected [1], [5]. However, they do not take care of the connectivity of the time-varying graph, nor guarantee to maintain the connectivity during an entire mission. The existing control laws are inadequate if the graph breaks during the dynamic process.

In this paper, we present a control strategy which accounts for the connectivity in cooperation with making formation. In Figure 1, we show a multi-agent time varying graph $G(t)$ consisting of four UAVs and one target at time t. Figure 1 describes how the communication links between the agents can change with time, and also reveals the risk of one agent being disconnected from the mobile network. The present work focuses on finding a control strategy which maintains the formation of UAVs around a target and the dynamic graph connectivity. Instead of assuming a constant information topology, in this paper, we consider a state-dependent weighted undirected graph to represent the time-varying information exchange topology among agents.

The rest of the paper is organized as follows. In Section II, we brief the related graph theory basics, and formulate the problem by defining the control objective. Section III states the existing formation controller, and presents the proposed

[1] Department of Electrical Engineering, University of Texas at San Antonio, San Antonio, TX 78249, USA rajdeepdutta.iisc@gmail.com
[2] Department of Aerospace Engineering, Indian Institute of Technology Kanpur, Kanpur 208016, India mangal@iitk.ac.in
[3] Department of Electrical Engineering, Utah State University, Logan, UT 84322, USA raj.drdo@gmail.com

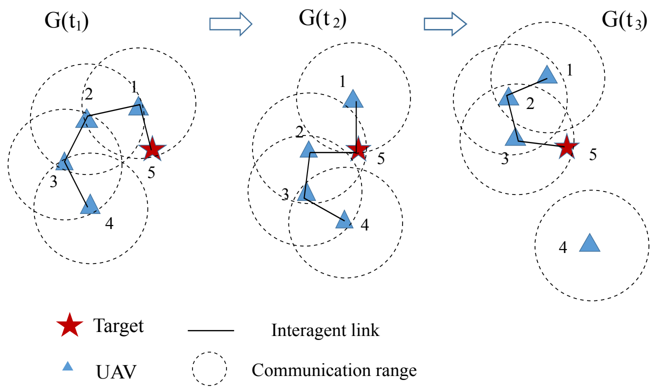


Fig. 1. A time varying multi-agent graph of four UAVs and one target, showing the risk of losing one agent from the network.

controller. In section IV, we present the simulation results and show the new controller's contributions over the earlier work. Finally, Section V concludes the paper.

II. Background and Problem Formulation

A. Graph Representation

A multi-agent system of UAVs and target, as shown in Figure 1, can be represented by a time-varying graph $G(t) = (V, E(t))$, where V is the set of all nodes or agents and $E(t)$ is the set of all edges between the nodes at time t. An edge is a connection between a distinct pair of nodes (i, j) of V. This connection exists if the corresponding relative distance between two agents is at most equal to the communication range. We assume that a node i is connected to all other nodes j which are in a circular neighborhood of i defined by the communication radius, R. The time-varying neighborhood of the ith agent can be expressed mathematically as follows.

$$N_i(t) = \{j \in V, j \neq i : \|\mathbf{r}_{ij}\| = \| \mathbf{p}_i - \mathbf{p}_j \| \leq R\} \tag{1}$$

where, $\mathbf{p_i}$ and $\mathbf{p}_j$ denote the position vectors of the ith and the jth nodes, respectively, and $\mathbf{r}_{ij}$ denotes the relative distance vector between the ith and jth nodes. It is to be noted that all agents have the same communication range, and the communication links are bidirectional, which in turn means that the graph $G(t)$ is undirected. In our case, the graph consisting of n UAVs and one target is represented by $G_{n+1}(t)$, which has $n + 1$ nodes and a variable number of edges between them.

B. Associated Matrices

A graph can be represented algebraically by the associated Laplacian matrix. The order of the Laplacian matrix is identical to the number of nodes present in the graph. In order to incorporate the notion of varying distance into the Laplacian, we consider the Laplacian matrix to be weighted [11]. The elements of the matrix are related with the corresponding graph edge weights, depending on the relative distance between corresponding pair of nodes. A state dependent Laplacian matrix [10] can be expressed as

$$L(t) = D(\mathbf{x}(t)) - A(\mathbf{x}(t)) \ , \tag{2}$$

where $\mathbf{x}(t)$ represents the state vector, $D(\mathbf{x})$ is the Degree matrix, and $A(\mathbf{x})$ is the Adjacency matrix. The elements of a weighted Adjacency matrix (a_{ij}) can be given by [10]

$$a_{ij} = e^{-\sigma \|\mathbf{r}_{ij}\|} \tag{3}$$

where σ is a positive constant representing the slope of communication quality over distance, which depends on the communication range R, and the effective antenna gains [16]. The Degree matrix is a diagonal matrix with elements

$$d_i = \sum_{j(\neq i)=1}^{n+1} a_{ij} \ .$$

The Laplacian matrix (L_{n+1}) elements are given by

$$l_{ij} = \begin{cases} -a_{ij} \text{ for } i \neq j; \\ \sum\limits_{j(\neq i)=1}^{n+1} a_{ij} \text{ for } i = j. \end{cases} \tag{4}$$

The Laplacian matrix associated with an undirected graph is positive semi-definite and symmetric. The smallest eigenvalue of the matrix λ_1 equals to zero, and the associated eigenvector is $\mathbf{1}$. The second smallest eigenvalue, $\lambda_2 \geq 0$, is called the *algebraic connectivity* of the graph, and the associated eigenvector is known as the Fiedler vector [9]. A higher positive value of λ_2 indicates a stronger connected graph, and $\lambda_2 = 0$ implies that the graph is not connected since there exists no spanning tree in the graph. As the graph $G_{n+1}(t)$ is state dependent and changes in accordance with the dynamics of agents, the connectivity of the graph $\lambda_2(t)$ is also state dependent.

C. Problem Statement and Objective

Consider a multi-agent network composed of n UAVs and one target. The order of the multi-agent system is $n+1$. Each UAV can exchange information about position, orientation, and control inputs with its neighbors, although the target does not receive information from UAVs. We assume there is no direct communication between the target and UAVs, however, the UAVs are able to estimate the target states. It is also assumed that all UAVs fly at a constant altitude.
For the ith UAV, where $i \in [1, n]$, let $\mathbf{p}_i(t) = [x_i(t),\ y_i(t)]^T$ denote the position at time t, v_i denote the velocity, ϕ_i denote the heading angle, and $\mathbf{u}_i$ denote the control vector. Then, the motion of the ith UAV is governed by

$$\dot{\mathbf{p}}_i = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \begin{bmatrix} v_i \cos\phi_i \\ v_i \sin\phi_i \end{bmatrix}$$

$$\ddot{\mathbf{p}}_i = \begin{bmatrix} \cos\phi_i & -v_i \sin\phi_i \\ \sin\phi_i & -v_i \cos\phi_i \end{bmatrix} \begin{bmatrix} \dot{v}_i \\ \dot{\phi}_i \end{bmatrix}$$

$$\ddot{\mathbf{p}}_i = M_i \mathbf{u}_i, \quad \mathbf{u}_i = \begin{bmatrix} \dot{v}_i \\ \dot{\phi}_i \end{bmatrix}, \quad M_i = \begin{bmatrix} \cos\phi_i & -v_i \sin\phi_i \\ \sin\phi_i & -v_i \cos\phi_i \end{bmatrix}.$$

The objective of a target-centric formation [1], [2] is to drive the UAVs to positions at a constant distance δ and angle ψ_i with respect to the target. Let the position and the velocity of the target be denoted by $\mathbf{r}_t = \mathbf{r}_{n+1}$ and $v_t = v_{n+1}$, respectively. Thus, the control objective can be expressed mathematically as follows

$$\mathbf{p}_i(t) - \mathbf{p}_t(t) \to \mathbf{P}_i \tag{5}$$
$$\dot{\mathbf{p}}_i(t) - \dot{\mathbf{p}}_t(t) \to 0 \tag{6}$$
$$\psi_{i+1} - \psi_i = \frac{2\pi}{n} \tag{7}$$
$$\lambda_2(L(t)) > 0 \ , \tag{8}$$

where $\mathbf{P}_i = \delta[\cos\psi_i \quad \sin\psi_i]^T$ for some pre-defined δ. Note that the UAV velocities of agents are bounded by $v_{min} \leq v_i \leq v_{max}$. Figure 2 shows a typical scenario of the desired formation of four UAVs around one target.

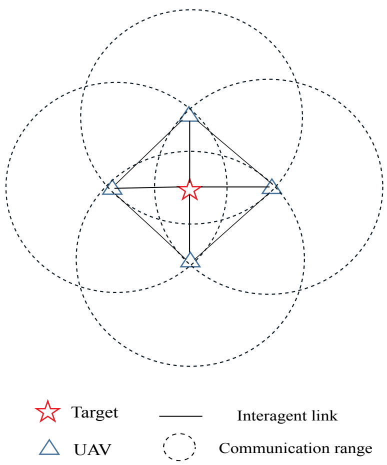


Fig. 2. A scenario: Desired formation for four UAVs around one target.

III. Formation Controller Development

A. Target-centric Formation Controller

The existing formation controller with complete target information [1], [2] is described below. The control law [1] for the ith agent to drive the relative position vector $\mathbf{p}_i - \mathbf{p}_j$ to the desired separation, $\mathbf{P}_i - \mathbf{P}_j$, is given by

$$M_i\mathbf{u}_i = M_j\mathbf{u}_j - \alpha_1(\hat{\mathbf{p}}_i - \hat{\mathbf{p}}_j) - \alpha_2(\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j), \quad i \neq j \tag{9}$$

where α_1 and α_2 are positive constants, and $\hat{\mathbf{p}}_i = \mathbf{p}_i - \mathbf{P}_i$. The control input $\mathbf{u}_i$ for the ith UAV $i = 1, 2, ..., n$ is as follows.

$$\mathbf{u}_i = \frac{M_i^{-1}}{\sum\limits_{j\in N_i} a_{ij}} \sum_{j\in N_i} a_{ij}[\ddot{\mathbf{p}}_j - \alpha_1(\hat{\mathbf{p}}_i - \hat{\mathbf{p}}_j) - \alpha_2(\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j)] \tag{10}$$

In previous works [1], [2], a_{ij} was selected as binary value (either 1 or 0) since the graph G_{n+1} was constant over time. In this work, we consider a dynamic graph $G_{n+1}(t)$ and time varying parameters a_{ij}. Equation (10) can be expressed in a compact form as

$$[L_{n+1} \otimes I_m]\ddot{\mathbf{p}} = -\alpha_1[L_{n+1} \otimes I_m]\hat{\mathbf{p}} - \alpha_2[L_{n+1} \otimes I_m]\dot{\mathbf{p}}. \tag{11}$$

where $m = \dim(\mathbf{p}_i) = 2$, $\otimes$ is the Kronecker product, $\mathbf{p} = [\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_{n+1}]^T \in \Re^{(n+1)m}$, $\mathbf{P} = [\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_{n+1}]^T \in \Re^{(n+1)m}$, and $\hat{\mathbf{p}} = \mathbf{p} - \mathbf{P} \in \Re^{(n+1)m}$. The final control law developed by the earlier work [1] is shown in the following.

$$[\ddot{\mathbf{p}} - (\mathbf{1} \otimes \ddot{\mathbf{p}}_t)] = -\alpha_1[\hat{\mathbf{p}} - (\mathbf{1} \otimes \mathbf{p}_t)] - \alpha_2[\dot{\mathbf{p}} - (\mathbf{1} \otimes \dot{\mathbf{p}}_t)] \tag{12}$$

B. An Upgraded Strategy Concerning Connectivity

It can be inferred from Equation (3) that the controller weighting factor a_{ij} becomes smaller as the relative distance increases. This implies that the existing controller [2], as shown in Equation (10), puts less control effort to the distant nodes of graph if a_{ij} are used as controller weights for time-varying network. In other words, the connectivity of the entire network fails. We redefine controller weights to overcome this drawback, and propose new controller weights as

$$b_{ij} = \gamma - a_{ij} \tag{13}$$

where γ is a positive constant satisfying $\gamma \geq 1$. Note that $\gamma \geq a_{ij}$ since $0 \leq a_{ij} \leq 1$, which makes $b_{ij} \geq 0$. In equation (13), a larger value of γ would make b_{ij} to be less influenced by the time-varying term, a_{ij}. In our case, we choose $\gamma = 1$ in order to make the controller weights b_{ij} vary between 0 and 1. Next, we design a new matrix W, associated with these weights, whose elements are given by

$$w_{ij} = \begin{cases} -b_{ij} \text{ for } i \neq j; \\ \sum\limits_{j(\neq i)=1}^{n+1} b_{ij} \text{ for } i = j. \end{cases} \tag{14}$$

The weight matrix W has a Laplacian matrix structure with positive b_{ij}, and so it is symmetric and positive semi-definite. The value of an element in W increases as the relative distance between the corresponding nodes increases so that the control efforts are enhanced for nodes that are moving apart. This modification is carried out to increase the time varying graph connectivity. The weight matrix W can be expressed as

$$W(t) = \gamma C - L(t) \ , \tag{15}$$

where C is a constant Laplacian matrix having all off-diagonal elements as ones. By differentiating Equation (15) w.r.t. time, we get

$$\dot{W}(t) = -\dot{L}(t) \ . \tag{16}$$

Thus, the modified control law $\mathbf{u}_i$ for $i = 1, 2, ..., n$, proposed in this paper is as follows.

$$\mathbf{u}_i = \frac{M_i^{-1}}{\sum\limits_{j\in N_i} w_{ij}} \sum_{j\in N_i} w_{ij}[\ddot{\mathbf{p}}_j - \alpha_1(\hat{\mathbf{p}}_i - \hat{\mathbf{p}}_j) - \alpha_2(\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j)] \tag{17}$$

Equation (17) reveals the fact that the control input for the agent i depends on the state information related to its own and neighbors $j \in N_i$ dynamics. In the following, we show the convergence of the modified controller with the help of the Lyapunov theory [17].

Theorem 1: The control law (17) drives n UAVs to a target-centric formation such that $\mathbf{p}_i - \mathbf{p}_t \to \mathbf{P}_i$ and $\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_t \to$

$\mathbf{0}$, and also guarantees $\lambda_2(t) > 0$ for all $t > 0$, based on a sufficient condition relating the initial and final network connectivity values with the controller parameters.

Proof: *Part 1, Lyapunov convergence:* Define a Lyapunov candidate function as

$$V \triangleq \frac{1}{2}\mathbf{e}^T W_* \mathbf{e} \ , \tag{18}$$

where $\mathbf{e} \triangleq [\hat{\mathbf{p}} - (\mathbf{1} \otimes \mathbf{p}_t)] \in \Re^{(n+1)m}$, and $W_* \triangleq (W_{n+1} \otimes I_m)$. It is to be noted that the last m entries of the vector $\mathbf{e}$ are zeros. The null vector of W_* contains all 1s as its elements. Hence, $\mathbf{e}$ does not lie in the null space of W_*, which implies that V is positive for all $\mathbf{e} \neq \mathbf{0}$ although W_* is positive semi-definite.

Next, we differentiate Equation (18) w.r.t. time and get

$$\dot{V} = \mathbf{e}^T W_* \dot{\mathbf{e}} + \frac{1}{2}\mathbf{e}^T \dot{W}_* \mathbf{e} \ . \tag{19}$$

Using backstepping technique [17], we define a new variable $\mathbf{z}$ as

$$\mathbf{z} \triangleq -\dot{\mathbf{e}} - k_1 \mathbf{e} \ . \tag{20}$$

An augmented Lyapunov candidate function is defined as

$$V_c \triangleq V + \frac{1}{2}\mathbf{z}^T W_* \mathbf{z} \ . \tag{21}$$

Taking time derivatives of the both sides of Equation (21), we get

$$\dot{V}_c = \dot{V} + \mathbf{z}^T W_* \dot{\mathbf{z}} + \frac{1}{2}\mathbf{z}^T \dot{W}_* \mathbf{z} \ . \tag{22}$$

Now, we substitute for $\dot{V}$ in Equation (22) using (19) and (20), and obtain

$$\dot{V}_c = -k_1 \mathbf{e}^T W_* \mathbf{e} + \mathbf{z}^T W_*(\dot{\mathbf{z}} - \mathbf{e}) + \frac{1}{2}(\mathbf{e}^T \dot{W}_* \mathbf{e} + \mathbf{z}^T \dot{W}_* \mathbf{z}) \ . \tag{23}$$

Using Equations (20) and (12), we get

$$\begin{aligned}
\mathbf{z}^T W_*(\dot{\mathbf{z}} - \mathbf{e}) &= \mathbf{z}^T W_*(-\ddot{\mathbf{e}} - k_1\dot{\mathbf{e}} - \mathbf{e}) \\
&= \mathbf{z}^T W_*(\alpha_1 \mathbf{e} + \alpha_2 \dot{\mathbf{e}} - k_1 \dot{\mathbf{e}} - \mathbf{e}) \\
&= \mathbf{z}^T W_*[(\alpha_1 - 1)\mathbf{e} + (\alpha_2 - k_1)\dot{\mathbf{e}}] \ .
\end{aligned}$$

By selecting $\alpha_1 = 1 + k_1 k_2$ and $\alpha_2 = k_1 + k_2$, we have $\mathbf{z}^T W_*(\dot{\mathbf{z}} - \mathbf{e}) = -k_2 \mathbf{z}^T W_* \mathbf{z}$. Then, Equation (23) becomes

$$\dot{V}_c = -k_1 \mathbf{e}^T W_* \mathbf{e} - k_2 \mathbf{z}^T W_* \mathbf{z} + \frac{1}{2}(\mathbf{e}^T \dot{W}_* \mathbf{e} + \mathbf{z}^T \dot{W}_* \mathbf{z}) \tag{24}$$

$$\dot{V}_c = -K\mathbf{y}^T(W_* \otimes I_2)\mathbf{y} + \frac{1}{2}\mathbf{y}^T(\dot{W}_* \otimes I_2)\mathbf{y} \tag{25}$$

$$\text{or, } \dot{V}_c = \mathbf{y}^T[-K(W_* \otimes I_2) + \frac{1}{2}\mathbf{y}^T(\dot{W}_* \otimes I_2)]\mathbf{y}, \tag{26}$$

where $K \triangleq \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}$, $\mathbf{y} \triangleq \begin{bmatrix} \mathbf{e} \\ \mathbf{z} \end{bmatrix}$, and k_1 and k_2 are constants related to the controller gain parameters. Equation (26) reveals that $\dot{V}_c$ will be negative-definite when the following condition is satisfied.

$$\begin{aligned}
& -K(W_* \otimes I_2) + \frac{1}{2}(\dot{W}_* \otimes I_2) < 0 \\
& \text{or, } (\dot{W}_* \otimes I_2) < 2K(W_* \otimes I_2) \\
& \text{or, } (\dot{W} \otimes I_4) < 2K(W \otimes I_4)
\end{aligned}$$

$$\text{or, } \dot{W} < hW \ , \tag{27}$$

where $h = 2 \cdot \min(k_1, k_2)$ is a controller parameter.

Using Equations (15) and (16), the inequality (27) is equivalent to

$$\dot{L}(t) > hL(t) - h\gamma C \ . \tag{28}$$

The time derivative of λ_2 can be expressed as [15]

$$\dot{\lambda_2} = v_2^T \dot{L} v_2 \ . \tag{29}$$

where v_2 is the Fiedler vector, i.e. the eigenvector of L corresponding to the eigenvalue λ_2. Now, we apply the necessary matrix algebra to inequality (28) using (29), and get

$$\dot{\lambda_2}(t) > h\lambda_2(t) - h\gamma c, \tag{30}$$

where, $c = v_2^T C v_2 = \sum_{i\sim j} C(i,j)(v_2(i) - v_2(j))^2$.

As all off-diagonal elements of the constant Laplacian matrix are ones, $C(i,j) = 1, \ i \neq j$, we obtain

$$c = \sum_{i\sim j}(v_2(i) - v_2(j))^2 \ = \ 3\sum_i v_2^2(i) - \sum_i v_2(i). \tag{31}$$

We use the fact that the Fiedler vector v_2 is normalized, i.e. $\sum_i v_2^2(i) = 1$, and the summation of all its elements equals to zero, i.e. $\sum_i v_2(i) = 0$. We now evaluate c using Equation (31), which is $c = 3$. Thus, the inequality (30) becomes

$$\dot{\lambda_2} > h(\lambda_2 - 3\gamma) \ . \tag{32}$$

Thus, $\dot{V}_c$ becomes negative-definite if Inequality (32) is satisfied. Then, the equilibrium point $\mathbf{y} = [\mathbf{e}, \mathbf{z}]^T$ is asymptotically stable and converges to the origin, which in turn means $\mathbf{e} \to \mathbf{0} \ \Rightarrow \ \hat{\mathbf{p}}_i \to \mathbf{p}_t$, or, $\mathbf{p}_i - \mathbf{p}_t \to \mathbf{P}_i$, and $\mathbf{z} \to \mathbf{0} \ \Rightarrow \ \dot{\mathbf{p}}_i \to \dot{\mathbf{p}}_t$.

Part 2, Connectivity Maintenance: Here, we determine a sufficient condition from inequality (30), under which the controller maintains connectivity of the network for all $t > 0$. We obtain the following differential equation from the inequality (30).

$$\dot{\lambda_2}(t) = h(\lambda_2(t) - 3\gamma) + \kappa \ . \tag{33}$$

where κ is a positive constant. By solving the above differential Equation for $\lambda_2(t)$, we get

$$\lambda_2(t) = \exp^{ht}\lambda_2(0) + \int_0^t \exp^{h(t-\tau)}(-3h\gamma + \kappa)d\tau \tag{34}$$

Now, for the formation control mission, we know that $\lim_{t\to\infty} \dot{\lambda_2}(t) = 0$ and $\lim_{t\to\infty} \lambda_2(t) = \lambda_2^{fin}$ (desired formation connectivity). After applying these boundary conditions into equation (33), we get $\kappa = h(3\gamma - \lambda_2^{fin})$. Note that $\lambda_2^{fin} < 3\gamma$ since both h and κ are positive. Then, we substitute κ into equation (33), and obtain the following differential equation.

$$\dot{\lambda_2}(t) = h(\lambda_2(t) - \lambda_2^{fin}) \ . \tag{35}$$

The solution of the above differential equation (35) can be expressed as follows.

$$\lambda_2(t) = \exp^{ht}\lambda_2(0) - h\lambda_2^{fin}\int_0^t \exp^{h(t-\tau)}\,d\tau$$
$$= \exp^{ht}\lambda_2(0) + \lambda_2^{fin}(1-\exp^{ht})$$

$$\text{or, } \lambda_2(t) = \exp^{ht}\{\lambda_2(0) + \lambda_2^{fin}(\exp^{-ht}-1)\}\ . \quad (36)$$

It is evident from the above solution (36) that $\lambda_2(t) > 0$ for all $t > 0$ whenever $\{\lambda_2(0) + \lambda_2^{fin}(\exp^{-ht}-1)\} > 0$ is satisfied, which leads to the following condition.

$$\exp^{-ht} > 1 - \frac{\lambda_2(0)}{\lambda_2^{fin}}\ , \quad (37)$$

where $\lambda_2(0)$ and λ_2^{fin} are the initial and final network connectivity measures, and h is the controller parameter. Note that the above condition (37) is restricted by the case where initial connectivity is higher than the final connectivity.

IV. Numerical Results

In this section, we present simulation results to demonstrate the effectiveness of the proposed cooperative control strategy by showing a comparison performance study of the proposed control law and the one reported in [1], [2]. We consider a group of four UAVs trying to make a formation around one target. The parameters used in the simulation are:

- Desired distance from a UAV to the target: $\delta = 100$m ;
- Desired angle of a UAV w.r.t. the target: $\psi_i = \frac{2\pi i}{3}$;
- UAV velocity bounds: $v_{min} = 5$ m/s, and $v_{max} = 25$ m/s ;
- Target velocity: $v_t = 10$ m/s, and initial heading: $\psi_t = 0$ rad ;
- Controller parameters: $\gamma = 1$, $\alpha_1 = 1$ and $\alpha_2 = 1.4$.

The slope σ of the exponential communication model is parameterized using the communication range R and a constant ρ, i.e. $\sigma = \frac{\rho}{R}$. We select a high value of $\rho\ (= 18)$ so that the interagent communication becomes almost null ($\exp^{-18}$) whenever the separation between them goes above the range R. In simulation, we use a communication range (R) of 150 m. It is assumed that the target starts from the position (0,0) and maneuvers in a sinusoidal path with an acceleration input $[0 \quad 0.5\sin(\frac{2\pi t}{50})]^T$. The initial conditions of the four UAVs are shown in Table I. The initial conditions are selected so that the UAVs start from different position coordinates, with different velocities and heading directions while staying connected. We use the same set of initial conditions in all simulations.

TABLE I

Initial States of the UAVs

UAV	X co-ord (m)	Y co-ord (m)	v_i (m/s)	ϕ_i (rad)
1	18.2249	71.4778	8	0
2	-11.6509	97.6854	8.5	0.7854
3	-1.4301	133.4849	9	1.5708
4	3.8123	103.1000	9.5000	2.3562

First, we consider the control law (10), with time-varying controller weights a_{ij} chosen according to Equation (3). Figure 3 shows that the controller (Equation (10)) fails to drive UAVs to the desired formation. Figure 3(a) points out that two of the agents diverge from the team after around 6 secs. In this case, the mobile network loses the connectivity and leads to a failure in making formation. Figure 3(c) reveals that the network connectivity fails abruptly around 5 secs; this dip in the connectivity depicts that the connection among multiple agents may become very poor during the dynamic process because of their dynamic constraints or initial conditions. For Figure 4, the controller weights w_{ij} in Equation (17) are selected in accordance with Equations (13) and (14). Figure 4 shows the results of the proposed control law, shown in Equation (17), applied to the UAVs. Figures 4(a)-(b) show that the control law successfully drives multiple UAVs to the desired formation. Figure 4(c) illustrates that the control law forces UAVs to maneuver while maintaining connectivity as the connectivity value, λ_2, approaches near zero (in the order of 10^{-7}). Although Figure 4(c) shows that the network connectivity has ripples near zero after 10 secs, the controller is able to make the formation maintaining the connectivity above zero. It is to be noted that in our numerical example, the desired final connectivity (Figure 2) is much lower than the initial connectivity of the network, which actually validates the controller performance in the worst scenario where the desired formation may have poor connections among the agents.

V. Conclusions

In this paper, we developed a novel formation controller which not only controls multiple UAVs to follow a desired formation around a target but also maintains mobile network connectivity throughout time. In order to accomplish the task, we considered an undirected weighted state-dependent graph to capture the time-varying information exchange topology among the agents in the dynamic network, and modified an existing target-centric formation controller.

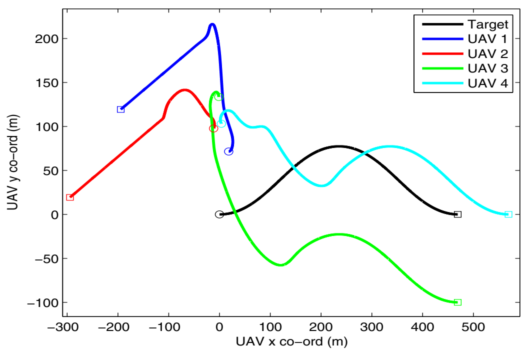
We designed new controller weights taking into account the network connectivity during the dynamic process. We analyzed the convergence of the proposed controller using the Lyapunov theory. The nonlinear Lyapunov theory leads to a sufficient condition, which leaves a room for future research on connectivity maintenance subjected to the formation control. Simulation results validate the proposed controller as well as present a comparison study over the results of the earlier controller. The controller works successfully even with low initial connectivity.

References

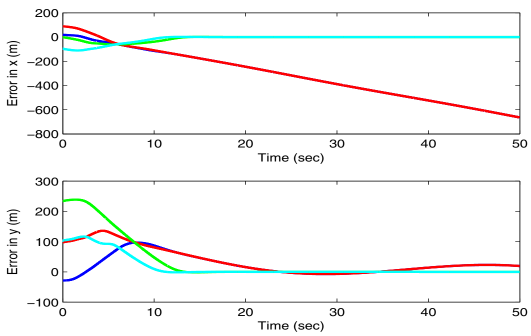
[1] R. Sharma, M. Kothari, C. N. Taylor, and I. Postlethwaite, "Cooperative target-capturing with inaccurate target information," in *American Control Conference*, 2010, pp. 5520-5525.

[2] M. Kothari, R. Sharma, I. Postlethwaite, R. Beard, and D. Pack, "Cooperative Target-capturing with Incomplete Target Information," *J. Intell Robot Syst*, pp. 1-12, 2013.

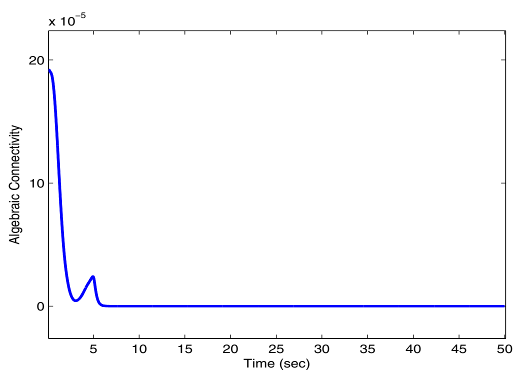
[3] H. Kawakami and T. Namerikawa, "Cooperative target-capturing strategy for multi-vehicle systems with dynamic network topology," in *American Control Conference*, 2009.



(a) Agents' trajectories in cartesian coordinates

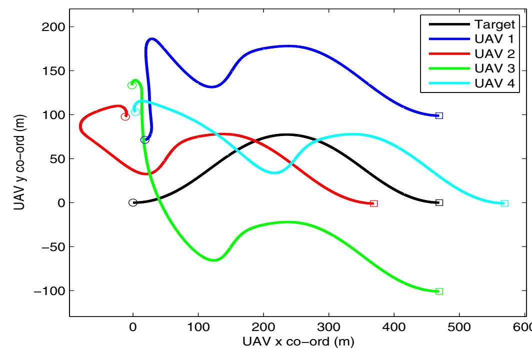


(b) Position errors

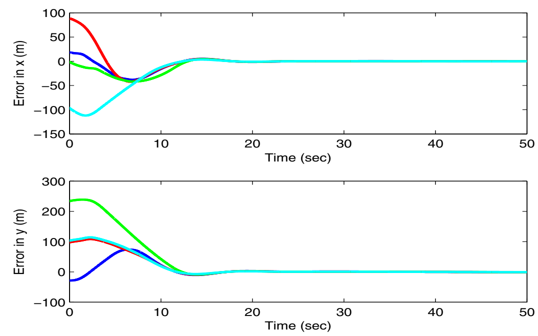


(c) Connectivity over time

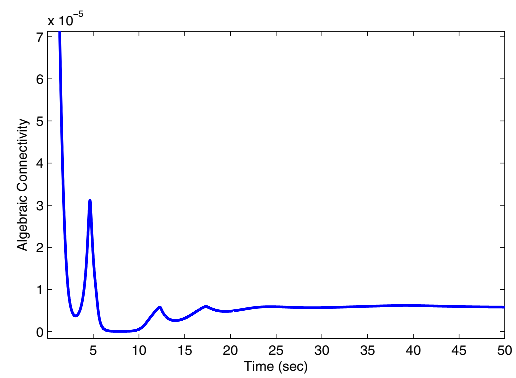
Fig. 3. Results of the existing controller for dynamic graph; ○ Initial position, □ Final position.



(a) Agents' trajectories in cartesian coordinates



(b) Position errors



(c) Connectivity over time

Fig. 4. Results of the modified controller for dynamic graph; ○ Initial position, □ Final position.

[4] Y. Q. Chen and Z. Wang, "Formation control: a review and a new consideration," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Aug. 2005, pp. 3181-3186.
[5] W. Ren, "Multi-vehicle consensus with a time-varying reference state," *Systems & Control Letters*, vol. 56, pp. 474-483, 2007.
[6] H. Tanner, G. Pappas, and V. Kumar, "Leader-to-formation stability, Robotics and Automation," *IEEE Trans. Robot. Autom*, vol. 20, no. 3, pp. 443-455, 2004.
[7] L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques, "Leader follower formation control of nonholonomic mobile robots with input constraints," *Automatica*, vol. 44, no. 5, pp. 1343-1349, 2008.
[8] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Trans. on Robotics and Automation*, vol. 14, no. 6, pp. 926-939, Dec. 1999.
[9] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 2, pp. 298-305, 1973.
[10] M. C. De Gennaro and A. Jadbabaie, "Decentralized Control of Connectivity for Multi-Agent Systems," in *IEEE Conference on Decision and Control*, San Diego, Dec. 2006, pp. 3628-3633.
[11] P. Wei and D. Sun, "Weighted Algebraic Connectivity: An Application to Airport Transportation Network," in *Proceedings of IFAC World Congress*, 2011.
[12] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoritic connectivity control of mobile robot networks," in *Proc. of the IEEE*, vol. 99, no. 9, pp. 1525-1540, 2011.
[13] M. M. Zavlanos and G. J. Pappas, "Controlling connectivity of dynamic graphs," in *Proc. of the IEEE Decision and Control*, 2005, pp. 6388-6393.
[14] Y. Kim and M. Mesbahi, "On Maximizing the Second Smallest Eigenvalue of a State-Dependent Graph Laplacian," *IEEE Trans. on Automatic Control*, vol. 51, no. 1, pp. 116-120, Jan. 2006.
[15] J. R. Magnus, "On Differentiating Eigenvalues and Eigenvectors," *Economic Theory*, vol. 1, no. 2, pp. 179-191, Aug. 1985.
[16] J. D. Kraus, *Antennas*. McGraw-Hill, 1988.
[17] H. K. Khalil, *Nonlinear systems*. Prentice Hall, 2002.