Consensus-based Cooperative Control for Geometric Configuration of UAVs
Flying in Formation

Yasuhiro Kuriki¹ and Toru Namerikawa¹

¹ Faculty of Science and Technology, School of Integrated Design Engineering, Keio University, Kanagawa, Japan
(Tel: +81-45-563-1151 Ext. 43051; E-mail: yasuhiro@nl.sd.keio.ac.jp, namerikawa@nl.sd.keio.ac.jp)

Abstract: In this paper, we study cooperative control problems with a multi-UAV system expressed as a first-order system. Specifically, we describe a problem that cooperatively controlled UAVs change geometric configuration between multiple UAVs while the UAVs are flying in formation. In other words, each of the UAVs must intentionally change relative position between the UAV and the leader while the UAVs are cooperatively controlled. We propose a control algorithm to change geometric configuration among the UAVs arbitrarily while the cooperatively controlled UAVs are flying in formation. The control algorithm is based on a consensus algorithm. A leader-follower structure is also applied to provide the UAVs with commands from a leader. The control algorithm has an advantage in finding a controller gain to stabilize the multi-UAV system. In addition, the convergence speed does not depend on the network structure of the system. Finally, the proposed approach is validated by some simulations.

Keywords: Cooperative Control, Consensus, Formation, UAVs

1. INTRODUCTION

Cooperative control problems with a multi-vehicle system have attracted a lot of attention from many researchers in recent years [1], specifically cooperative control problems using a consensus-based algorithm. And then, many algorithms have been developed for the problems under different assumptions. Authors in [2], [4]-[11] and [15]-[17] considered a first-order system, authors in [5], [7], and [11] considered a second-order system, and authors in [3], [12]-[14] considered a linear system, just to name a few. These cooperative control technologies are expected to be applied to actual vehicles such as UAVs, artificial satellites, and autonomous mobile observation robots as well as sensor networks.

There is a possibility that a multi-agent system can perform tasks more efficiently than a single highly-functional agent does. Let us consider some examples of this efficiency. First, a multi-agent system carries out tasks more quickly than the single agent does. Second, a multi-agent system has much better fault-tolerance than the single agent does. For example, when an agent breaks down and the breakdown results in an inability to go on with tasks, a group of healthy agents might fulfill the tasks without any change in the original objectives using the surviving healthy agents. The last, compared to the single agent, a multi-agent system has a cost advantage. Now, suppose two agents: One has highly sophisticated function but is too costly. The other has only limited function and deliver only poor performance but is affordable. The cooperative system composed of a group of the latter agents performs tasks just as or better than the former agent does.

Agreement problems of formation and flocking play an important role in cooperative control problems with a multi-vehicle system, because setting the states such as a position and velocity to the same value is often useful for actual problems. In addition, a network is indispensable for cooperative control, because it is necessary for vehicles to exchange information each other. In recent years, information and communication technology have been progressing rapidly. Using this advanced network technology, a multi-vehicle system is expected to be applied to actual systems. A network, however, is not always continuously or statically connected to each other. For this reason, cooperative control problems with dynamical networks [2, 4, 5, 7, 16] or communication delays have also been studied by many researchers.

Cooperative control problems with formation control have been widely studied. These problems are classified as centralized control and decentralized control. The main characteristics of the former is that a leader calculates control inputs for every follower collecting information from the followers. On the other hand, the main characteristics of the latter is that the followers calculate their own control inputs by themselves exchanging information each other. Because the former largely depends on a highly reliable network, most researchers focus on a decentralized approach. The graph theory treatment is used to express a network structure of a multi-vehicle system.

We have been studying cooperative control problems with a multi-vehicle system [14]-[17], especially focusing on a formation control problems using a consensus algorithm. In [15], we proposed a control algorithm for multiple columnar vehicles to surround a moving object cooperatively without depending on a network structure. This method, however, resulted in fixing the geometric configuration, with respect to a fixed system of coordinates, even when the object moved changing the traveling direction as shown in Fig. 1. Generally, every cooperatively controlled UAV flies aiming at a single cooperative objective. In addition, how the UAV flies is affected by the other UAVs. The UAVs do not fly aiming at its own objective. As a result, each of the cooperatively controlled UAVs must establish an individual ob-
jective, which is different from the cooperative one, to change geometric configuration of formation. Therefore, how the cooperatively controlled UAVs with an individual objective is appropriately controlled is a continuing issue.

To solve this problem, an objective of a group of UAVs and that of each UAV must be separately considered. We propose a control algorithm that UAVs fly in formation and also change the geometric configuration arbitrarily. A consensus-based cooperative control algorithm is applied to achieve the cooperative objective, and a leader-follower structure is also applied to achieve the individual object.

An outline of this paper is as follows. In section 2, modeling a UAV and a multi-UAV system are stated, and then a control objective is defined. In section 3, we propose a control algorithm to achieve the control objective defined in section 2, and then we prove the theorem. Section 4 presents simulation results to validate the proposed approach, and finally, concluding remarks are stated in Section 5.

Fig. 1 The movement of a multi-UAV system when the algorithm in [15] is applied

2. PROBLEM STATEMENT

2.1 Modeling A Multi-UAV system

Suppose that there are $N$ UAVs which have the same motion characteristics. Also, each of the UAVs has directivity, and each of them is expressed as a first-order dynamical model. This is formulated as

$$\dot{r}_i = u_i, \quad i \in \{1, 2, \cdots, N\},$$

where $r_i \in \mathbb{R}^2$ is an $i^{th}$ UAV position, and $u_i \in \mathbb{R}^2$ is an $i^{th}$ UAV control input.

A multi-UAV system is modeled as a group of dynamical system which exchanges information each other. To describe this network composed of the multiple UAVs, we use the graph theory [18].

A graph $\mathcal{G} = (V, \mathcal{A})$ is applied to model the information interaction among UAVs. $V = \{v_1, v_2, \cdots, v_N\}$ is a set of nodes, and $\mathcal{A} \subseteq V \times V$ is a set of edges. The edge $(v_i, v_j)$ in the edge set of the graph denotes that there is a network path from an $i^{th}$ UAV to a $j^{th}$ UAV. This means that the $j^{th}$ UAV can get information from the $i^{th}$ UAV.

There are two types of graphs. One is an undirected graph where both an $i^{th}$ and a $j^{th}$ node can get information from each other. The other is a digraph where an $i^{th}$ node can get information from a $j^{th}$ node, but in the reverse direction, the $j^{th}$ node cannot get information from the $i^{th}$ node. A graph has some important characteristics: First, a graph is defined as connected if there is a network path between an $i^{th}$ and a $j^{th}$ node for every pair of different vertices. This means that all UAVs can get information from each other through a network. Second, a directed tree is a digraph where every node has exactly one parent node except for one node, called a root. The root has no parent and has a directed path to every other node. Also, a spanning tree of $\mathcal{G}$ is a tree which contains all nodes of $\mathcal{G}$.

Let $\mathcal{A} \in \mathbb{R}^{N \times N}$, $\mathcal{D} \in \mathbb{R}^{N \times N}$ and $\mathcal{L} \in \mathbb{R}^{N \times N}$ be, respectively, an adjacency matrix, a degree matrix and a graph Laplacian matrix related to the graph $\mathcal{G}$. The component of the adjacency matrix $a_{ij}$ is given by

$$a_{ij} = \begin{cases} 1, & \text{for } (v_i, v_j) \in \mathcal{A} \\ 0, & \text{otherwise} \end{cases}.$$  

(2)

This means that if an $i^{th}$ UAV is obtaining information from a $j^{th}$ UAV through a network, $a_{ij}$ is set to one, otherwise $a_{ij}$ is set to zero.

The degree matrix $\mathcal{D}$ is an in-degree matrix given by

$$\mathcal{D} = \text{diag}(\deg(v_1), \deg(v_2), \cdots, \deg(v_N)),$$

(3)

where $\deg(v_i)$ is a number of communication links arriving at the node $v_i$.

The graph Laplacian matrix is defined as

$$\mathcal{L} = \mathcal{D} - \mathcal{A}.$$  

(4)

The graph Laplacian $\mathcal{L}$ has following properties: If a graph has a directed spanning tree, the graph Laplacian $\mathcal{L}$ has a single eigenvalue at zero. And besides, all nonzero eigenvalues of the graph Laplacian have positive real part.

2.2 A Control Objective

In this paper, a control objective is defined as follows: $N$ UAVs follow their leader, and both the UAVs and the leader move in formation as shown in Fig. 2. In addition, each of the UAVs has an ability to arbitrarily change the geometric configuration of formation. Note that both an actual and a virtual leader are acceptable. Also, UAVs are called as followers since the UAVs follow their leader.

Specifically, each follower $r_f(t)$ will converge to a time-variant target position $r_f(t) + d_f(t)$. The time-variant target position for a desired geometric configuration of formation is commanded by a leader. $r_f$ is a position of an $i^{th}$ follower, $d_f \in \mathbb{R}^2$ is a relative position between the $i^{th}$ follower and a leader, and $r_f \in \mathbb{R}^2$ is a position of the leader as shown in Fig. 3.

To achieve this control objective, we make a set of assumptions as follows:

Assumption 1: Every UAV must be connected from a leader on the network, but all of the UAVs are not necessarily directly connected from the leader. Also, the network between the UAVs is undirected graph.
Assumption 2: The movement of the leader must be independent from any UAV, that is, the movement of the leader is not affected by any UAVs.

3. PROPOSED APPROACH

In this section, we propose a control algorithm to achieve the control objective as mentioned in section 2.2.

To achieve the control objective, an objective of a group of UAVs and that of each UAV are separately considered. The first one is that a group of UAVs cooperatively flies in formation. The other one is that each of the UAV changes the geometric configuration of formation. A consensus-based cooperative control algorithm is applied to achieve the former cooperative objective. In addition, a leader-follower structure is applied so that a leader individually provides each of the followers with time-variant positions \( r_L + d_i \), for all \( i = 1, \ldots, N \). The leader gives each of the followers the target positions for a desired geometric configuration of formation. In other words, this commands from the leader will lead to changes in the geometric configuration of formation.

The control law which should be applied to an \( i^{th} \) UAV is given by

\[
u_i = \frac{1}{N+1} \sum_{j=1}^{N+1} a_{ij} \left\{ -k (\dot{r}_i - \dot{r}_j) + \dot{r}_j + (d_i - d_j) \right\}, \quad i \in \{1, 2, \ldots, N\},
\]

where \( k \in \mathbb{R} \) is a positive control gain, the subscript of \( N+1 \) denotes a leader, and \( \dot{r}_j \) is defined as a vector subtracting a relative position between an \( j^{th} \) follower and a leader from a position of the \( j^{th} \) follower.

Note that \( a_{ij} \) is a value which indicates whether an \( i^{th} \) UAV gets information from a \( j^{th} \) UAV or not, that is, if obtaining some information, \( a_{ij} = 1 \), otherwise \( a_{ij} = 0 \).

For the control input and the multi-UAV system composed of the UAVs and the leader, a following theorem concerning desired convergence is derived.

**Theorem 1**: Suppose that a multi-UAV system composed of \( N \geq 1 \) UAVs expressed as (1) and a leader. Also, the assumptions 1 and 2 are satisfied. When the control protocol (5) with a positive controller gain \( k \) is applied to each of the UAV, then the control objective is asymptotically achieved.

**Proof**: Applying the control protocol (5) to the \( i^{th} \) UAV expressed as (1), we get

\[
\dot{r}_i = \frac{1}{N+1} \sum_{j=1}^{N+1} a_{ij} \left\{ -k (\dot{r}_i - \dot{r}_j) + \dot{r}_j + (d_i - d_j) \right\}, \quad i \in \{1, 2, \ldots, N\}.
\]

Also, get

\[
\sum_{j=1}^{N+1} a_{ij} \{ \dot{r}_i - \dot{r}_j \} = -k \sum_{j=1}^{N+1} a_{ij} \{ \dot{r}_i - \dot{r}_j \}, \quad i \in \{1, 2, \ldots, N\}.
\]

Eq. (9) is rewritten in a matrix-vector form as

\[
(\mathcal{L} \otimes I_2)(\dot{\hat{r}} + k \hat{r}) = 0,
\]

where \( \mathcal{L} \in \mathbb{R}^{(N+1) \times (N+1)} \) is the graph Laplacian of the multi-UAV system defined as (11), \( \otimes \) is the kronecker
product, \( I_2 \) is a two-dimensional unit matrix, and \( \hat{r} = [\hat{r}_1^T \hat{r}_2^T \ldots \hat{r}_N^T \hat{r}_{N+1}^T]^T \in \mathbb{R}^{2(N+1)}. \)

Let us examine the relationship between the eigenvalues of the graph Laplacian \( L \) exists.

It follows from the definition of the two matrix that one of the eigenvalues of the graph Laplacian \( L \) is zero, and the rest of the eigenvalues are equal to the eigenvalues of the matrix \( \mathcal{M} \). The graph Laplacian \( L \) has a single eigenvalue at zero, and all nonzero eigenvalues of the graph Laplacian have positive real part when the graph has a directed spanning tree, that is, when the assumptions 1 and 2 are satisfied. As a result, all eigenvalues of the matrix \( \mathcal{M} \) have positive real part, that is, the matrix \( \mathcal{M} \) is positive definite. Therefore, the inverse matrix of the matrix \( \mathcal{M} \) exists.

Because the inverse matrix of the matrix \( \mathcal{M} \) exists, we get next equation (15) from (13).

\[
\hat{r} + kr - \hat{r}_{N+1} - k\hat{r}_{N+1} - \hat{d} - kd = 0. \tag{15}
\]

Hence, we get a general solution of the differential equation (15).

\[
r(t) = (r(0) - \hat{r}(0)) e^{-kt} + \hat{r}(t)_{N+1} + d(t), \tag{16}
\]

where \( r(0) = [r(0)_1^T r(0)_2^T \ldots r(0)_N^T]^T \in \mathbb{R}^{2N} \) is initial positions of the followers. Also, \( \hat{r}(0)_{N+1} = [r_L(0)_1^T r_L(0)_2^T \ldots r_L(0)_N^T]^T \) and \( d(0) = [d_1(0)^T d_2(0)^T \ldots d_N(0)^T]^T \) are initial commands of a leader. Note that a leader individually provides both its own position \( r_{N+1} \) and relative positions \( d \) for each of the followers.

The first term of (16) will converge to zero, if and only if \( k > 0 \).

\[
\lim_{t \to \infty} (r(0) - \hat{r}(0))_{N+1} - d(0) e^{-kt} = 0. \tag{17}
\]

Hence, we can get

\[
\begin{align*}
& r_1 \to r_L + d_1 \\
& r_2 \to r_L + d_2 \\
& \vdots \\
& r_N \to r_L + d_N,
\end{align*}
\]

as to \( t \to \infty \). \tag{18}

Therefore, it is proved that the control objective is asymptotically achieved when the control protocol (5) with a positive controller gain \( k \) is applied to the followers.

Corollary 1: For any network structure satisfying the assumptions 1 and 2, the convergence speed does not depend on the network structure, but only depends on the controller gain \( k \).

Proof: When a network structure satisfies the assumptions 1 and 2, we can get Eq. (15). Therefore, the convergence speed only depends on the controller gain \( k \).

4. SIMULATION RESULTS

In this section, we present some simulation results to validate the performance of the proposed control algorithm.

4.1 Simulation Setup

Simulations are carried out for two cases as shown in Table 1. We consider a group of three UAVs and a leader trying to fly in formation. The controller gain \( k \) is set to 5 so that the control objective is achieved throughout the simulations. Details are described in the following sections.

<table>
<thead>
<tr>
<th>Network Structure</th>
<th>CASE I</th>
<th>CASE II</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NET1</td>
<td>NET2</td>
</tr>
</tbody>
</table>

4.1.1 Network Structure

Let us consider two types of network structures. The first network, which is named NET1, is shown in Fig. 4 (a): Every UAV can get information directly from their leader. The second network, which is named NET2, is
shown in Fig. 4 (b): The only first UAV can get information directly from their leader while every UAV is connected from their leader, and the other UAVs indirectly get information of their leader through the other UAVs.

4.1.2 Leader’s Path

A single leader’s path is specified throughout the simulations. The leader flies in an elliptical orbit whose major axis is 10(m), minor axis is 8(m), and cycle is 30(s). Also, the initial position is [0 0](m).

4.1.3 Formation Configuration

A single configuration is specified throughout the simulations. Each UAV keeps a certain distance from the leader, and also keeps a certain azimuth from the traveling direction of the leader. Note that the traveling direction of the leader is formulated as a unit velocity vector \( \hat{r}_{N+1}/|\hat{r}_{N+1}| \). We describe this information as a desired relative position \( d_i \) for an \( i^{th} \) UAV. This is formulated as

\[
\begin{align*}
\dot{d}_i &= \frac{|d_i|}{|\hat{r}_{N+1}|} R(\theta_i) \hat{r}_{N+1} & (|\hat{r}_{N+1}| \geq 0.1), \\
\dot{d}_i &= |d_i| R(\theta_i) r_0 & (|\hat{r}_{N+1}| < 0.1),
\end{align*}
\]

where \(|d_i|\) is a relative distance between the \( i^{th} \) UAV and the leader, \( \theta_i \) is an azimuth which goes counterclockwise from the leader’s traveling direction as shown in Fig. 5, and \( r_0 = [1 1]^T/\sqrt{2} \). Also, \( R(\theta_i) \) is a two-dimensional rotation matrix given by

\[
R(\theta_i) = \begin{bmatrix}
\cos(\theta_i) & -\sin(\theta_i) \\
\sin(\theta_i) & \cos(\theta_i)
\end{bmatrix}.
\]

Note that the relative position \( d_i \) is set to a certain fixed point which does not depend on the leader’s traveling direction while the velocity is slower than 0.1(m/s).

The values of \(|d_i|\) and \( \theta_i \) are shown in Table 2. These parameter will create geometric configuration in shape of a isosceles triangle.

<table>
<thead>
<tr>
<th>UAV</th>
<th>( d_1 )</th>
<th>( d_2 )</th>
<th>( d_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative Distance (</td>
<td>( d_i</td>
<td>))</td>
<td>2</td>
</tr>
<tr>
<td>Azimuth (( \theta_i ))</td>
<td>0</td>
<td>90</td>
<td>270</td>
</tr>
</tbody>
</table>

4.2 Simulation Results

We show the results of the simulation which is carried out for 20(s) under the cases shown in Table 1. Figs. 6 and 8 show the results of the case CASE I, and Figs. 7 and 9 show the results of the case CASE II.

Figs. 6 and 7 show the trajectories of the UAVs and the leader, and the positions every 2 seconds. Figs. 8 and 9 show the difference from the desired position.

Now, let us examine the results. It follows from the results of the case CASE I with CASE II that the control objective is achieved even when every UAV does not keep obtaining the information directly from their leader, and the convergence speed does not depend on the network structure as mentioned in Corollary 1. Furthermore, every UAV arrives at the desired position simultaneously because of the consensus-based cooperative control.

5. CONCLUSIONS AND FUTURE WORK

We showed that a multi-UAV system can be modeled using the graph theory, then proposed a control algorithm to change geometric configuration among the UAVs arbitrarily while the cooperatively controlled UAVs are flying in formation. Extending the control algorithm that a group of vehicles expressed as a first-order system cooperatively surround a moving columnnr object, we applied a leader-follower structure so that a leader can individually provide information of the target positions for each of the UAVs. To achieve the control objective, the network composed of the UAVs and the leader must satisfy requirements that every UAV can obtain commands from the leader and the network between the UAVs is bidirectional.

Our extensive simulation results showed that the proposed algorithm was validated and effective for controlling geometric configuration of formation among a group of the UAVs and the leader.

In this paper, we considered the UAV expressed as a first-order system, so studying and designing a controller for a more general system will be a challenging problem for future work in this area.

REFERENCES


