Consensus-based cooperative formation control for multi-quadcopter system with unidirectional network connections

Toru Namerikawa and Yasuhiro Kuriki
Department of Integrated Design Engineering, Graduate School of Science and Technology, Keio University, Yokohama, Japan.
Email: namerikawa@sd.keio.ac.jp, yasuhiro@nl.sd.keio.ac.jp

Ahmed Khalifa

Department of Industrial Electronics and Control Engineering, Faculty of Electronic Engineering, Menoufia University, Menoufia, Egypt.
Email: ahmed.khalifa@el-eng.menofia.edu.eg

In this article, we consider cooperative control issues for a multi-unmanned aerial vehicle (UAV) system. We propose a cooperative formation control strategy with unidirectional network connections between UAVs. Our strategy is to apply a consensus-based algorithm to the UAVs so that they can cooperatively fly in formation. First, we show that UAV models on the horizontal plane and in the vertical direction are expressed as a fourth- and second-order system, respectively. Then, we show that the stability discriminants of the multi-UAV system on the horizontal plane and in the vertical direction are expressed as polynomials. For a network structure composed of bidirectional or unidirectional network connections under the assumption that the network has a directed spanning tree, we provide conditions for formation control gains such that all roots of the polynomials have negative real parts in order for the UAVs to asymptotically converge to the positions for a desired formation by using the generalized Routh stability criterion. The proposed control algorithms are validated through simulations, and experiments are performed on multiple commercial small UAVs to validate the proposed control algorithm.

1 Introduction

In recent years, cooperative control problems with multi-vehicle systems have attracted much research attention [1]. Cooperative control technologies have enormous potential for application to vehicles such as Unmanned Aerial Vehicles (UAVs), artificial satellites, and autonomous mobile observation robots as well as distributed sensor networks. A multi-vehicle system may be able to perform tasks more efficiently than a single highly functional vehicle.

In recent years, formation control problems for a multi-vehicle system have been widely studied, and many control algorithms have been developed for this purpose. A popular formation control strategy is to apply a consensus algorithm [2–7].

In [5], the design of nonlinear decentralized controllers is proposed to maintain a tight formation of a team of quadcopters. Further, the effects of both the time delays and communication failures in the network are studied and evaluated.

Some studies on consensus-based algorithms for cooperative control problems have assumed that vehicles can be expressed as a first-order system [8–10]. The results of a first-order system can be directly extended to a second-order system [11]. Some researchers have mainly used complicated methods such as an optimization approach [12] or Linear Matrix Inequality (LMI) [13].

In [14], the problem of simultaneous tracking and formation control for multi autonomous agents system is addressed. The control and estimation architecture is placed in a consensus framework. The solution is based on agents evolve in a space that possesses a measurable vector field (e.g., temperature, or magnetic field, ..., etc.). Their approach utilizes the concept of virtual leader and the graph theory. Also, they consider the time-varying communication issue inside the network. Furthermore, the tracking and formation control is achieved in a two-level framework; one for the virtual leader estimation and the other is for the formation control.

In [15], the authors propose a solution to the formation problem for teams of unicycles, as an example of a nonholonomic mobile robot, based on the consensus algorithms and potential fields. Stability of the proposed control framework is studied. In addition, [16] presents an optimal and robust formation algorithm for non-holonomic vehicles.

We have studied the cooperative control problems for multi-vehicle systems [17–19] with a particular focus on solving formation control problems using a consensus algorithm.

In [17], we expressed the UAV dynamics on the horizontal plane as a fourth-order system and proposed a consensus-

based control algorithm for a group of UAVs to fly in formation cooperatively. The proposed approach is validated via numerical simulations. In [18], we expressed the UAV dynamics in the vertical direction as a second-order system and proposed a consensus-based formation control algorithm with collision-avoidance capability. In this work, the network connections between the quadrotors must be bidirectional.

In [19], we present a cooperative formation control strategy for a multi-UAV system with unidirectional network links. Our strategy is to apply a consensus-based algorithm and leader-follower structure to the UAVs so that they can cooperatively fly in formation in vertical direction only. In addition, the network connections between the quadrotors must be bidirectional. Moreover, experiments are performed on multiple commercial small UAVs to validate the proposed formation control algorithm with collision-avoidance capability. However, in our previous work, the network connections between UAVs must be bidirectional to ensure the stability of the proposed formation control algorithms.

The use of unidirectional network connections enables more alternatives to the network structure. The consensus-based control algorithm for a group of first-order systems can accept bidirectional or unidirectional network connections if the network structure has a directed spanning tree [10].

On the other hand, this does not apply to consensus-based control algorithms for a group of second- or higher-order systems. In this study, for a group of UAVs expressed as a fourth-order system and another group expressed as a second-order system, we provide conditions for formation control algorithms under network structures with unidirectional network connections between UAVs.

Our work builds on this literature but differs in many ways. First, the network connection between the quadrotors is considered to be unidirectional. Second, we explicitly model the team as fourth- and second-order systems. Third, the proposed control framework has advantage that all the agents do not have to directly connected to the leader and it is sufficient for an agent to be connected with its neighbors. Fourth, such a formulation permits a discussion of stability and convergence properties of the system to the desired formation shape given the communication and sensing graphs by using the generalized Routh - Hurwitz criterion. Especially, the third and fourth ones should have strong novelties and contributions in this field.

2 Problem statement

In this section, we describe how to model a quadrotor and multi-UAV system and define the control objective.

2.1 Modeling a quadrotor

We assume N quadrotors with the same motion characteristics. The multi-UAV system consists of N quadrotors and a leader. Each quadrotor has four propellers and a controller. The propellers are individually given control commands by the controller. To simply model the quadrotor mathematically, we assume the following. First, the quadrotor flies slowly enough to ignore external aerodynamic forces such as aerodynamic drag and a blade vortex interface acting on the body. Second, the propellers respond to thrust commands fast enough to ignore the time delay from when the controller gives the propellers the thrust command until the propellers actually produce the commanded thrust. Finally, a yawing moment is never produced.

Under the condition that hovering at a constant altitude is an equilibrium point of a nonlinear quadrotor system, a linearized model of the quadrotor on a horizontal plane is given by

$$\frac{d}{dt}\begin{bmatrix} r_i^{(0)} \\ r_i^{(1)} \\ r_i^{(2)} \\ r_i^{(3)} \end{bmatrix} = \begin{bmatrix} r_i^{(1)} \\ r_i^{(2)} \\ r_i^{(3)} \\ M_i \end{bmatrix}, \ i \in \{1,2,\cdots,N\}, \tag{1}$$

where $r_i^{(k)} \in \mathbb{R}^2$ and $M_i \in \mathbb{R}^2$ are the state and control input, respectively, of quadrotor i on the horizontal plane. The superscript denotes the order of the variable, and the subscript i indicates which quadrotor is being referred to.

Specifically, $r_i^{(k)}$ is defined as the combined vector $r_i^{(k)} = [r_x^{(k)} \ r_y^{(k)}]^T$, $k \in \{0,1,2,3\}$, $i \in \{1,2,\cdots,N\}$, when longitudinal variables $r_x^{(k)}$, $k \in \{0,1,2,3\}$, are defined as $r_x^{(0)} = x$, $r_x^{(1)} = u$, $r_x^{(2)} = -g\theta$, and $r_x^{(3)} = -gq$, and lateral variables $r_y^{(k)}, k \in \{0,1,2,3\}$, are defined as $r_y^{(0)} = y$, $r_y^{(1)} = v$, $r_y^{(2)} = g\phi$, and $r_y^{(3)} = gp$. Furthermore, M_i is defined as the combined vector $M_i = [-\frac{g}{I_{yy}}M_\theta \quad \frac{g}{I_{xx}}M_\phi]^T$ using the longitudinal control input M_θ and lateral control input M_ϕ. Note that the controller input is a magnitude of the moment about the longitudinal or lateral axes. The magnitude of the thrust that each propeller should produce can be derived from the magnitude of the moment using the geometric position of the propellers.

Under the same condition as the horizontal plane, the linearized model of the quadrotor in the vertical direction is given by

$$\frac{d}{dt}\begin{bmatrix} h_i^{(0)} \\ h_i^{(1)} \end{bmatrix} = \begin{bmatrix} h_i^{(1)} \\ \tilde{T}_{total_i} \end{bmatrix}, \quad i \in \{1,2,\cdots,N\}, \tag{2}$$

where $h^{(0)} = h$, $h^{(1)} = -w$, and $\tilde{T}_{total} = T_{total}/m$. Note that $\tilde{T}_{total}/4$ is equally assigned to each propeller. Table 1 defines the above symbols.

2.2 Modeling a multi-UAV system

We can model a multi-UAV system as a group of dynamical systems where multiple UAVs and a leader exchange information with each other. This network can be mathematically described using graph theory ([20]).

We use the graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ to model information interaction among N UAVs, where $\mathcal{V} = \{v_1, v_2, \cdots, v_N\}$ is a

Table 1. Definition of Symbols.

Symbol	Definition
x, y (m)	Position (x-axis, y-axis)
h (m)	Altitude
u, v, w (m/s)	Velocity (x-axis, y-axis, z-axis)
ϕ, θ (rad)	Attitude angle (Roll, Pitch)
p, q (rad/s)	Angular velocity (Roll, Pitch)
M_ϕ, M_θ (Nm)	Moment command (Roll, Pitch)
T_{total} (N)	Total thrust command
I_{xx}, I_{yy} (kgm^2)	Inertia moment (Roll, Pitch)
m (kg)	Quadrotor mass
g (m/s^2)	Gravity constant

set of nodes and $\mathcal{A} \in \mathcal{V} \times \mathcal{V}$ is a set of edges. The edge (v_i, v_j) in the edge set of the graph denotes a network path from UAV i to UAV j. This means that UAV j can obtain and use information from UAV i.

A directed tree is a digraph where every node, except for one node called the root, has exactly one parent node. The root has no parent and has a directed path to every other node. A directed spanning tree of $\mathcal{G}$ is a tree that 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 an adjacency matrix, degree matrix, and graph Laplacian matrix, respectively, related to the graph $\mathcal{G}$. The component of the adjacency matrix $\mathcal{A} = [a_{ij}]$ is given by

$$a_{ij} = \begin{cases} 1, \text{ for } (v_j, v_i) \in \mathcal{A} \\ 0, \textit{otherwise}, \end{cases} \tag{3}$$

This means that a_{ij} is set to 1 if UAV i is obtaining information from UAV j through a network; otherwise, a_{ij} is set to 0.

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

$$\mathcal{D} = \text{diag}(\deg(v_1), \deg(v_2), \cdots, \deg(v_N)), \tag{4}$$

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

The graph Laplacian matrix $\mathcal{L}$ is defined as

$$\mathcal{L} = \mathcal{D} - \mathcal{A}, \tag{5}$$

which can be calculated by

$$\mathcal{L} = \begin{bmatrix} \sum_{j=1}^{N+1} a_{1j} & -a_{12} & \dots & -a_{1N} & -a_{1(N+1)} \\ -a_{21} & \sum_{j=1}^{N+1} a_{2j} & \dots & -a_{2N} & -a_{2(N+1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_{N1} & -a_{N2} & \dots & \sum_{j=1}^{N+1} a_{Nj} & -a_{N(N+1)} \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}, \tag{6}$$

where a_{ij} can be determined before the starting the team operation as it depends on the network topology from as given in (3).

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

2.3 Control objective

We consider the following mission: N quadrotors fly in formation following their leader in three-dimensional space. Fig. 1 shows three quadrotors flying in formation while following their leader. In this figure, the UAVs are required to make a triangular formation which resembles the one used in [21]. However, each of the vehicles has an ability to arbitrarily change the geometric configuration of formation. Note that both an actual and a virtual leader are acceptable [22].

More specifically, each quadrotor converges to a time-variant desired position. These positions are determined by the desired geometric configuration of the formation.

To achieve this control objective, we make the following assumptions:

Assumption 1. *The communication graph having a spanning tree with root at the leader.*

Assumption 2. *The movement of the leader must be independent of the quadrotors, i.e., must not be affected by any of the quadrotors.*

3 Proposed approach

In this section, we present our proposed control algorithm to achieve the control objective given in Section 2.3. As described in (1) and (2), the quadrotor model is expressed as a fourth-order system on the horizontal plane and as a second-order system in the vertical direction. Therefore, to fly in formation in three-dimensional space, separate control algorithms are applied for formation flying on the horizontal plane and in the vertical direction.

3.1 Formation flying on horizontal plane

For formation flying, we consider the objectives of a group of quadrotors and of each quadrotor separately. The

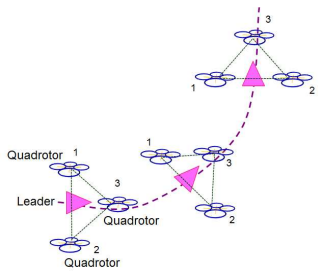

Fig. 1. Desired formation for three UAVs and leader.

first objective is for the group of quadrotors to fly cooperatively in formation. The second objective is for each quadrotor to generate a geometric configuration of the formation. We apply a consensus-based cooperative control algorithm to achieve the former cooperative objective and a leader–follower structure to achieve the latter individual objective. The leader individually provides only the directly connected quadrotors with its own position and desired positions for formation. The advantages of our proposed control algorithm are that all quadrotors do not necessarily need to be directly connected to the leader and that it is sufficient for a quadrotor to be connected with its neighbors if the network structure satisfies assumptions 1 and 2. The control law for quadrotor i is given by

$$M_i(t) = -\sum_{j=1}^{N+1} a_{ij}\left[\sum_{k=0}^{3}\beta_k(\hat{r}_i^{(k)} - \hat{r}_j^{(k)})\right], \quad i \in \{1,2,\cdots,N\}, \tag{7}$$

$$\hat{r}_j^{(k)} = r_j^{(k)} - d_{r_j}^{(k)}, j \in \{1,2,\cdots,N+1\}, \quad k \in \{0,1,2,3\}, \tag{8}$$

where subscript $N+1$ denotes a leader and $\beta_k \in \mathbb{R}$, $k \in \{0,1,2,3\}$, are positive control gains. The gains must satisfy conditions (9)–(14) for the desired formation flying. Furthermore, $d_{r_j}^{(k)} \in \mathbb{R}^2$, $k \in \{0,1,2,3\}$, is the desired relative state between quadrotor j and the leader on the horizontal plane. The states of the leader and the desired relative states are provided by the leader to only the directly connected quadrotors. Note that a UAV must obtain information including the position, velocity, attitude angle, angular velocity, and desired relative states from the connected quadrotors.

The following theorem for the desired convergence is derived.

Theorem 1. *Suppose that a multi-UAV system comprises $N(\geq 1)$ quadrotors expressed as (1) and a leader, and the control protocol (7) is applied to each quadrotor. In addition, assumptions 1 and 2 are satisfied. Let $\lambda = \lambda_R + i\lambda_I \in \mathbb{C}$ be a nonzero eigenvalue of the graph Laplacian of a multi-UAV system. When the control gains β_k for $k \in \{0,1,2,3\}$ are selected so as to satisfy conditions (9) and (10) for the real eigenvalues and when the control gains β_k for $k \in \{0,1,2,3\}$ are selected so as to satisfy conditions (9)–(14) for the complex eigenvalues, then all states of the quadrotors on the horizontal plane asymptotically converge to the desired states.*

$$\beta_k > 0, \ \forall k \in \{0,1,2,3\} \tag{9}$$

$$(\beta_1\beta_2\beta_3 - \beta_0\beta_3^2)\lambda_R - \beta_1^2 > 0 \tag{10}$$

$$\delta_3 = \beta_3^2\lambda_R - \beta_2 > 0 \tag{11}$$

$$\beta_2\delta_2 - \delta_1 > 0 \tag{12}$$

$$\delta_4 = (\beta_1^2 - \beta_0\beta_3^2\lambda_R)\lambda_R^2 - \beta_0\delta_3\lambda_I^2 > 0 \tag{13}$$

$$\delta_1\left(\delta_1\left(\delta_3\lambda_I^2 + \beta_3^2\lambda_R^3\right) - \beta_1^2\beta_3\lambda_R^2\right) - \beta_0\beta_3^2\delta_4 > 0 \tag{14}$$

where $\delta_1 = \beta_1\beta_2 - \beta_0\beta_3$ *and* $\delta_2 = \beta_2\beta_3\lambda_R - \beta_1$.

Proof. By applying the control protocol (7) to quadrotor i expressed as (1), we obtain

$$\dot{r}_i^{(3)} = -\sum_{j=1}^{N+1} a_{ij}\left[\sum_{k=0}^{3}\beta_k(\hat{r}_i^{(k)} - \hat{r}_j^{(k)})\right], \quad i \in \{1,2,\cdots,N\}. \tag{15}$$

From the proof of Theorem 1 in [18], equation (15) can be rewritten in matrix-vector form as

$$\begin{aligned}\begin{bmatrix}\dot{r}^{(0)}\\ \dot{r}^{(1)}\\ \dot{r}^{(2)}\\ \dot{r}^{(3)}\end{bmatrix} &= \begin{bmatrix} 0_{2N} & I_{2N} & 0_{2N} & 0_{2N}\\ 0_{2N} & 0_{2N} & I_{2N} & 0_{2N}\\ 0_{2N} & 0_{2N} & 0_{2N} & I_{2N}\\ -\beta_0\tilde{\mathcal{M}} & -\beta_1\tilde{\mathcal{M}} & -\beta_2\tilde{\mathcal{M}} & -\beta_3\tilde{\mathcal{M}}\end{bmatrix}\begin{bmatrix}r^{(0)}\\ r^{(1)}\\ r^{(2)}\\ r^{(3)}\end{bmatrix}\\ &+\begin{bmatrix} 0_{2N} & 0_{2N} & 0_{2N} & 0_{2N}\\ 0_{2N} & 0_{2N} & 0_{2N} & 0_{2N}\\ 0_{2N} & 0_{2N} & 0_{2N} & 0_{2N}\\ \beta_0\tilde{\mathcal{M}} & \beta_1\tilde{\mathcal{M}} & \beta_2\tilde{\mathcal{M}} & \beta_3\tilde{\mathcal{M}}\end{bmatrix}\begin{bmatrix}\tilde{r}_{N+1}^{(0)}\\ \tilde{r}_{N+1}^{(1)}\\ \tilde{r}_{N+1}^{(2)}\\ \tilde{r}_{N+1}^{(3)}\end{bmatrix}\\ &+\begin{bmatrix} 0_{2N} & 0_{2N} & 0_{2N} & 0_{2N}\\ 0_{2N} & 0_{2N} & 0_{2N} & 0_{2N}\\ 0_{2N} & 0_{2N} & 0_{2N} & 0_{2N}\\ \beta_0\tilde{\mathcal{M}} & \beta_1\tilde{\mathcal{M}} & \beta_2\tilde{\mathcal{M}} & \beta_3\tilde{\mathcal{M}}\end{bmatrix}\begin{bmatrix}d_r^{(0)}\\ d_r^{(1)}\\ d_r^{(2)}\\ d_r^{(3)}\end{bmatrix},\end{aligned} \tag{16}$$

where $I_n \in \mathbb{R}^{n\times n}$ is an n-dimensional unit matrix and $0_n \in \mathbb{R}^{n\times n}$ is an n-dimensional zero matrix. Furthermore, the matrix $\mathcal{M} \in \mathbb{R}^{N\times N}$ is defined as (17), and $\tilde{\mathcal{M}} = \mathcal{M} \otimes I_2$, where $\otimes$ denotes the Kronecker product. In addition, $r^{(k)} = [r_1^{(k)^T}\ r_2^{(k)^T}\ \cdots\ r_N^{(k)^T}]^T \in \mathbb{R}^{2N}$ and $\tilde{r}_{N+1}^{(k)} = \mathbf{1}_N \otimes r_{N+1}^{(k)} \in \mathbb{R}^{2N}$, where $\mathbf{1}_N = [1\ 1\ \cdots\ 1]^T \in \mathbb{R}^N$, and $d_r^{(k)} = [d_{r_1}^{(k)^T}\ d_{r_2}^{(k)^T}\ \cdots\ d_{r_N}^{(k)^T}]^T \in \mathbb{R}^{2N}$, for $k \in \{0,1,2,3\}$.

$$\mathcal{M} = \begin{bmatrix}\sum_{j=1}^{N+1} a_{1j} & -a_{12} & \cdots & -a_{1N}\\ -a_{21} & \sum_{j=1}^{N+1} a_{2j} & \cdots & -a_{2N}\\ \vdots & \vdots & \ddots & \vdots\\ -a_{N1} & -a_{N2} & \cdots & \sum_{j=1}^{N+1} a_{Nj}\end{bmatrix} \tag{17}$$

Note that the matrix $\mathcal{M}$ is generated by eliminating the last row and column of the graph Laplacian $\mathcal{L}$.

Now, consider the solution to differential equation (16). To examine the stability of (16), consider the homogeneous equation of (16) expressed as

$$\dot{\rho}_r = \mathcal{N}_r \rho_r, \tag{18}$$

where $\rho_r = [r^{(0)^T}\ r^{(1)^T}\ r^{(2)^T}\ r^{(3)^T}]^T$ and the matrix $\mathcal{N}_r$ is expressed as

$$\mathcal{N}_r = \begin{bmatrix} 0_{2N} & I_{2N} & 0_{2N} & 0_{2N} \\ 0_{2N} & 0_{2N} & I_{2N} & 0_{2N} \\ 0_{2N} & 0_{2N} & 0_{2N} & I_{2N} \\ -\beta_0\tilde{\mathcal{M}} & -\beta_1\tilde{\mathcal{M}} & -\beta_2\tilde{\mathcal{M}} & -\beta_3\tilde{\mathcal{M}} \end{bmatrix}. \tag{19}$$

The stability of (18) is determined by the eigenvalues of the matrix $\mathcal{N}_r$. First, let λ and s be an eigenvalue and eigenvector of the matrix $\mathcal{M}$, respectively:$\mathcal{M}s = \lambda s$. In addition, let μ_r and σ_r be an eigenvalue and eigenvector of the matrix $\mathcal{N}_r$, respectively: $\mathcal{N}_r\sigma_r = \mu_r\sigma_r$. Then, from (18), we can obtain the next equality:

$$\mathcal{N}_r\sigma_r = \mu_r\sigma_r, \tag{20}$$

where σ_r is expressed as

$$\sigma_r = \begin{bmatrix} (s \otimes 1_2) \\ \mu_r(s \otimes 1_2) \\ \mu_r^2(s \otimes 1_2) \\ \mu_r^3(s \otimes 1_2) \end{bmatrix}. \tag{21}$$

From the lowest row block of (19) and $\mathcal{M}s = \lambda s$, we can obtain the following stability discriminant:

$$\mu_r^4 + \beta_3\lambda\mu_r^3 + \beta_2\lambda\mu_r^2 + \beta_1\lambda\mu_r + \beta_0\lambda = 0. \tag{22}$$

Equation (21) shows the relationship between the eigenvalues of the matrix $\mathcal{M}$ and those of the matrix $\mathcal{N}_r$. Specifically, this indicates that the matrix $\mathcal{N}_r$ has four eigenvalues μ_r for each eigenvalue λ of the matrix $\mathcal{M}$.

Here, we examine this relationship. From the definition of the two matrices, one of the eigenvalues of the graph Laplacian $\mathcal{L}$ is zero, and everything other than the zero eigenvalue is equal to the eigenvalues of the matrix $\mathcal{M}$. In general, a graph Laplacian $\mathcal{L}$ has a single eigenvalue at zero, and all nonzero eigenvalues of the graph Laplacian have a positive real part when the graph has a directed spanning tree. Satisfying assumptions 1 and 2 would indicate that the graph has a directed spanning tree. Therefore, the graph Laplacian $\mathcal{L}$ of the multi-UAV system has a single eigenvalue at zero, and all nonzero eigenvalues of the graph Laplacian have a positive real part when assumptions 1 and 2 are satisfied. As a result, all eigenvalues λ of the matrix $\mathcal{M}$ have a positive real part when assumptions 1 and 2 are satisfied: $\lambda_R > 0$.

Now, all real parts of the eigenvalues of the matrix $\mathcal{N}_r$ must be negative so that all states of the quadrotors on the horizontal plane can asymptotically converge to the desired states.

First, consider the real eigenvalues λ. In this case, equation (21) has only real coefficients. From the Routh-Hurwitz stability criterion, control gains β_k, $k \in \{0,1,2,3\}$, must satisfy the following conditions for all real parts of μ_r to be negative.

$$\beta_0\lambda > 0,\ \beta_1\lambda > 0,\ \beta_2\lambda > 0,\ \beta_3\lambda > 0, \tag{23}$$

$$\begin{vmatrix} \beta_3\lambda & \beta_1\lambda \\ 1 & \beta_2\lambda \end{vmatrix} > 0, \begin{vmatrix} \beta_3\lambda & \beta_1\lambda & 0 \\ 1 & \beta_2\lambda & \beta_0\lambda \\ 0 & \beta_3\lambda & \beta_1\lambda \end{vmatrix} > 0, \beta_0\lambda \begin{vmatrix} \beta_3\lambda & \beta_1\lambda & 0 \\ 1 & \beta_2\lambda & \beta_0\lambda \\ 0 & \beta_3\lambda & \beta_1\lambda \end{vmatrix} > 0. \tag{24}$$

From (23) and (24), we can obtain conditions (9)–(10).

Second, consider the complex eigenvalues λ. In this case, equation (21) has complex coefficients. We use Theorem 2 in [23], which is the generalized Routh stability criterion for the polynomials with complex coefficients, to provide the conditions for all real parts of μ_r to be negative. Applying Theorem 2, we can obtain conditions (9)–(14) for all real parts of μ_r to be negative when the network satisfies assumptions 1 and 2.

Applying these appropriate gains to controller (7) produces the following convergence:

$$\lim_{t\to\infty} \rho_r = 0. \tag{25}$$

Next, we consider the particular solution of (16). One of them can be given by

$$\begin{bmatrix} r^{(0)} \\ r^{(1)} \\ r^{(2)} \\ r^{(3)} \end{bmatrix} = \begin{bmatrix} \tilde{r}_{N+1}^{(0)} \\ \tilde{r}_{N+1}^{(1)} \\ \tilde{r}_{N+1}^{(2)} \\ \tilde{r}_{N+1}^{(3)} \end{bmatrix} + \begin{bmatrix} d_r^{(0)} \\ d_r^{(1)} \\ d_r^{(2)} \\ d_r^{(3)} \end{bmatrix}. \tag{26}$$

This validation can be confirmed by substituting (26) for (16). To confirm this validation, we use $\tilde{r}_{N+1}^{(4)} = 0$ and $d_r^{(4)} = 0$ because the leader provides each quadrotor with the desired state, not the desired input. Note that $\tilde{r}_{N+1}^{(4)} = 0$ and $d_r^{(4)} = 0$ denote the input of the leader and the desired relative input between the quadrotor and the leader, respectively.

The general solution of the nonhomogeneous differential equation (16) is the sum of the particular solution and the general solution of the homogeneous equation. Therefore, if the controller gains β_k for $k \in \{0,1,2,3\}$ are appropriately selected—that is, these gains are selected so as to

satisfy conditions (9)–(14)—the general solution asymptotically converges to

$$\begin{bmatrix} r^{(0)} \\ r^{(1)} \\ r^{(2)} \\ r^{(3)} \end{bmatrix} \rightarrow \begin{bmatrix} \tilde{r}_{N+1}^{(0)} \\ \tilde{r}_{N+1}^{(1)} \\ \tilde{r}_{N+1}^{(2)} \\ \tilde{r}_{N+1}^{(3)} \end{bmatrix} + \begin{bmatrix} d_r^{(0)} \\ d_r^{(1)} \\ d_r^{(2)} \\ d_r^{(3)} \end{bmatrix}, \quad \text{as} \quad t \rightarrow \infty. \tag{27}$$

This results in the convergence to only the commands from the leader.

From the element of the first row block in (27), we prove that formation flying on the horizontal plane is asymptotically achieved when control protocol (7) with appropriate controller gains β_k is applied to quadrotor i expressed as (1). □

3.2 Formation flying in vertical direction

For formation flying in the vertical direction, we apply almost the same control algorithm as that for formation flying on the horizontal plane. The difference is that the control algorithm for formation flying in the vertical direction is based on a consensus-based algorithm for a second-order system because the quadrotor model in the vertical direction is expressed as a second-order system.

The control law for quadrotor i is given by

$$\tilde{T}_{total_i}(t) = -\sum_{j=1}^{N+1} a_{ij} \left[\sum_{k=0}^{1} \gamma_k (\hat{h}_i^{(k)} - \hat{h}_j^{(k)}) \right], \quad i \in \{1,2,\cdots,N\}, \tag{28}$$

$$\hat{h}_j^{(k)} = h_j^{(k)} - d_{h_j}^{(k)}, j \in \{1,2,\cdots,N+1\}, \quad k \in \{0,1\}, \tag{29}$$

where subscript $N+1$ denotes a leader and $\gamma_k \in \mathbb{R}$, $k \in \{0,1\}$, are positive control gains. Furthermore, $d_{h_j}^{(k)} \in \mathbb{R}$, $k \in \{0,1\}$, is the desired relative state between quadrotor j and the leader in the vertical direction. Note that a UAV must obtain information including the altitude, rate of climb, and desired relative states from the connected quadrotors.

The following theorem for the desired convergence is derived.

Theorem 2. *Suppose that a multi-UAV system comprises $N(\geq 1)$ quadrotors expressed as (2) and a leader, and that control protocol (28) is applied to each quadrotor. In addition, assumptions 1 and 2 are satisfied. Let $\lambda = \lambda_R + i\lambda_I \in \mathbb{C}$ be a nonzero eigenvalue of the graph Laplacian of a multi-UAV system. When the control gains γ_k for $k \in \{0,1\}$ are selected so as to satisfy condition (30) for the real eigenvalues, and when the control gains γ_k for $k \in \{0,1\}$ are selected so as to satisfy conditions (30) and (31) for the complex eigenvalues, then all states of the quadrotors in the vertical direction asymptotically converge to the desired states.*

$$\gamma_k > 0, \quad \forall k \in \{0,1\} \tag{30}$$

$$(\gamma_1^2 \lambda_R - \gamma_0)\lambda_I^2 + \gamma_1^2 \lambda_R^3 > 0 \tag{31}$$

Proof. By applying control protocol (28) to quadrotor i expressed as (2), and by using the same approach as the proof of Theorem 1, Theorem 2 can be proved. □

4 Simulation and experimental validation

In this section, we present simulation results to validate the performance of the proposed control algorithms. In addition, we present the experimental validation of the proposed control algorithm for application to a multi-UAV platform.

4.1 Simulation setup

We considered a group of three quadrotors and a leader as well as the network with a directed spanning tree, as shown in Fig. 2.

The leader flew in an elliptical trajectory at a constant altitude. The major axis was 50 m, minor axis was 40 m, and cycle was 400 s, where the initial position was [0 0 1.3] m. Each quadrotor kept a certain distance from the leader and a certain azimuth from the traveling direction of the leader. The relative distance on the horizontal plane between each quadrotor and the leader was 5 m, and the altitude gap between them was 0 m. The azimuth was the angle going counterclockwise from the traveling direction of the leader. The azimuths of the first, second, and third quadrotors were 0 deg, 90 deg, and 270 deg, respectively. The information on the desired relative positions was provided by the leader.

Two simulations were conducted to validate the proposed control algorithms. The difference between the two simulations was in the control gains that were applied to the control algorithms. In Case I, a set of control gains was selected so as to satisfy conditions (9)–(14) and (30)–(31), which would result in the desired convergence. On the other hand, in Case II, a set of control gains was selected so as to not satisfy these conditions, which would result in divergence. The control gains for the formation on the horizontal plane and in the vertical direction are shown in Table 3.

4.2 Experimental setup

We developed an experimental platform using the Parrot AR.Drone 2.0, a small commercial UAV. The AR.Drone is equipped with the following sensors: a three-axis accelerometer, a three-axis gyroscope, a three-axis magnetometer, a sonar sensor, a pressure sensor, and front and vertical cameras. The data measured from the sensors and other information on the states of the AR.Drone can be obtained in real

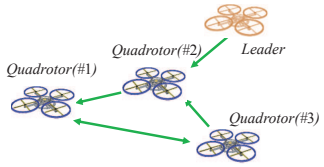

Fig. 2. Network structure for simulations and experimental validation.

Table 2. Control gains.

Control gains	Value
β_0, β_1, β_2, β_3 (for convergence)	1.0, 5.0, 20, 3.0
β_0, β_1, β_2, β_3 (for divergence)	1.0, 5.0, 20, 0.7
γ_0, γ_1 (for convergence)	2.9, 3.9
γ_0, γ_1 (for divergence)	1.0, 0.13

time. The AR.Drones cannot communicate with each other wirelessly; they do so through a control device via an access point, as shown in Fig. 3. Therefore, data can be sent wirelessly by all AR.Drones to a control device such as a laptop in real time. However, the AR.Drone has several restrictions that affected our experiments: it cannot calculate user commands on its onboard embedded computer; it can only be controlled with attitude angle commands or a rate of climb command; and it is not equipped with sensors that can acquire the horizontal position precisely enough for experimental validation. With regard to the first restriction, the laptop was used to calculate all commands for each AR.Drone from the measured data. With regard to the second restriction, the proposed control law outputs acceleration or moment commands, not attitude angle commands or a rate of climb command. Therefore, the commands from the proposed control law should be converted into equivalent commands that the AR.Drone can accept.

Because of the third restriction and the safety considerations of the experiments, only an experiment to validate the formation control algorithm in the vertical direction with control gains for convergence was conducted in this study. The control gains and other parameters in the experiment were the same as those in the simulations.

4.3 Simulation and experiment results

We present the simulation and experimental results here. Figs. 4 and 5 show the simulation results for Case I and Figs. 7– 9, those for Case II. Fig. 6 shows the simulation and experimental results for Case I.

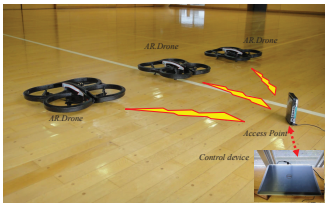

Fig. 3. AR.Drone: control structure.

Figs. 4 and 7 show the trajectories of the quadrotors and the leader on the horizontal plane and their positions every 10 s. Figs. 5 and 8 show the difference from the desired position on the horizontal plane. Fig. 6 shows the altitude and rate of descent. In Fig. 6, the solid lines illustrate the experimental results, and the dotted line illustrates the simulation results conducted under the same conditions as the experiments.

Figs. 4 and 5 show that each quadrotor converged to the desired position for formation flying on the horizontal plane when the proposed formation control algorithm on the horizontal plane (7) with the control gains that satisfy the conditions was applied to each quadrotor. On the other hand, Figs. 7 and 8 show that each quadrotor was in divergence when the proposed formation control algorithm on the horizontal plane (7) with the control gains that did not satisfy the conditions was applied to each quadrotor. These results validate the control strategy for formation flying on the horizontal plane and the conditions for the control gains.

Fig. 6 shows that the experimental results corresponded with the simulation results and that each AR.Drone was able to fly in formation at the desired altitude of 1.3 m when the proposed formation control algorithm in the vertical direction (28) with the control gains that satisfy the conditions was applied to each quadrotor. On the other hand, Fig. 9 shows that each quadrotor was in divergence when the proposed formation control algorithm in the vertical direction (28) with the control gains that did not satisfy the conditions was applied to each quadrotor. These results validate the control strategy for formation flying in the vertical direction and the conditions for the control gains.

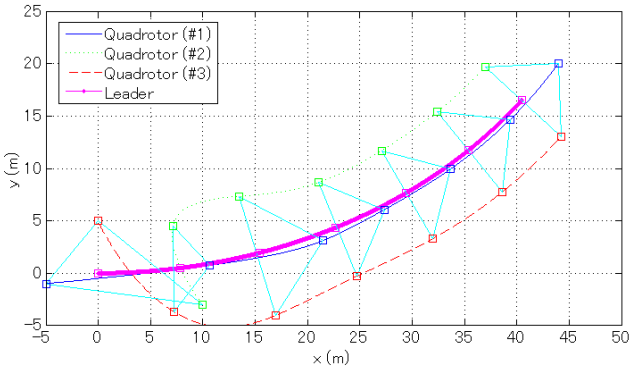

Fig. 4. Trajectory on horizontal plane using formation control algorithm (7) with control gains for convergence (Case I).

5 Conclusions

We have briefly demonstrated how to express a linearized model of quadrotors and how to model a multi-UAV system mathematically using graph theory and then provided formation control algorithms for UAVs to cooperatively fly

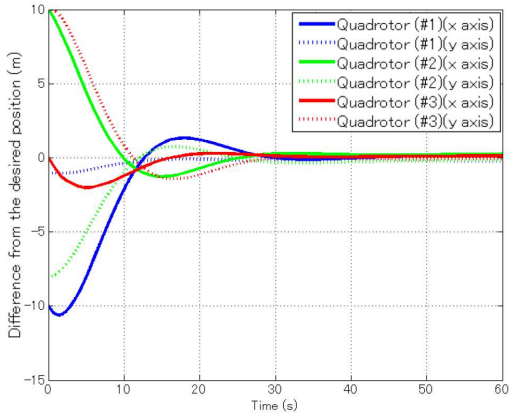

Fig. 5. Difference from the desired positions on horizontal plane using formation control algorithm (7) with control gains for convergence (Case I).

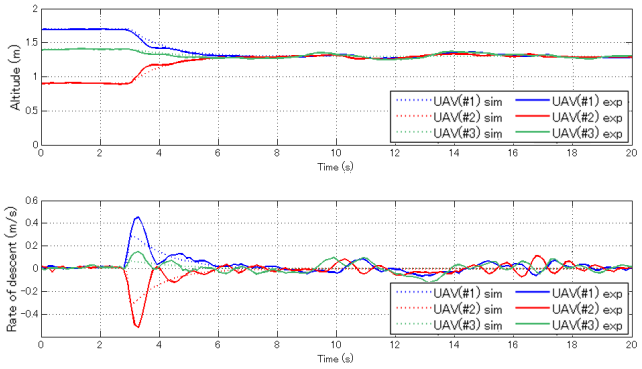

Fig. 6. Simulation and experimental results in vertical direction using formation control algorithm (28) with control gains for convergence (Case I).

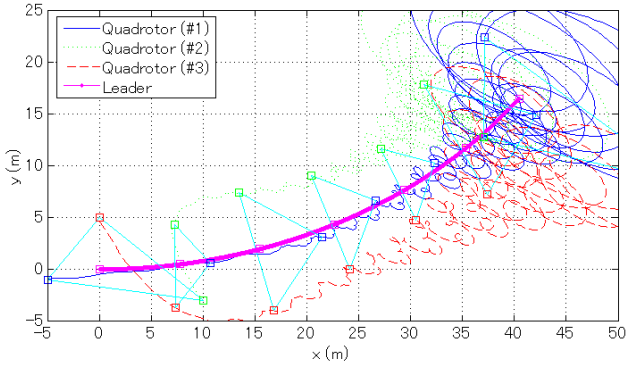

Fig. 7. Trajectory on horizontal plane using formation control algorithm (7) with control gains for divergence (Case II).

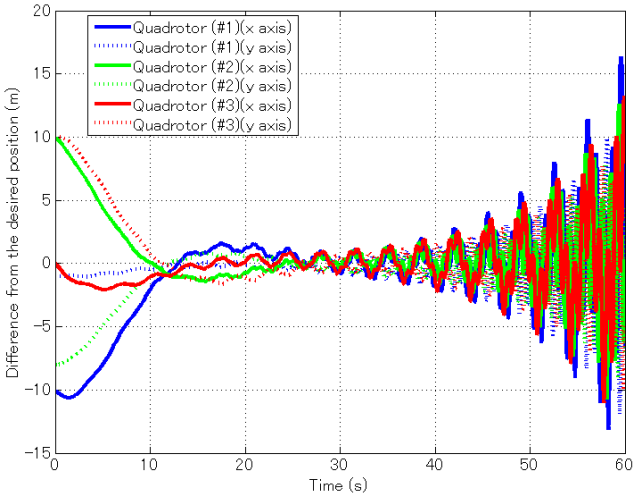

Fig. 8. Difference from the desired positions on horizontal plane using formation control algorithm (7) with control gains for divergence (Case II).

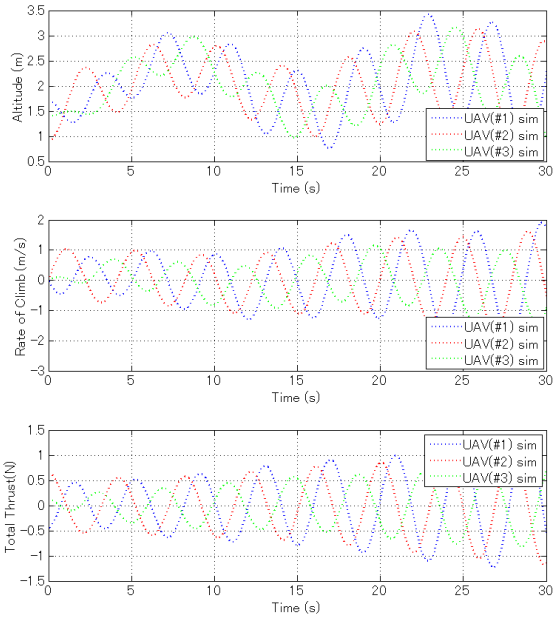

Fig. 9. Simulation results in vertical direction using formation control algorithm (28) with control gains for divergence (Case II).

in formation in three-dimensional space. Separate formation control algorithms were applied to the quadrotors for the horizontal plane and vertical direction. We showed that the stability discriminants of the multi-UAV system on the horizontal plane and in the vertical direction are expressed as polynomials. For the network structures composed of bidirectional or unidirectional network connections with the assumption that the network has a directed spanning tree, we provided conditions for formation control gains such that all the roots of the polynomials have negative real parts, that is, the UAVs can asymptotically converge to the positions for desired formation by using the generalized Routh stability criterion.

Our simulation and experimental results validated that the proposed algorithms are effective for formation flying under network structures with bidirectional or unidirectional connections. Furthermore, our experiments using three AR.Drones also validated the proposed algorithm's effective-

ness.

We developed an experimental platform using small commercial UAVs. Although the UAVs lacked enough functionality for full experimental validation, they were effective and economically efficient. We did not conduct experiments to validate the formation control algorithm on the horizontal plane because of a lack of a precise position sensor. Therefore, developing an experimental platform by using precise positions on the horizontal plane and conducting experiments to validate the formation control algorithm on the horizontal plane remain challenging issues.

References

[1] R. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, pp. 571–583, 2007.

[2] C.-Q. Ma and J.-F. Zhang, "Necessary and sufficient conditions for consensusability of linear multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1263–1268, 2010.

[3] X. Dong, Y. Zhou, Z. Ren, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles with switching interaction topologies," *Control Engineering Practice*, vol. 46, pp. 26–36, 2016.

[4] Z. Li, X. Liu, W. Ren, and L. Xie, "Distributed tracking control for linear multiagent systems with a leader of bounded unknown input," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 518–523, 2013.

[5] M. Turpin, N. Michael, and V. Kumar, "Trajectory design and control for aggressive formation flight with quadrotors," *Autonomous Robots*, vol. 33, no. 1-2, pp. 143–156, 2012.

[6] Z. Kan, T. Yucelen, E. Doucette, and E. Pasiliao, "A finite-time consensus framework over time-varying graph topologies with temporal constraints," *Journal of Dynamic Systems, Measurement, and Control*, 2017.

[7] J. Xi, Z. Shi, and Y. Zhong, "Consensus and consensualization of high-order swarm systems with time delays and external disturbances," *Journal of Dynamic Systems, Measurement, and Control*, vol. 134, no. 4, p. 041011, 2012.

[8] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1532, 2004.

[9] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," in *Proceedings of the IEEE*, pp. 215–233, 95(1, 2007.

[10] W. Ren, "Consensus tracking under directed interaction topologies: Algorithms and experiments," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 1, pp. 230–237, 2010.

[11] Z. Meng, W. Ren, Y. Cao, and Z. You, "Leaderless and leader-following consensus with communication and input delays under a directed network topology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 41, no. 1, pp. 75–88, 2011.

[12] P. Wieland and F. Allgöwer, "Growing optimally rigid formations," in *2012 American Control Conference*, pp. 3901–3906, 2012.

[13] A. Yang, W. Naeem, G. W. Irwin, and K. Li, "Stability analysis and implementation of a decentralized formation control strategy for unmanned vehicles," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 2, pp. 706–720, 2014.

[14] M. Porfiri, D. G. Roberson, and D. J. Stilwell, "Tracking and formation control of multiple autonomous agents: A two-level consensus approach," *Automatica*, vol. 43, no. 8, pp. 1318–1328, 2007.

[15] K. D. Listmann, M. V. Masalawala, and J. Adamy, "Consensus for formation control of nonholonomic mobile robots," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 3886–3891, IEEE, 2009.

[16] R. P. Anderson and D. Milutinović, "Stochastic optimal enhancement of distributed formation control using kalman smoothers," *Robotica*, vol. 32, no. 2, pp. 305–324, 2014.

[17] Y. Kuriki and T. Namerikawa, "Formation control of uavs with a fourth-order flight dynamics," in *Decision and Control (CDC), IEEE 52nd Annual Conference 2013*, pp. 6706–6711, 2013.

[18] Y. Kuriki and T. Namerikawa, "Consensus-based cooperative formation control with collision avoidance for a multi-uav system," in *2014 American Control Conference*, pp. 2077–2082, 2014.

[19] Y. Kuriki and T. Namerikawa, "Experimental validation of cooperative formation control with collision avoidance for a multi-uav system," in *Automation, Robotics and Applications (ICARA), 2015 6th International Conference on*, pp. 531–536, IEEE, 2015.

[20] C. Godsil and G. Royle, *Algebraic Graph Theory*. 2001: Springer, 2001.

[21] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE transactions on robotics and automation*, vol. 17, no. 6, pp. 947–951, 2001.

[22] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 3, pp. 2968–2973, IEEE, 2001.

[23] X. Xie, "Stable polynomials with complex coefficients," in *Decision and Control (CDC), IEEE 24th Annual Conference*, pp. 324–325, 1985.