APPLICATIONS OF
MODELLING AND SIMULATION
http://arqiipubl.com/ams eISSN 2600-8084 VOL 4, 2020, 159-170

Consensus Tracking Control of Multiple Quadrotors with a Cooperative Leader

Izzuddin Mat Lazim[1*], Abdul Rashid Husain[2], Liyana Ramli[1], Nurul Adilla Mohd Subha[2] and Mohd Ariffanan Mohd Basri[2]

[1]Faculty of Engineering and Built Environment, Universiti Sains Islam Malaysia, 71800 Negeri Sembilan, Malaysia
[2]School of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia

[*]Corresponding author: nh.izzuddin@usim.edu.my

Submitted 03 April 2020, Revised 22 April 2020, Accepted 23 April 2020.

Abstract: This paper discusses the design of a consensus tracking algorithm with the practical implementation of solving distributed altitude and attitude tracking of multiple quadrotors. In contrast to most of the existing works that consider a leader agent that is non-cooperative, this paper proposes an algorithm that allows the leader agent to receive feedback from a subset of the followers. Firstly, the feedback linearization technique is utilized on the nonlinear quadrotor dynamics which yield a simple linear system. Then, the proposed consensus tracking algorithm is designed and employed to the resulting linear system to achieve consensus tracking on altitude and attitudes via local interaction between neighbours. Results obtained show that the proposed consensus tracking for a group of quadrotors can improve connectivity preservation as compared to the non-cooperative leader in the presence of an obstacle. Based on the formulation of the controller, the methodology can be easily adapted to various systems such as attitude synchronization of multiple satellite systems.

Keywords: Consensus; Multi-agent Systems; Quadrotor.

1. INTRODUCTION

Recent technological advances and continuous development of miniaturized low-cost communication, computation, actuation, and sensing devices have accelerated research on cooperative control and enable deployment of multiple agents for various missions which previously done by a single expensive agent. A multi-agent system comprises of multiple autonomous subsystems interacting with each other via a communication network. These agents can be robots, vehicles, sensors, or process plants that work cooperatively to achieve certain tasks. Compared to the single-agent control, the cooperative control of multiple agents has advantages such as larger redundancy, higher robustness, and greater fault-tolerant [1]. Due to these advantages, cooperative control of multi-agent systems has potential applications such as in surveillance and reconnaissance systems, combat, hazardous material handling, and distributed reconfigurable sensor networks [2].

Issues that are being investigated under cooperative control include consensus control, formation control, rendezvous, flocking, and containment control. Among these subjects, consensus is the fundamental problem for cooperative control where it concerns the ability of agents to reach a common state based on local interaction with their neighbours via a sensing or communication network [3]. It can be categorized into two main classes: consensus regulator problem and consensus tracking problem. For consensus regulator problems, distributed controllers are implemented on each agent to drive them to reach a common value which depends on initial conditions [4]. Although consensus regulator is useful in applications such as vehicles rendezvous and clock synchronization, many applications require the agents to reach consensus with external references such as attitude synchronization of multiple satellite system. This type of consensus is known as consensus tracking, distributed tracking [5], or external consensus [6]. The two main competing objectives in consensus tracking are the convergence to a common state, as in consensus regulator, and the convergence of the common state to its desired reference state.

Most of the existing consensus tracking algorithms are developed by assuming the leader with zero input that generates the desired trajectory for the multi-agent systems [7]–[10]. This type of leader can generate signals which include step, ramp, and sinusoidal waveforms [11]. However, various situations require non-zero control input for the leader such as reaching a desirable water level in networked water tanks [6]. On the other hand, several studies assume that the control inputs or states of the leader are available to the followers [12]–[14]. This requirement has restricted system scalability as the communication bandwidth increases when the number of agents increases. In [15], consensus tracking with a non-cooperative leader was considered in which only followers act to achieve consensus. In [16], this algorithm has been extended for solving the formation control of quadrotors. The leader, in this case, is an active leader with an underlying dynamic. Nevertheless, it is more practical to consider a distributed consensus tracking with a cooperative leader, as the leader takes to account the followers' ability to follow its trajectory. For example, when a follower is blocked by an obstacle, the multi-agent system is less compromised due to the ability of the leader to get explicit feedback from the followers. In [6], consensus tracking of multiple water tanks with a cooperative leader is described. However, the algorithm proposed was based on transfer function dynamics which limits the applicability to systems that have zero initial condition and single-input-single-output case.

One of the potential applications of the distributed consensus tracking algorithm is on cooperative control of multiple quadrotors. The quadrotor platform is gaining popularity among researchers due to the advantages including omnidirectional, simple mechanical design, and safe to interact in close proximity [17]. Compared to the single quadrotor control problem, cooperative control for a group of quadrotors is more challenging due to the complexity of the dynamic interaction between subsystems (quadrotors), on top of the nonlinear and underactuated nature of each quadrotor. A common approach in realizing cooperative control of multiple quadrotors is by linearizing the nonlinear quadrotor model at an operating point [14], [18]. However, this approach does not guarantee the stability of the quadrotors outside of the operating point. Therefore, it is more desirable to implement nonlinear control of the quadrotor. In [16], a nonlinear control using feedback linearization technique which yields a linear decoupled quadrotor model was proposed. Then, a well-established linear controller was implemented on the multi-agent layer. This approach decouples the stability of the multi-agent layer and the local control of individual vehicles. Besides, the overall control design is simplified by separating the single quadrotor design with the cooperative control design, as opposed to the single-layer approach using nonlinear controllers such as backstepping [19], sliding mode control [20], and dynamic surface control [21].

Motivated by the above studies, we consider a distributed consensus algorithm with a cooperative leader for altitude and attitude tracking of multi-agent quadrotors. The contributions of this paper are as follows: (i) we consider a leader that has a non-zero input and able to take explicit feedback from a subset of the followers, instead of the leader that is non-cooperative [15], [16] or has zero input [7]–[9], [22]. The studies in [15], [16] directly deal with the problem of present interest. However, their proposed algorithms restrict the active leader from acquiring information from the followers. It is found that the distributed consensus tracking with a cooperative leader increases the connectivity preservation in the presence of an obstacle. (ii) We consider the agents that represented in general state-space form instead of the transfer function representation in [6]. This allows the algorithm to be applied to a wide range of systems that have multi-input-multi-output (MIMO) and nonzero initial values including the linearized model of the nonlinear equation and the integrator dynamic as a special case. (iii) The proposed algorithm relaxes the requirement of the reference signal by using a series of points instead of states form as presented by Ren [23].

The rest of the paper is organized as follows. Firstly, the nonlinear dynamics of a quadrotor and feedback linearization control of the dynamics are described. Then, the design of a consensus tracking algorithm for the linear multi-agent system is presented in detail. Next, the integration of the single quadrotor control and the multi-agent control is described, followed by the simulation results and discussion. The conclusion of the study is presented in the final part of the paper.

2. MODELLING AND CONTROL OF A QUADROTOR

2.1 Quadrotor Model

Consider the configuration of a quadrotor as shown in Figure 1, where (F_1, F_2, F_3, F_4) are the lift forces generated by the four rotors, while (p_x, p_y, p_z) and (ϕ, θ, ψ) denote the absolute position with respect to earth frame F_e, and orientation (roll, pitch, yaw) of the quadrotor, respectively.

By considering the translational and rotational components, the nonlinear model of a quadrotor is given by [16]

$$\ddot{p}_x = \frac{c(\phi)s(\theta)c(\psi) + s(\phi)s(\psi)}{M} u^{\{1\}} \tag{1}$$

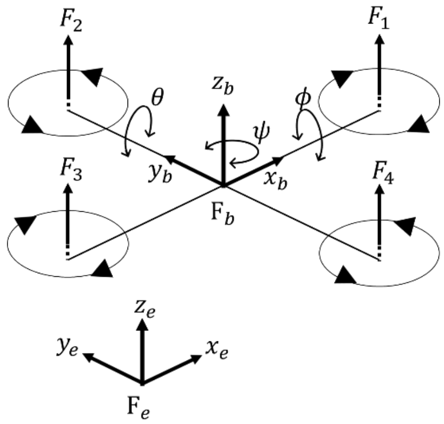

Figure 1. Configuration of a quadrotor with the earth fixed frame F_e and body fixed frame F_b

$$\ddot{p}_y = \frac{c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi)}{M} u^{\{1\}} \tag{2}$$

$$\dot{\mathbf{w}} = \mathrm{f} + \sum_{j=1}^{4} \mathrm{g}_j u^{\{j\}}, \quad \tilde{\mathrm{y}} = \mathrm{h}(\mathbf{w}) \in \mathbb{R}^m \tag{3}$$

where $\mathbf{w} = \left[p_z, \dot{p}_z, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}\right]^T \in \mathbb{R}^n$, $\mathrm{h}(\mathbf{w}) = [h_1, \dots, h_m]^T$, and

$$\mathrm{f} = \begin{bmatrix} \dot{p}_z \\ -a_g \\ \dot{\phi} \\ \frac{\dot{\theta}\dot{\psi}(I_y - I_z)}{I_x} \\ \dot{\theta} \\ \frac{\dot{\phi}\dot{\psi}(I_z - I_x)}{I_y} \\ \dot{\psi} \\ \frac{\dot{\phi}\dot{\theta}(I_x - I_y)}{I_z} \end{bmatrix}, \mathrm{g}_1 = \begin{bmatrix} 0 \\ \frac{c(\phi)c(\theta)}{M} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \mathrm{g}_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{I_x} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \mathrm{g}_3 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{I_y} \\ 0 \\ 0 \end{bmatrix}, \mathrm{g}_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{I_z} \end{bmatrix}$$

with abbreviations $s(\cdot) = \sin(\cdot)$, $c(\cdot) = \cos(\cdot)$. Meanwhile, $\mathbf{u} = \left[u^{\{1\}}, u^{\{2\}}, u^{\{3\}}, u^{\{4\}}\right]^T$, $I_\sigma(x, y, z)$, M and a_g are the control inputs of the system, the moment of inertia along each axis, the mass of quadrotor, and the gravitational acceleration, respectively. Notice that the quadrotor is a nonlinear system that poses a challenge in designing a control algorithm for the multi-agent layer. For the formulation of the consensus tracking algorithm, the quadrotor model is simplified into linear equations by utilizing a static feedback linearization technique.

2.2 Feedback Linearization Control of a Quadrotor

Feedback linearization is one of the common methods used in controlling nonlinear systems. Compared to the conventional linearization using the Jacobian approach that produces a linear representation of a system at an operating point only, the feedback linearization technique yields a linear model that is an exact representation of the original nonlinear model over a large set of operating points. It involves the transformation of nonlinear dynamics into an equivalent linear system via coordinate transformation and nonlinear state feedback.

Let $L_g^k h_j$ denotes the k-th Lie derivative of h_j along g, and $r^{\{j\}}$ ($j = 1, \dots, m$) is the relative degree of j-th output. The objective of the feedback linearization control in this paper to represent the nonlinear quadrotor dynamics into differential equations given as

$$\bar{\mathrm{y}} = \alpha(\mathbf{w}) + \beta(\mathbf{w})\mathbf{u} \tag{4}$$

where

$$\alpha(\mathbf{w}) = \begin{bmatrix} L_f^{r^{\{1\}}} h_1 \\ \vdots \\ L_f^{r^{\{4\}}} h_4 \end{bmatrix} \tag{5}$$

$$\beta(\mathbf{w}) = \begin{bmatrix} L_{g_1} L_f^{r^{\{1\}}} h_1(\mathbf{w}) & \cdots & L_{g_m} L_f^{r^{\{1\}}-1} h_1(\mathbf{w}) \\ \vdots & \ddots & \vdots \\ L_{g_1} L_f^{r^{\{m\}}} h_m(\mathbf{w}) & \cdots & L_{g_m} L_f^{r^{\{m\}}-1} h_m(\mathbf{w}) \end{bmatrix} \tag{6}$$

and $\bar{\mathrm{y}}$ is a vector including the output derivatives.

Lemma 1: [16] If the relative degrees $r^{\{k\}}$ that satisfy $\sum_{k=1}^{m} r^{\{k\}} = n$ and $\beta(\mathbf{w})$ is non-singular, then the system (3) can be transformed into an equivalent linear system using the following control law.

$$\mathbf{u} = \beta^{-1}(-\alpha + \mathbf{v}) \tag{7}$$

where $\mathbf{v} = \left[v^{\{1\}}, \dots, v^{\{m\}}\right]^T$ is the control input vector for the resulting linear system. By choosing the output $\tilde{\mathrm{y}} = [p_z, \phi, \theta, \psi]^T$, the relative degrees of the output are $r^{\{1\}} = r^{\{2\}} = r^{\{3\}} = r^{\{4\}} = 2$, thus satisfying $\sum_{k=1}^{4} r^{\{k\}} = n$. On the other hand, matrix

$\beta(\mathbf{w})$ is given as

$$\beta(\mathbf{w}) = \begin{bmatrix} c(\phi)c(\theta)/M & 0 & 0 & 0 \\ 0 & 1/I_x & 0 & 0 \\ 0 & 0 & 1/I_y & 0 \\ 0 & 0 & 0 & 1/I_z \end{bmatrix}$$

which is non-singular during the normal operation of the quadrotor, (i.e. $|\phi|, |\theta| < 45^o$). Based on Lemma 1, the dynamics in Equation (3) can be transformed into an equivalent linear system using Equation (7), given by

$$u^{\{1\}} = \frac{M\left(v^{\{1\}} + a_g\right)}{c(\phi)c(\theta)} \tag{8}$$

$$u^{\{2\}} = I_x v^{\{2\}} - \dot{\theta}\dot{\psi}\left(I_y - I_z\right) \tag{9}$$

$$u^{\{3\}} = I_y v^{\{3\}} - \dot{\phi}\dot{\psi}(I_z - I_x) \tag{10}$$

$$u^{\{4\}} = I_z v^{\{4\}} - \dot{\phi}\dot{\theta}\left(I_x - I_y\right) \tag{11}$$

Figure 2 shows the block diagram for the configuration of feedback linearization and quadrotor dynamics. Finally, the equivalent quadrotor's altitude and attitude dynamics can be written as

$$\ddot{p}_z = v^{\{1\}} \tag{12}$$
$$\ddot{\phi} = v^{\{2\}} \tag{13}$$
$$\ddot{\theta} = v^{\{3\}} \tag{14}$$
$$\ddot{\psi} = v^{\{4\}} \tag{15}$$

Using the feedback linearization technique, each quadrotor can now be treated as a linear decoupled sub-system. This allows the implementation of a linear consensus tracking algorithm on each input-output channel in the multi-agent layer.

3. CONSENSUS TRACKING OF MULTI-AGENT SYSTEM WITH A COOPERATIVE LEADER

3.1 Problem Formulation

Consider N identical agents modelled by the following linear dynamics.

$$\dot{\mathrm{x}}_i = \mathrm{A}_i \mathrm{x}_i + \mathrm{B}_i v_i, \quad y_i = \mathrm{C}_i \mathrm{x}_i, \quad i = 1,2,\dots,N \tag{16}$$

where $\mathrm{x}_i \in \mathbb{R}^n$, $v_i \in \mathbb{R}^p$ and y_i are the state vector, control input and measured output of agent i respectively. Meanwhile $\mathrm{A}_i \in \mathbb{R}^{n\times n}$, $\mathrm{B}_i \in \mathbb{R}^{n\times p}$, $\mathrm{C}_i \in \mathbb{R}^{q\times n}$ are constant matrices. As the objective of this paper is for the altitude/attitude tracking of multiple quadrotors, x_i may represent the altitude and its rate, or attitudes with their angular rates, while v_i correspond to $v^{\{1\}}, v^{\{2\}}, v^{\{3\}}$ or $v^{\{4\}}$ in Equations (12) - (15) of the i-th quadrotor. Details of the implementation will be presented in the next section.

Without loss of generality, let the agent that receives an external reference signal, $r(t)$ be the leader and assigned as $i = 1$, while all other agents are followers. It is also assumed that the leader is active, which implies that its state keeps changing throughout the entire process. The leader's dynamics are identical to the followers' except that the input v_1 is subjected to a subset of followers' states and an external reference signal $r(t)$. Let $\mathrm{x} = [\mathrm{x}_1^T, \mathrm{x}_2^T, \dots, \mathrm{x}_N^T]^T$ and $\mathrm{v} = [v_1, v_2, \dots, v_N]^T$, then the dynamics for N agents can be written collectively using Kronecker product $\otimes$ given by

$$\dot{\mathrm{x}} = (\mathrm{I}_N \otimes \mathrm{A}_i)\mathrm{x} + (\mathrm{I}_N \otimes \mathrm{B}_i)\mathrm{v} \in \mathbb{R}^{N.n} \tag{17}$$

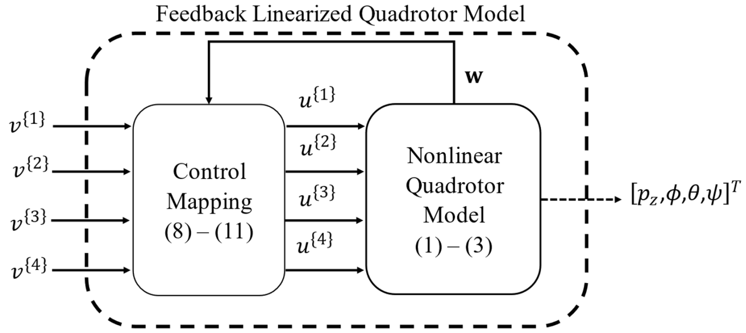

Figure 2. Configuration for the feedback linearization technique of nonlinear quadrotor model

with I_N denotes the $N \times N$ identity matrix. The interaction between agents is represented by a graph $\mathcal{G}$ that has a spanning tree. In other words, there is a directed path from the root node, which is the external reference signal, $r(t)$ to every agent. It is assumed that only a subset of the followers can observe the leader's states (e.g. position and orientation). This is practical in situations where the interaction between agents is limited by communication range and sensing ability.

This paper considers a cooperative leader where the leader also takes account of the followers' ability to follow its trajectory. Under this consideration, the interaction between agents and the external reference signal $r(t)$ can be represented using the following so-called cooperative-leader matrix:

$$\mathrm{H} = \mathrm{L} + \mathrm{Q} \in \mathbb{R}^{N \times N} \tag{18}$$

where L is the Laplacian matrix which defines the communication topologies between agents and $\mathrm{Q} = \mathrm{diag}(q_1, q_2, \dots, q_N)$ is the diagonal matrix of gains which describes the connection between the external reference signal $r(t)$ with the leader node. A brief overview of the Laplacian matrix is given in the appendix. If the node has access to the external reference, then $q_i > 0$, else $q_i = 0$. It is also assumed that only one leader exists in the group which is agent $i = 1$. Under this assumption, Q is given as

$$\mathrm{Q} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{19}$$

Lemma 2: [11] If the graph $\mathcal{G}$ contains a spanning tree with the root node being the reference signal *r(t)*, the cooperative-leader matrix H is non-singular and all its eigenvalues are in the open right-half plane.

3.2 Design of Consensus Tracking Algorithm with a Cooperative Leader

To ensure the states of the agents achieve consensus and converge to the given external reference input $r(t)$ simultaneously, a local distributed controller is proposed given as

$$v_i = \begin{cases} -K \sum_{j=2}^{N} a_{ij} \mathrm{x}_{ij} + F\xi_i + \varepsilon & i = 1 \\ -K \sum_{j=1}^{N} a_{ij} \mathrm{x}_{ij} + F\xi_i & i = 2,3, \dots, N \end{cases} \tag{20}$$

Here, $\mathrm{x}_{ij} = \mathrm{x}_i - \mathrm{x}_j$ and $\varepsilon = Gr - \mathrm{K}\mathrm{x}_i$, where $\mathrm{K} = [K_1, K_2, \dots, K_N]$ is a constant feedback gain vector, while G and F are constant gain to be designed. ξ_i is the integral of the tracking error given as

$$\xi_i = \begin{cases} -\int_0^t \left\{ \sum_{j=2}^{N} a_{ij} y_{ij} + y_{ir} \right\} dt & i = 1 \\ -\int_0^t \left\{ \sum_{j=1}^{N} a_{ij} y_{ij} \right\} dt & i = 2,3, \dots, N \end{cases} \tag{21}$$

where $y_{ij} = y_i - y_j$ and $y_{ir} = y_i - r(t)$. If the quadrotor i receives the state of neighboring quadrotor j, then $a_{ij} = 1$, otherwise $a_{ij} = 0$. The integral terms for the output error between agents in Equation (21) are added to ensure the multi-agent system achieves zero steady-state consensus, or $\lim_{t \to \infty} \left(y_i(t) - y_j(t) \right) = 0$. Notice that the leader ($i = 1$) is also influenced by the error between its output and the reference signal $r(t)$. This guarantees that the leader converges to the desired consensus value $r(t)$, or $\lim_{t \to \infty} \left(y_i(t) - r(t) \right) = 0$. By noting that $y_i = \mathrm{C}_i \mathrm{x}_i$, Equation (20) can be written collectively as

$$\begin{aligned} \mathrm{v} &= -(\mathrm{I}_N \otimes \mathrm{K})(\mathrm{H} \otimes \mathrm{I}_n)\mathrm{x} + G[1, \mathbf{0}]^T r - F \int_0^t (\mathrm{I}_N \otimes \mathrm{C}_i)(\mathrm{H} \otimes \mathrm{I}_n)\mathrm{x} - [1, \mathbf{0}]^T r dt \\ &= -(\mathrm{H} \otimes \mathrm{K})\mathrm{x} + G[1, \mathbf{0}]^T r - F \int_0^t (\mathrm{H} \otimes \mathrm{C}_i)\mathrm{x} - [1, \mathbf{0}]^T r dt \end{aligned} \tag{22}$$

where I_n denotes the $n \times n$ identity matrix, and $\mathbf{0}$ denotes a zero matrix with an appropriate dimension. Let $\boldsymbol{\xi} = [\xi_1, \xi_2, \dots, \xi_N]^T$, $\mathrm{X} = [x^T, \boldsymbol{\xi}^T]^T \in \mathbb{R}^{N.n+N}$, $\mathrm{p}_1 = G[B_1^T, \mathbf{0}]^T \in \mathbb{R}^{N.n}$ and $\mathrm{p}_2 = [1, \mathbf{0}]^T \in \mathbb{R}^N$, then the augmented representation for the closed-loop multi-agent system by substituting Equation (22) into Equation (17) yields

$$\begin{aligned}\mathrm{X} &= \begin{bmatrix}\mathrm{m}_{11} & \mathrm{m}_{12}\\ \mathrm{m}_{21} & \mathbf{0}\end{bmatrix}\mathrm{X} + \begin{bmatrix}\mathrm{p}_1\\ \mathrm{p}_2\end{bmatrix} r \\ &= \mathrm{MX} + \mathrm{P}r \in \mathbb{R}^{N.n+N}\end{aligned} \tag{23}$$

where

$$\begin{aligned}\mathrm{m}_{11} &= \mathrm{I}_N \otimes \mathrm{A}_i - (\mathrm{H} \otimes \mathrm{B}_i \mathrm{K})\\ \mathrm{m}_{12} &= F(\mathrm{I}_N \otimes \mathrm{B}_i)\\ \mathrm{m}_{21} &= -(\mathrm{H} \otimes \mathrm{C}_i)\end{aligned}$$

Proposition 1: Let λ_i $(i = 1,2, \dots, N)$ be the eigenvalues of H. Then, the system in Equation (23) is asymptotically stable and achieve consensus tracking if and only if all the matrices

$$\begin{bmatrix}\mathrm{A}_i - \lambda_i \mathrm{B}_i \mathrm{K} & F\mathrm{B}_i\\ -\lambda_i \mathrm{C}_i & 0\end{bmatrix} \in \mathbb{R}^{n\times n}, \qquad i = 1,2, \dots, N \tag{24}$$

are asymptotically stable.

Proof: There exists a non-singular matrix $\varpi \in \mathbb{R}^{N\times N}$, such that

$$\mathrm{S} = \varpi^{-1}\mathrm{H}\varpi = \begin{bmatrix}\lambda_1 & 0 & \cdots & 0\\ 0 & \lambda_2 & \ddots & \vdots\\ \vdots & \ddots & \ddots & 0\\ 0 & \cdots & 0 & \lambda_N\end{bmatrix} \tag{25}$$

Similarity transformation of M is given as

$$\begin{aligned}\overline{\mathrm{M}} &= \hat{\varpi}^{-1}\mathrm{M}\hat{\varpi}\\ &= \begin{bmatrix}\mathrm{I}_N \otimes \mathrm{A}_i - (\mathrm{S} \otimes \mathrm{B}_i \mathrm{K}) & F(\mathrm{I}_N \otimes \mathrm{B}_i)\\ -(\mathrm{S} \otimes \mathrm{C}_i) & \mathbf{0}\end{bmatrix}\end{aligned} \tag{26}$$

with

$$\hat{\varpi} = \begin{bmatrix}\varpi \otimes \mathrm{I}_N & \mathbf{0}\\ \mathbf{0} & \varpi \otimes \mathrm{I}_n\end{bmatrix}$$

Notice that the entries of $\overline{\mathrm{M}}$ consists of diagonal sub-matrices. Under this condition, eigenvalues of $\overline{\mathrm{M}}$ are the union sets of eigenvalues for the following matrix

$$\overline{\mathrm{M}}_i = \begin{bmatrix}\mathrm{A}_i - \lambda_i \mathrm{B}_i \mathrm{K} & F\mathrm{B}_i\\ -\lambda_i \mathrm{C}_i & 0\end{bmatrix} \tag{27}$$

$\overline{\mathrm{M}}$ is Hurwitz if and only if all $\overline{\mathrm{M}}_i$ in Equation (27) are Hurwitz. Hence under this condition, M is Hurwitz as the eigenvalues of M and $\overline{\mathrm{M}}$ are equal. By Lemma 2, if the graph $\mathcal{G}$ has a spanning tree with the reference signal $r(t)$ as the root node, then all eigenvalues λ_i of H have positive real parts. Thus, a proper design of gains (F and K) to satisfy condition in Equation (24) guarantees the consensus among agents and reference tracking of the leader. This completes the proof. ■

Value of the gains which make $\overline{\mathrm{M}}_i$ in Equation (27) Hurwitz for all i can be obtained by considering the smallest real eigenvalue, $\min_{i=1,\dots,N} \mathrm{Re}(\lambda_i)$. For example, the gains can be obtained by using a simple pole placement method given as

$$(s-\mu_1)(s-\mu_2)\cdots(s-\mu_{n+1}) = \det\left(s\mathrm{I}_{n+1} - \begin{bmatrix}\mathrm{A}_i - \min\limits_{i=1,\dots,N}\mathrm{Re}(\lambda_i)\,\mathrm{B}_i\mathrm{K} & F\mathrm{B}_i\\ \min\limits_{i=1,\dots,N}\mathrm{Re}(\lambda_i)\,\mathrm{C}_i & \mathbf{0}\end{bmatrix}\right) \tag{28}$$

with $\mu_1, \dots, \mu_{n+1}$ are the desired pole locations, while $\mathrm{I}_{n+1} \in \mathbb{R}^{(n+1)\times(n+1)}$ is an identity matrix. The values of gains (F and K) are obtained by equating the like powers on both sides.

4. CONSENSUS TRACKING OF MULTIPLE QUADROTORS

This section describes the implementation of altitude and attitude tracking of multiple quadrotors using the proposed consensus tracking algorithm and the linearized quadrotor model. Firstly, the linear decoupled system in Equations (12) - (15) are rewritten, respectively, in state-space form as

$$\dot{\bar{p}}_{z_i} = \mathrm{A}_i \bar{p}_{z_i} + \mathrm{B}_i v_i^{\{1\}} \tag{29}$$
$$\dot{\bar{\phi}}_i = \mathrm{A}_i \bar{\phi}_i + \mathrm{B}_i v_i^{\{2\}} \tag{30}$$
$$\dot{\bar{\theta}}_i = \mathrm{A}_i \bar{\theta}_i + \mathrm{B}_i v_i^{\{3\}} \tag{31}$$
$$\dot{\bar{\psi}}_i = \mathrm{A}_i \bar{\psi}_i + \mathrm{B}_i v_i^{\{4\}} \tag{32}$$

with $\bar{p}_{z_i} = \left[p_{z_i}, \dot{p}_{z_i}\right]^T$, $\bar{\phi}_i = \left[\phi_i, \dot{\phi}_i\right]^T$, $\bar{\theta}_i = \left[\theta_i, \dot{\theta}_i\right]^T$, $\bar{\psi}_i = \left[\psi_i, \dot{\psi}_i\right]^T$, $\mathrm{A}_i = [0\ 1; 0\ 0]$ and $\mathrm{B}_i = [0\ 1]^T$. Note that the subscript i is added to indicate that the quantities are associated with i-th quadrotor. The control inputs $v_i^{\{j\}}$ $(j = 1, \ldots, 4)$ is obtained using Equation (20), which can be written collectively as in Equation (22) given as

$$\mathrm{V}^{\{1\}} = -(\mathrm{H} \otimes \mathrm{K})\tilde{\mathrm{p}}_z + G[1, \mathbf{0}]^T r_z - F \int_0^t (\mathrm{H} \otimes \mathrm{C}_i)\tilde{p}_z - [1, \mathbf{0}]^T r_z dt \tag{33}$$
$$\mathrm{V}^{\{2\}} = -(\mathrm{H} \otimes \mathrm{K})\tilde{\phi} + G[1, \mathbf{0}]^T r_\phi - F \int_0^t (\mathrm{H} \otimes \mathrm{C}_i)\tilde{\phi} - [1, \mathbf{0}]^T r_\phi dt \tag{34}$$
$$\mathrm{V}^{\{3\}} = -(\mathrm{H} \otimes \mathrm{K})\tilde{\theta} + G[1, \mathbf{0}]^T r_\theta - F \int_0^t (\mathrm{H} \otimes \mathrm{C}_i)\tilde{\theta} - [1, \mathbf{0}]^T r_\theta dt \tag{35}$$
$$\mathrm{V}^{\{4\}} = -(\mathrm{H} \otimes \mathrm{K})\tilde{\psi} + G[1, \mathbf{0}]^T r_\psi - F \int_0^t (\mathrm{H} \otimes \mathrm{C}_i)\tilde{\psi} - [1, \mathbf{0}]^T r_\psi dt \tag{36}$$

with $\mathrm{V}^{\{1\}} = \left[v_1^{\{1\}}, v_2^{\{1\}}, \ldots, v_4^{\{1\}}\right]^T$, $\mathrm{V}^{\{2\}} = \left[v_1^{\{2\}}, v_2^{\{2\}}, \ldots, v_4^{\{2\}}\right]^T$, $\mathrm{V}^{\{3\}} = \left[v_1^{\{3\}}, v_2^{\{3\}}, \ldots, v_4^{\{3\}}\right]^T$, $\mathrm{V}^{\{4\}} = \left[v_1^{\{4\}}, v_2^{\{4\}}, \ldots, v_4^{\{4\}}\right]^T$, $\tilde{p}_z = \left[\bar{p}_{z_1}^T, \bar{p}_{z_2}^T, \ldots, \bar{p}_{z_N}^T\right]^T$, $\tilde{\phi} = [\bar{\phi}_1^T, \bar{\phi}_2^T, \ldots, \bar{\phi}_N^T]^T$, $\tilde{\theta} = [\bar{\theta}_1^T, \bar{\theta}_2^T, \ldots, \bar{\theta}_N^T]^T$, and $\tilde{\psi} = [\bar{\psi}_1^T, \bar{\psi}_2^T, \ldots, \bar{\psi}_N^T]^T$. The desired altitude and attitude (roll, pitch, yaw) are given respectively as r_z, r_ϕ, r_θ and r_ψ. Then, the value of gains K and F can be obtained by using Equation (28) where $\mathrm{C}_i = [1\ 0]$, and $G = K_1$. The overall block diagram developed for the simulation of multi-agent quadrotors is depicted in Figure 3.

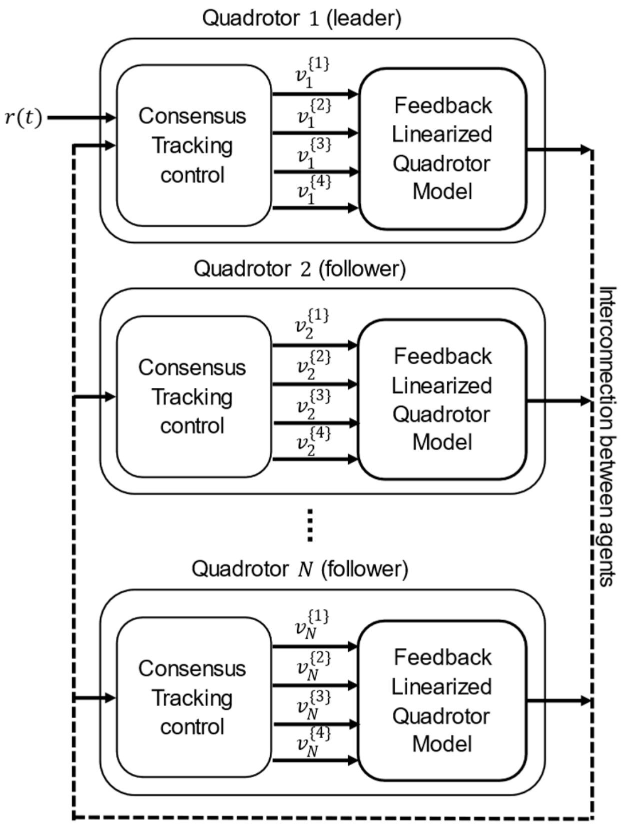

Figure 3. Block diagram for consensus tracking of multiple quadrotors

5. SIMULATION RESULTS AND DISCUSSION

In this section, we consider the altitude and attitude tracking problem of a five-quadrotor system to demonstrate the capability of the proposed consensus tracking algorithm. The quadrotor parameters are $\mathrm{M} = 0.6$ kg, $I_x = I_y = 0.005796\mathrm{kg.ms}^2$, $I_z = 0.010296\mathrm{kg.ms}^2$, and $a_g = 9.81\mathrm{ms}^{-2}$ [16]. Suppose that the local interaction topologies between quadrotors are illustrated in Figure 4, where only the leader ($i = 1$) has access to the desired reference signal $r(t)$. The incoming edges indicate that the quadrotors receive information (state) from the neighboring quadrotors. The difference between the two interaction topologies is emphasized in the dotted rectangle. Figure 4(a) corresponds to the interaction topology with a non-cooperative leader (no information flow from the followers to the leader), while Figure 4(b) correspond to the proposed cooperative leader (information flow from a subset of the followers to the leader exists). The cooperative-leader matrix H of the interaction topologies in Figure 4 (a) and (b) are given respectively as

$$\mathrm{H}_{(a)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad \mathrm{H}_{(b)} = \begin{bmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \tag{37}$$

Using the pole assignment method, the values of gains (K, F, and G) that give the desired poles as in Equation (28) with the interaction topology in Figure 4(b) at $\mu_1 = -0.0785 + 0.3938i, \mu_2 = -0.0785 - 0.3938i$ and $\mu_3 = -0.0050$ where $\min_{i=1,\ldots,N} \mathrm{Re}(\lambda_i) = 0.0810$ are $F = 0.01$, $\mathrm{K} = [2\ 2]$ and $G = 2$. We solve the consensus tracking problem by utilizing the proposed algorithm and present the following simulation results under different cases: Case 1 (without obstacle) and Case 2 (with an obstacle).

Case 1: Figure 5 - Figure 8 show the simulation results of the altitude and attitude tracking control of five quadrotors in the absence of obstacle with the interaction topology $\mathrm{H}_{(b)}$. The quadrotors are required to track different types of reference signals that are known only to the leader. Having different initial altitudes and attitudes, it can be seen from the figures that the quadrotors can track the reference signal via local communication exchange using the proposed consensus tracking algorithm in Equation (20). A similar result is obtained using the interaction topology $\mathrm{H}_{(a)}$.

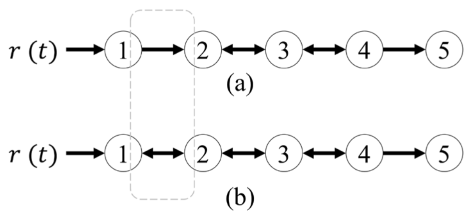

Figure 4. Interaction topologies between quadrotors with (a) non-cooperative leader, and (b) cooperative leader

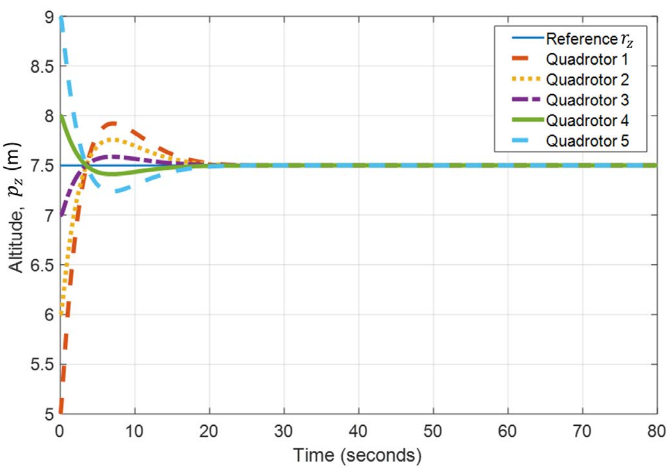

Figure 5. Altitude (p_z) tracking of multiple quadrotors

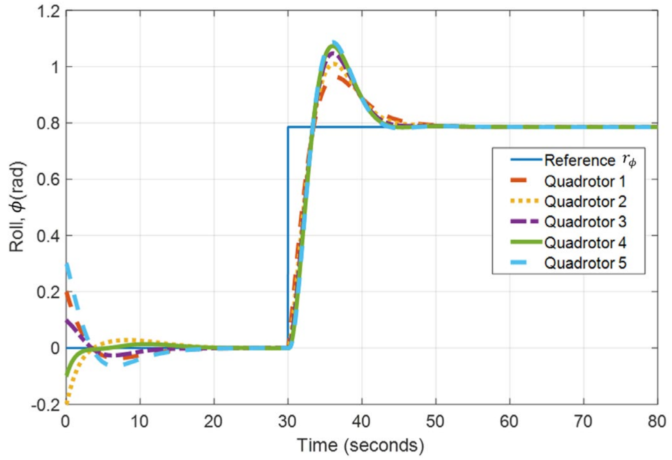

Figure 6. Attitude (ϕ) tracking of multiple quadrotors

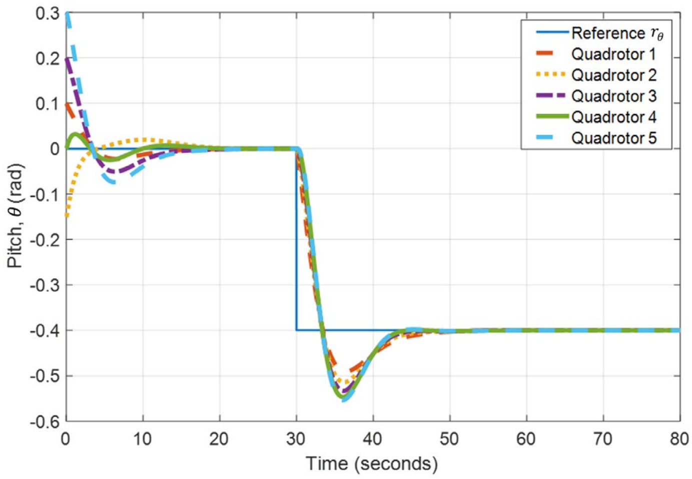

Figure 7. Attitude (θ) tracking of multiple quadrotors

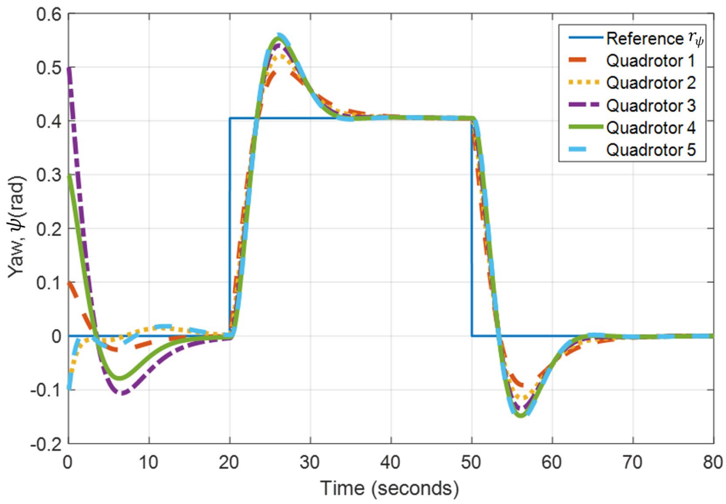

Figure 8. Attitude (ψ) tracking of multiple quadrotors

Case 2: Figure 9 and Figure 10 show the altitudes of quadrotor i at $t \in \{0,20,40,60,80,100,120,140,160\}s$ in the presence of a virtual obstacle with the interaction topologies given in $\mathrm{H}_{(a)}$ and $\mathrm{H}_{(b)}$, respectively. As a benchmark, the algorithm proposed in [15] with interaction topology $\mathrm{H}_{(a)}$ is adopted for the consensus tracking shown in Figure 9 which does not allow the leader to take explicit feedback from the followers (non-cooperative leader). On the other hand, Figure 10 shows the altitude tracking using our proposed algorithm in Equation (20) that allows the leader to cooperate with the follower(s) with the interaction topology $\mathrm{H}_{(b)}$.

The leader quadrotor ($i = 1$) receives an external reference signal given as $r_z(t) = 0.5t$. Suppose that the maximum communication range between quadrotors is four meters. If the quadrotors are separated by more than the maximum range, then the communication between quadrotors is disconnected. It is assumed that quadrotor five encountered a virtual obstacle at $t \in [50, 85]s$. To simulate the virtual obstacle effect, the state derivative of the quadrotor is forced to zero ($\dot{p}_{z_5} = \ddot{p}_{z_5} = 0$) between the period. With the interaction topology $\mathrm{H}_{(a)}$ and the algorithm proposed in [15], Figure 9 shows that quadrotors $i = 2,3,4$ are slowed down by quadrotor five that encountered the obstacle at $t \in [50, 85]s$. Since the leader quadrotor ($i = 1$) is non-cooperative and receives no feedback from the followers, it continued tracking the reference signal, $r_z(t)$ and causes a large separation between the quadrotors which eventually disconnect the communication between quadrotor four and five as shown starting from $t = 80s$.

On the other hand, Figure 10 shows that quadrotor $i = 2, 3,4$ are also slowed down by the quadrotor five due to the obstacle at $t \in [50, 85]s$. Owing to the ability to take feedback from the follower (quadrotor $i = 2$) as shown in Figure 4 (b), the leader quadrotor ($i = 1$) is also slowed down and kept the separation under the maximum communication range. Thus, the interactions between the quadrotors are preserved as shown at $t = 80s$ and allow the quadrotor $i = 2, 3,4$ to catch up the leader after the obstacle is gone as illustrated at $t \in \{100,120,140,160\}s$. This shows that having a leader that able to take feedback from the followers is useful as it can preserve the communication connection better than the non-cooperative leader type.

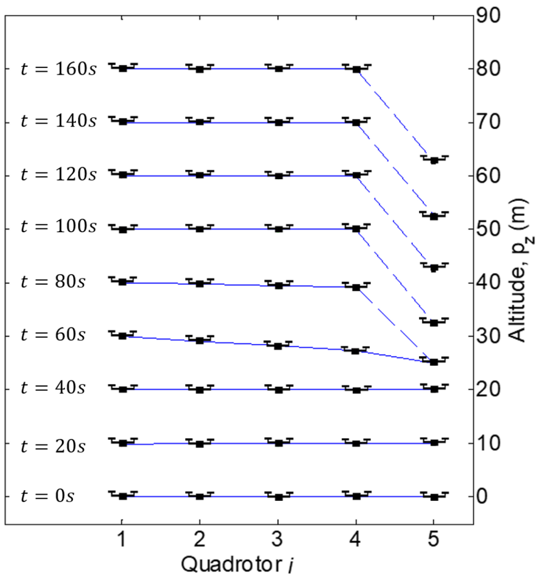

Figure 9. Altitude tracking of quadrotor i in the presence of an obstacle using the interaction topology $\mathrm{H}_{(a)}$ and consensus tracking algorithm proposed by [15]

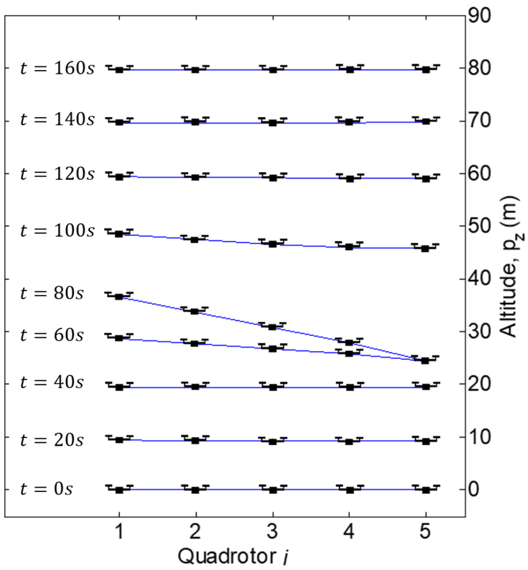

Figure 10. Altitude tracking of quadrotor i in the presence of an obstacle using the interaction topology $\mathrm{H}_{(b)}$ and our proposed consensus tracking algorithm in Equation (20)

6. CONCLUSION

In this paper, a distributed consensus tracking algorithm is proposed with application to altitude/attitude tracking of multi-agent quadrotors. The nonlinear mathematical model of the quadrotor is presented and linearized using feedback linearization technique which yields linear decoupled equations. Then, the proposed linear consensus tracking algorithm with a cooperative leader is described. The main contribution of this paper is on the derivation of consensus tracking algorithm with a leader that has a non-zero input and able to take explicit feedback from a subset of the followers, instead of the leader that is non-cooperative. From the simulation results, it can be concluded that having a cooperative leader improves the connectivity preservation of a multiagent system towards obstacles as compared to the non-cooperative leader. The proposed algorithm serves as the fundamental concept for distributed cooperative control before the extension to the formation control of multi-agent vehicles. Future research directions include the consideration of practical issues such as network delays, time-varying interaction topology, communication failure, and change of leaders.

REFERENCES

[1] X. Wang, Z. Zeng and Y. Cong, Multi-agent distributed coordination control: Developments and directions via graph viewpoint, *Neurocomputing*, 199, 2016, 204–218.

[2] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control: theory and applications*. London: Springer London, 2008.

[3] H. Chu and W. Zhang, Adaptive consensus tracking for linear multi-agent systems with input saturation, *Transactions of the Institute of Measurement and Control*, 38(12), 2016, 1434–1441.

[4] H. Zhang and F. L. Lewis, Adaptive cooperative tracking control of higher-order nonlinear systems with unknown dynamics, *Automatica*, 48(7), 2012, 1432–1439.

[5] Z. Li, G. Wen, Z. Duan and W. Ren, Designing fully distributed consensus protocols for linear multi-agent systems with directed graphs, *IEEE Transactions on Automatic Control*, 60(4), 4, 2015, 1152–1157.

[6] N. A. M. Subha and G. P. Liu, External consensus in multi-agent systems with large consecutive data loss under unreliable networks, *IET Control Theory and Applications*, 10, 2016, 989–1000.

[7] J. Wang, K. Chen and Q. Ma, Adaptive leader-following consensus of multi-Agent systems with unknown nonlinear dynamics, *Entropy*, 16(9), 2014, 5020–5031.

[8] W. Liu, Q. Wu, S. Zhou and G. Yin, Leader-follower consensus control of multi-agent systems with extended Laplacian matrix, *The 27th Chinese Control and Decision Conference*, Qingdao, 2015, 5393–5397.

[9] J. Xiang, W. Wei and Y. Li, Synchronized output regulation of linear networked systems, *IEEE Transactions on Automatic Control*, 54(6), 2009, 1336–1341.

[10] H. Chu, Y. Cai and W. Zhang, Consensus tracking for multi-agent systems with directed graph via distributed adaptive protocol, *Neurocomputing*, 166, 2015, 8–13.

[11] F. L. Lewis, H. Zhang, K. Hengster-Movric and A. Das, *Cooperative control of multi-agent systems: optimal and adaptive design approaches*, London: Springer-Verlag London, 2014.

[12] Z. Li, Z. Duan, G. Chen and L. Huang, Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint, *IEEE Transactions on Circuits and Systems*, 57(1), 2010, 213–224.

[13] Y. Hong, G. Chen and L. Bushnell, Distributed observers design for leader-following control of multi-agent networks, *Automatica*, 44(3), 2008, 846–850.

[14] J. A. Guerrero, P. Castillo, S. Salazar and R. Lozano, Mini rotorcraft flight formation control using bounded inputs, *Journal of Intelligence and Robotic Systems*, 65(1), 2012, 175–186.

[15] 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*, 58(2), 2013, 518–523.

[16] Mahmood and Y. Kim, Leader-following formation control of quadcopters with heading synchronization, *Aerospace Science and Technology*, 47, 2015, 68–74.

[17] M. A. Mohd Basri, A. R. Husain and K. a. Danapalasingam, Enhanced backstepping controller design with application to autonomous quadrotor unmanned aerial vehicle, *Journal of Intelligence and Robotic Systems*, 79(2), 2015, 295–321.

[18] Y. Kuriki and T. Namerikawa, Formation control of UAVs with a fourth-order flight dynamics, *IEEE Conference on Decision and Control*, Florence, 2013, 6706–6711.

[19] J. Ghommam, L. F. Luque-Vega, B. Castillo-Toledo, and M. Saad, Three-dimensional distributed tracking control for multiple quadrotor helicopters, *Journal of the Franklin Institute*, 353(10), 2016, 2344–2372.

[20] Y. Wang, Q. Wu, and Y. Wang, Distributed consensus protocols for coordinated control of multiple quadrotors under a directed topology, *IET Control Theory and Applications*, 7(14), 2013, 1780–1792.

[21] Y. Wang, Q. Wu, and Y. Wang, Distributed cooperative control for multiple quadrotor systems via dynamic surface control, *Nonlinear Dynamics*, 75(3), 2014, 513–527.

[22] X. Wang, Y. Hong, J. Huang, and Z. P. Jiang, A distributed control approach to a robust output regulation problem for multi-agent linear systems, *IEEE Transactions on Automatic Control*, 55(12), 2010, pp. 2891–2895.

[23] W. Ren, Multi-vehicle consensus with a time-varying reference state, *System Control Letters*, 56(7-8), 2007, 474–483.

APPENDIX

Local interaction between agents using Graph Theory

The interactions between N agents in a network can be represented using a simple graph $\mathcal{G}$. Agent i ($i = 1,2,\ldots,N$) is connected to the neighbouring agent j ($j \neq i$) by edges. The set of neighbouring agents that connects with agent i which are represented by incoming edges is denoted by $\mathcal{N}_i$. A commonly used mathematical tool in graph theory is the Laplacian matrix L defined as

$$\mathrm{L} = \left[l_{ij}\right]_{N\times N}, \;\; l_{ij} = \begin{cases} |\mathcal{N}_i| & \text{for } i = j \\ -1 & \text{for } j \in \mathcal{N}_i \\ 0 & \text{Others} \end{cases}$$

More descriptive explanation on graph theory concepts that are essential in the study of the multi-agent system can be found in the book by [11].