Research Article

Decentralized Control for Scalable Quadcopter Formations

Qasim Ali and Sergio Montenegro

Institute of Aerospace Information Technology, University of Würzburg, 97074 Würzburg, Germany

Correspondence should be addressed to Qasim Ali; qasim.ali@uni-wuerzburg.de

An innovative framework has been developed for teamwork of two quadcopter formations, each having its specified formation geometry, assigned task, and matching control scheme. Position control for quadcopters in one of the formations has been implemented through a Linear Quadratic Regulator Proportional Integral (LQR PI) control scheme based on explicit model following scheme. Quadcopters in the other formation are controlled through LQR PI servomechanism control scheme. These two control schemes are compared in terms of their performance and control effort. Both formations are commanded by respective ground stations through virtual leaders. Quadcopters in formations are able to track desired trajectories as well as hovering at desired points for selected time duration. In case of communication loss between ground station and any of the quadcopters, the neighboring quadcopter provides the command data, received from the ground station, to the affected unit. Proposed control schemes have been validated through extensive simulations using MATLAB®/Simulink® that provided favorable results.

1. Introduction

Mini Unmanned Aerial Vehicles (MUAVs) have drawn attention of scientific community from diversified backgrounds due to their versatile applications and obvious advantages over manned platforms. These applications range from simple toys found in supermarkets to highly sophisticated applications like offshore wind power station monitoring and so forth [1]. Remarkable advantages over manned platforms include their much lower manufacturing and operational costs, risk avoidance for human pilots, flying safely low and slow, and realization of operations that are beyond inherent human limitations. MUAVs are also more environment friendly as they cause less air pollution and noise. Recent research focuses on the use of multiple inexpensive vehicles flying together, while maintaining required relative separations, to carry out the tasks efficiently compared to a single exorbitant vehicle. Redundancy also does away with the risk of loss of a single whole-mission dependent vehicle. Some interesting applications in this domain include cooperative grasping, traffic monitoring, mobile communication relays, weather monitoring, and pesticide spraying. A group of MUAVs are also drawing a lot of attention for 3D aerial maps generation [2]. Unmanned is therefore unmatched. Performance requirements are high when flying in close proximity to each other. Although realization of safe operations for MUAV formations is challenging, the achieved advantages justify the laborious work involved. MUAVs are classified into two broad categories, namely, fixed wing and rotary wing aircraft (rotorcraft). Quadcopter is the most popular rotorcraft due to its agility, simplicity in design, ease of construction, ability to hover at desired points, and no requirement for a take-off/landing strip. However, control design is generally more complex. Quadcopter is the focus of our present paper.

The idea of formation flight was inspired by the nature-like bees and birds. Such behavior exhibited by the living species is characterized by self-propelled individuals following some *basic rules* for their collective motion and parallel actions. Our control strategy is also somewhat inspired by this natural behavior. For UAV swarm behavior, job of a control engineer is to implement *self-defined rules* through appropriate controller design for desired collective actions keeping in view the UAV dynamics. There may be different constellation architectures for a formation depending upon the mission requirements. Simple mathematical models to mimic the natural swarm behavior are generally based on the rules such as the following: (1) all individuals move in

the same direction, (2) individuals remain reasonably close to their neighbors, and (3) individuals avoid collisions with their neighbors. Some of the agents in a swarm may have superior sensing, computation, or communication abilities, called as leaders [3]. However, for such configurations, loss of the leader may result in collapse of the mission. Therefore, for our study, we have chosen a virtual leader strategy that promises mission accomplishment. In virtual leader approach, the vehicles in the formation jointly synthesize a single, possibly fictitious, leader vehicle whose trajectory acts as a leader for the group [4]. Note that a single leader may not efficiently control the large formations or swarms; hence multiple leaders with networked systems may be necessarily required; for example, we have exploited two virtual leaders for the proposed networked system.

A number of control techniques have been successfully employed for formation flight of UAVs. Some of the contributions are mentioned here. A distributed control scheme is suggested in [5], where units are not coupled rather making independent motions cooperatively. A cluster of UAVs has been used as a phased array antenna in [6] to show the feasibility of a distributed control strategy. Here each quadcopter, with a 2D model, has a local controller that is based on the information of its own states as well as states of a subset of other vehicles in the formation. A simple formation geometry is assumed in this paper, where the three units remain in a straight line maintaining a distance of 1 m from each other. Load transportation with multiple quadcopters is useful when the load is heavier compared with the maximum thrust of a single quadcopter or when additional redundancy is required for safety. Cooperative grasping and transportation were demonstrated, in addition to performing other stunning actions, with a cluster of autonomous micro-UAVs by GRASP laboratory at University of Pennsylvania [7]. An important approach to control design is *model following* where it is desired for the quadcopter to perform like an ideal model with desired flying qualities. Pitch pointing flight control laws have been designed in [8] by using the model following control scheme utilizing an eigenstructure assignment and Command Generator Tracker (CGT). In [9], the tracking control law was combined with an eigenstructure assignment and optimization technique to compute the feedback and feedforward gain matrices. The scheme was applied for pitch pointing control. Coordination and trajectory tracking control design for a leader-follower structure of multiple mini rotorcraft was simulated in [10] using nonlinear coordinated control design with state feedback. The scheme for leader-follower was extended to an arbitrary number of units in [11], and the network dynamics and manageability of swarms were analyzed based on open-loop H_2 norm of the network. Due focus has been given in [3] to the networks that are leader symmetric, restricted to the case when there is only one leader present. Graph theory has been exploited by a number of researchers in the domain of formation control, for example, using directed graphs [12] and UAV swarm modeling [11]. Problem is formulated while converting the graph into Laplacian matrix that also gives an insight, through its eigenvalues, to the communication topology and connectivity.

Formation control problem has been formulated in [13, 14] based on consensus approach. In [13], only one agent receives information from virtual leader. In [14] a time-varying formation is considered that may be useful for scenarios like rotation formation. Only one control input has been used in this study. Reference [15] controls a formation of three quadcopters using LQR as local controller and PI as formation controller. However, PI controller parameters are chosen experimentally, and the control laws adopted are sensitive to parameter changes. Fixed height of quadcopters is assumed in [14, 15].

The work in our present paper assumes the potential applications of photogrammetry and fire extinguishing through quadcopter fleets. Here consensus algorithms are not desired as we require the quadcopter fleets to track the trajectories of our interest, rather than those decided by the agents themselves. For our present paper, every agent in a formation receives information from virtual leader that adds to redundancy. Four control inputs to each quadcopter are used, and the controller designing has been described in much detail. The control scheme is robust and adaptive, and tracking performance validates the approach. Both formations track the trajectories with varying heights.

Portions of this work have been reported in our previous work [16]. However, for present paper, there are additional contributions like reference model that resides within the quadcopters to explicitly define the performance criteria for the quadcopters to make them behave like the model with desired dynamics. Here we have exploited a control strategy where each quadcopter's controller is based on its own states and the commanded values for position. This control architecture may be considered as *decentralized* where each agent is controlled by a local controller C_i which accesses the state of agent i with no information exchange among the other vehicles, as defined in [6]. Decentralized control has been exploited in other works, as in [17]. Main contribution of this paper is presentation of a simple framework to control a network of two quadcopter formations from ground stations exploiting the notion of virtual leader, so that loss of a leader does not affect the mission accomplishment. We have endeavored to tailor two control schemes, namely, Linear Quadratic Regulator Proportional Integral (LQR PI) based on explicit model following and LQR PI servomechanism to control two quadcopter formations. Here we consider a full-state vector of quadcopter while tracking only the outputs of interest (performance output). Implementation of this simple, adaptive, and robust scheme to control quadcopter formations from ground stations is not seen in the literature, as per the knowledge of authors. Performance of two control schemes has also been compared. LQR PI control scheme based on model following has also the potential for use even for some critical flight phases, like automatic flare control for smooth touch down, where the model dictates the desired trajectory.

Further description of this paper is organized into four sections. Architecture of suggested scheme is proposed in Section 2. The problem formulation is given in Section 3. Simulation results and analysis are reported in Section 4.
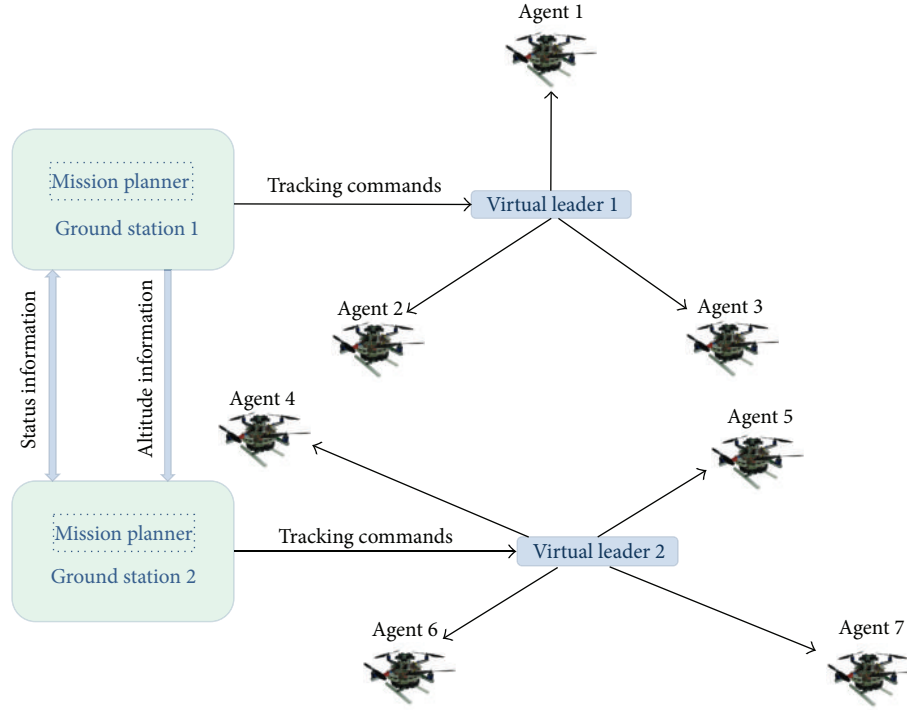
Figure 1: Architecture for two formations.

Finally conclusions are drawn and future work is envisaged in Section 5.

2. Architecture

The complete framework to control cluster of quadcopter formations for mission accomplishment is presented in this section. The objective, as an example, has been set to extinguish the fire over an area using two quadcopter formations with the defined roles of spraying the fluid and sending live data to the ground station, respectively. Main constituent parts include (1) ground stations, (2) mission planners, (3) virtual leaders (or virtual reference points), and (4) quadcopters with allied equipment (fire extinguisher or camera). The architecture with interconnections is shown in Figure 1.

We consider two formations comprising of three and four quadcopters with triangle and parallelogram geometric shapes, respectively. Parallelogram shape is chosen to cover maximum area underneath, though any arbitrary shape is possible with this scheme. The arrangement may be considered as *cluster of formations*. From now on, we will call triangle formation *formation 1* and parallelogram formation *formation 2*. Agents in formation 1 are numbered as 1–3 and those in formation 2 are labeled as 4–7 as shown in Figures 1, 3, and 4. Formation 1 is meant for aerial stereo-photography or videography that sends live data to the ground station 1. Though two quadcopters can meet the purpose, the third unit is added in formation 1 for redundancy purpose. The four quadcopters in formation 2 are equipped with fire extinguishers to spray the fluid on area under fire. This formation is

demonstrated, in Section 4, to make sinusoidal motions in X-Y plane as well for spraying the fluid at maximum area. Each formation is homogeneous as the agents are structurally identical within a formation. A pictorial view for the scenario is shown in Figure 2.

Each ground station encompasses a *mission planner*, whose role is to define the mission navigation legs in order to scan the area of interest. Tracking commands comprise 3D position. Reference signals are injected from ground station to the quadcopters in respective formation. These agents in the formation may be considered to follow a virtual leader. Agents are required to maintain customized separations from the respective virtual leader. Minimum distance between agents is to be considered to avoid collision and aerodynamic interactions. Two ground stations share the commanded altitude information to ensure safe height separation between the formations. These stations also act as standby to each other in case of malfunction of either unit and therefore continuously exchange the status information. Thus, it may be considered as one system of two subformations. The whole scheme has been simulated using the MATLAB/Simulink. Simulation results are presented in Section 4.

3. Problem Formulation

In this paper, we have assumed directed information flow to the agents. Communication topology of both formations is shown in Figures 3 and 4, respectively, where every node (or vertex) represents a quadcopter and every edge shows a communication link. Every agent receives command values from ground station to follow a virtual leader (VL).
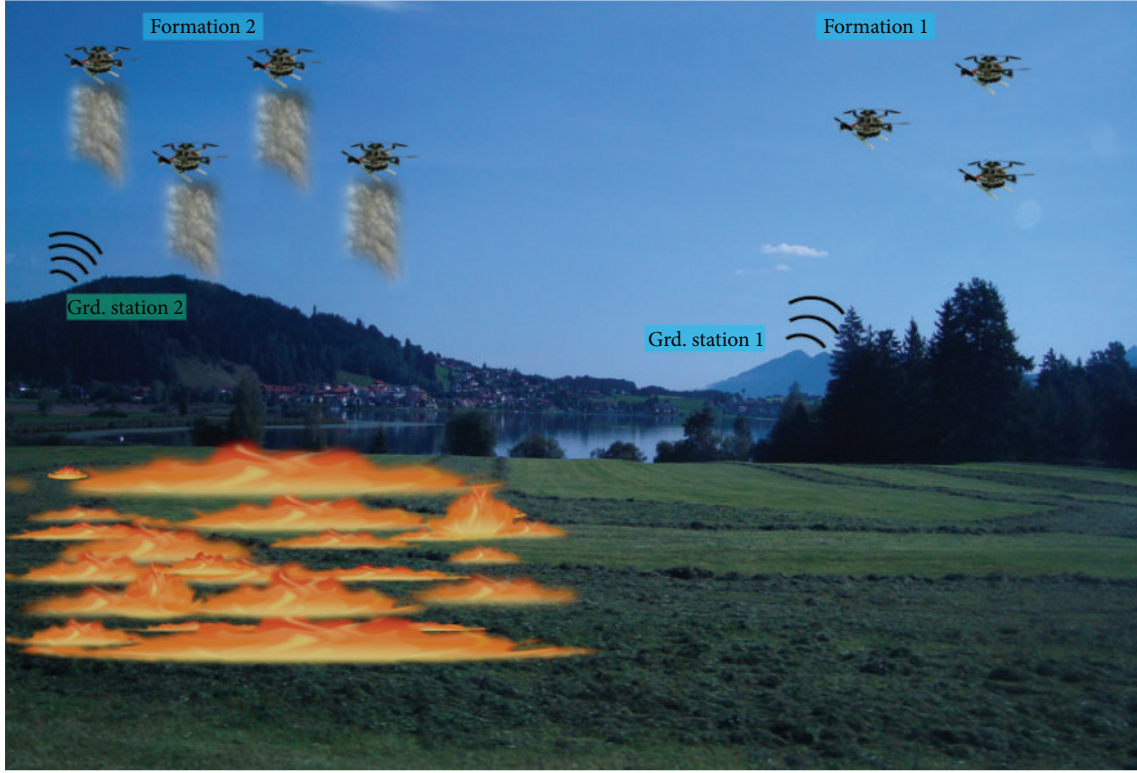
FIGURE 2: The mission scenario for fire extinguishment by quadcopter formations.
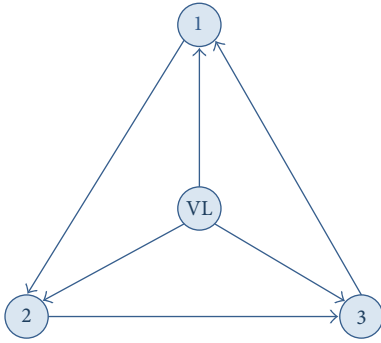

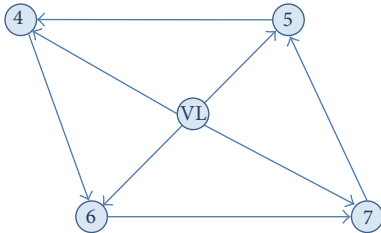FIGURE 3: Communication topology for formation 1.

FIGURE 4: Communication topology for formation 2.

In addition, one of the agents is redundantly providing the data, received from the ground station, to the neighboring agent to handle the individual communication link failure.

However, at one instant in time, only the signal from either virtual leader or neighboring unit is utilized in the control scheme.

Neighborhood of the agents is defined in terms of metric distance or topological distance. Neighborhood based on metric rules may be defined as follows.

Definition 1. A node j is a neighbor of node i if l_2 norm of displacement vector d between the two nodes is less than δ. Mathematically,

$$\|d\|_2 < \delta, \tag{1}$$

where δ is the range of communication media. On the other hand, *topological distance model* relies on surrounding agents which are immediate neighbors irrespective of their distances. Interactions are based on topological rules rather than metric rules. For this study, we have used the notion of neighborhood based on metric rules.

3.1. Quadcopter Dynamic Model. A quadcopter has two pairs of rotors in cross-configuration that can rotate at different angular velocities to achieve rotational and translational motions. Detailed motion dynamics of quadcopter are described in a number of publications, for example, [16]. For the sake of brevity, these are not repeated here. Quadcopter motion dynamics are illustrated with the help of Figure 5.
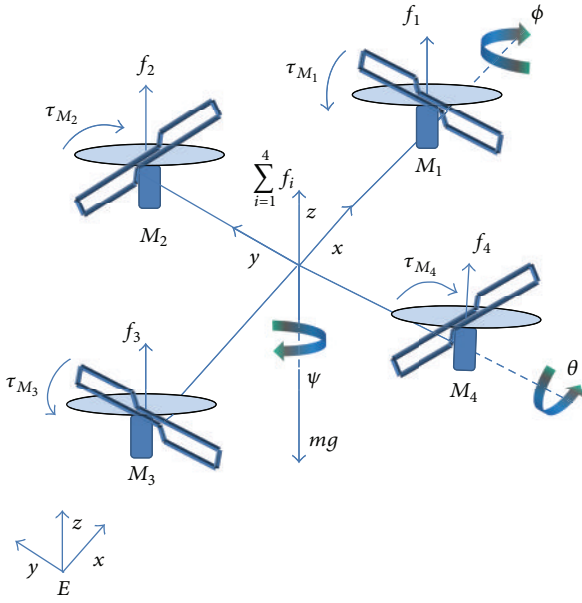
FIGURE 5: Quadcopter motion dynamics.

Quadcopter dynamics are modeled as per the following equations [18]:

$$u = f_1 + f_2 + f_3 + f_4,$$
$$f_i = k_i \omega_i^2, \quad i = 1, \ldots, 4,$$
$$m\ddot{x} = -u \sin \theta,$$
$$m\ddot{y} = u \cos \theta \sin \phi,$$
$$m\ddot{z} = u \cos \theta \cos \phi - mg, \quad (2)$$
$$\ddot{\psi} = \tau_\psi,$$
$$\ddot{\theta} = \tau_\theta,$$
$$\ddot{\phi} = \tau_\phi.$$

Here u shows the total thrust generated by four rotors, f_i is the force produced due to the rotation of rotor i, $k_i > 0$ is a constant, ω_i is the angular speed of motor i, m is mass of quadcopter, and g is gravitational constant. τ_ψ, τ_θ, and τ_ϕ represent the control inputs for yawing, pitching, and rolling moments, respectively. The above system (2) was linearized and corresponding Linear Time-Invariant (LTI) model of quadcopter given in [19] was used in this paper. Details for linearization and trim points may be viewed in [19] and are not included here for brevity. Quadcopter system under consideration has four inputs, twelve states, and six outputs. Our state vector comprises $[x \quad \dot{x} \quad y \quad \dot{y} \quad z \quad \dot{z} \quad \psi \quad \dot{\psi} \quad \theta \quad \dot{\theta} \quad \phi \quad \dot{\phi}]$. It represents the 3D position of center of mass of quadcopter in x-, y-, and z-direction relative to the earth-fixed frame E and Euler angles named as the yaw angle (ψ) around the z-axis, the pitch angle (θ) around the y-axis, and the roll angle (ϕ) around the x-axis, respectively, with their time derivatives denoted

with dot ($\cdot$) overhead. Output vector is $[x \quad y \quad z \quad \psi \quad \theta \quad \phi]$, and performance output vector is $[x \quad y \quad z]$. Our control vector is

$$\mho = [u_1 \quad u_2 \quad u_3 \quad u_2]^T = [u - mg \quad \tau_\psi \quad \tau_\theta \quad \tau_\phi]^T \quad (3)$$

that corresponds to throttle input, yawing, pitching, and rolling moments, respectively. We assume that mass of quadcopters (including peripherals) in formation 1 is 2 kg and that of each unit in formation 2 is 3 kg. As mass of *micro*-UAV is 0.1–0.5 kg [20], hence a new acronym of MUAV has been introduced in this paper to refer to *mini*-UAV.

3.2. Formation Dynamic Model. Desired dynamics of the reference model, defined as an LTI system, are as follows:

$$\dot{x}_r = A_r x_r + B_r r(t),$$
$$y_r = C_r x_r + D_r r(t). \quad (4)$$

Suffix r represents the reference model and q will represent the follower quadcopter plant. For the system (4), $x_r \in R^{n_r}$ are states of the reference model and $r(t)$ is the external bounded command value from ground station. For scheme implementation point of view, it may be considered coming from the virtual leader. $y_r \in R^{p_r}$ is the performance output vector of reference model. Reference model state space matrices (A_r, B_r, C_r, D_r) are of corresponding dimensions. For the reference model to exhibit stable dynamics, its state matrix A_r needs to be *Hurwitz*; that is, all its eigenvalues have strictly negative real part.

Quadcopter plant dynamics are described as follows:

$$\dot{x}_q = A_q x_q + B_q \mho,$$
$$y_q = C_q x_q + D_q \mho. \quad (5)$$

Here $x_q \in R^{n_q}$ are states of the quadcopter, $\mho \in R^{m_q}$ is the control vector, and $y_q \in R^{p_q}$ is the performance output vector of quadcopter with $p_q \leq m_q$ [21]. Here p_q and m_q indicate the number of outputs y_q and the number of control inputs $\mho$, respectively. The matrices (A_q, B_q, C_q, D_q) are of corresponding dimensions. With this scheme, maximum number of trackable outputs may not exceed number of control inputs (four in case of quadcopter). It is assumed that the matrix pair (A_q, B_q) is stabilizable and state vectors x_r and x_q are available as these are to be used for formulation of control signal $\mho$ for quadcopter.

Before proceeding to the controller design, we first do the controllability check for the quadcopters using the controllability matrix $P = \begin{bmatrix} B_q & A_q B_q & A_q^2 B_q & \cdots & A_q^{n_q-1} B_q \end{bmatrix}$ and find it to be full ranked. A reference model resides within each agent in formation 2 to define the ideal output response to be followed by that agent. All the closed-loop poles of reference model may be assigned to desired locations through pole placement method. Though reference model dynamics are manipulated here through pole placement method, the same may be realized through some other control scheme as well, for example, LQR. Reference model in formation 2 receives a known bounded command $r(t)$ from respective

ground station that may or may not vary with time. We are mainly interested to determine the control input $\mathrm{Ų}$ for the quadcopter such that the desired 3D formation geometry, given by the relative distance vector $r = \begin{bmatrix} r_x & r_y & r_z \end{bmatrix}^T$, is maintained. For this, an LQR controller with PI feedback connection [21] is implemented for the quadcopter. Design objectives in LQR scheme are defined through Q and R matrices. Q matrix, a positive semidefinite matrix, shows the weightage (or importance) to states. R matrix, a positive definite matrix, indicates weightage of control efforts corresponding to control inputs. The controller can be tuned by changing the elements in the Q and R matrices to achieve a desirable response. An efficient LQR control scheme is based on finding the right weighting factors. Model following LQR PI control scheme given in [21] has been exploited for our present study and tailored to control quadcopter formations from ground stations through virtual leaders. Open-loop dynamics of a quadcopter and reference model, from (5) and (4), may be formulated as

$$
\begin{bmatrix} \dot{x}_q \\ \dot{x}_r \end{bmatrix} = \begin{bmatrix} A_q & O_{n_q \times n_r} \\ O_{n_r \times n_q} & A_r \end{bmatrix} \begin{bmatrix} x_q \\ x_r \end{bmatrix} + \begin{bmatrix} B_q \\ O_{n_r \times m_q} \end{bmatrix} \mathrm{Ų}
$$
$$
+ \begin{bmatrix} O_{n_q \times m_q} \\ B_r \end{bmatrix} r(t). \tag{6}
$$

Equation (6) may be written more concisely as

$$
\dot{X} = AX + B\mathrm{Ų} + B_R r(t). \tag{7}
$$

Now tracking error Δy, from (4) and (5), may be written as

$$
\Delta y = y_q - y_r = \begin{bmatrix} C_q & -C_r \end{bmatrix} \begin{bmatrix} x_q \\ x_r \end{bmatrix} + D_q \mathrm{Ų} + (-D_r) r(t) \tag{8}
$$

or

$$
\Delta y = CX + D\mathrm{Ų} + D_R r(t). \tag{9}
$$

We aim to find such $\mathrm{Ų}$ that the tracking error asymptotically tends to zero in the presence of any known, bounded, and possibly time-varying command $r(t)$. In order to track a step-input command, an integral control is applied. The integrated tracking error may be represented as

$$
\dot{e}_y = \Delta y. \tag{10}
$$

Equations (7), (9), and (10) may be combined as

$$
\begin{bmatrix} \dot{e}_y \\ \dot{X} \end{bmatrix} = \begin{bmatrix} O_{p_q \times p_q} & C \\ O_{(n_q+n_r) \times p_q} & A \end{bmatrix} \begin{bmatrix} e_y \\ X \end{bmatrix} + \begin{bmatrix} D \\ B \end{bmatrix} \mathrm{Ų}
$$
$$
+ \begin{bmatrix} D_R \\ B_R \end{bmatrix} r(t), \tag{11}
$$

concisely written as

$$
\dot{\widetilde{X}} = \widetilde{A}\widetilde{X} + \widetilde{B}\mathrm{Ų} + \widetilde{B}_R r(t). \tag{12}
$$

3.3. Control Strategies.

It is assumed that external command $r(t)$ is a step-input command with zero errors. We now differentiate (12) with respect to time and introduce new variables $\mathrm{w} = \begin{bmatrix} \dot{e}_y & X \end{bmatrix}^T$ and $v = \dot{\mathrm{Ų}}$, which yields

$$
\dot{\mathrm{w}} = \widetilde{A}\mathrm{w} + \widetilde{B}v. \tag{13}
$$

We here assume that agent's complete state knowledge is available for implementation of LQR PI controller, which we use to control the open-loop dynamics of (13). We use the control input v to minimize the LQR cost. Cost function J may be defined as

$$
J = \int_0^\infty \left(\mathrm{w}^T Q \mathrm{w} + v^T R v \right) dt. \tag{14}
$$

Here Q and R are LQR weight matrices. Now we solve the Algebraic Riccati Equation (ARE)

$$
\widetilde{A}^T P + P\widetilde{A} + Q - P\widetilde{B}R^{-1}\widetilde{B}^T P = 0 \tag{15}
$$

and get the solution $P = P^T > 0$ that is used to compute the LQR gain matrix K given as

$$
K = R^{-1}\widetilde{B}^T P. \tag{16}
$$

The gain matrix K is of the following form:

$$
K = \begin{bmatrix} K_I^e & K_P^{x_q} & K_P^{x_r} \end{bmatrix}. \tag{17}
$$

Here subscript indicates the type of gain (proportional or integral) and superscript shows the variable to whom this gain matrix is applied. Now

$$
\dot{\mathrm{Ų}} = -\begin{bmatrix} K_I^e & K_P^{x_q} & K_P^{x_r} \end{bmatrix} \begin{bmatrix} \dot{e}_y \\ \dot{x}_q \\ \dot{x}_r \end{bmatrix}. \tag{18}
$$

Integrating (18) and ignoring constants of integration, LQR PI control solution for the quadcopter in formation 2 is given as

$$
\mathrm{Ų} = K_I^e \frac{(y_r - y_q)}{s} - K_P^{x_q} x_q - K_P^{x_r} x_r. \tag{19}
$$

LQR PI control scheme guarantees perfect tracking [16, 22]. Although our derivation is based on step reference command, the resulting control system gives good time response for any arbitrary reference command signal $r(t)$ [22], as will be demonstrated in simulation section. This command generator is capable of handling a wide range of motion trajectories, including position unit step commands, unit ramp commands, and oscillatory commands [23]. Figure 6 shows the interconnection diagram for LQR PI control scheme based on model following architecture. Here the reference model acts as a command prefilter and the quadcopter smoothly tracks the output of reference model. Dynamics of reference model and follower quadcopter are defined by (4) and (5), respectively.
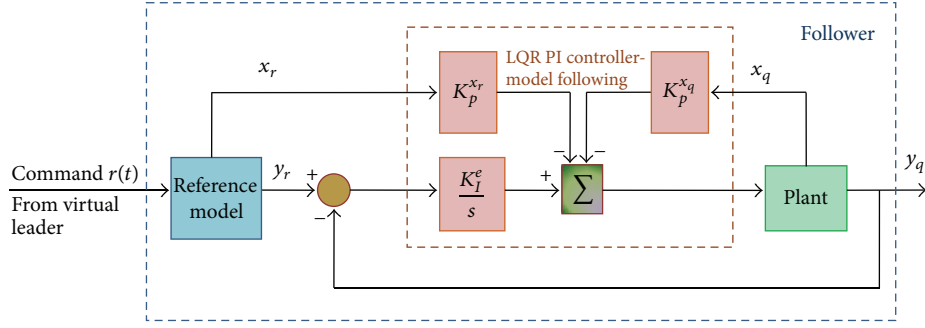
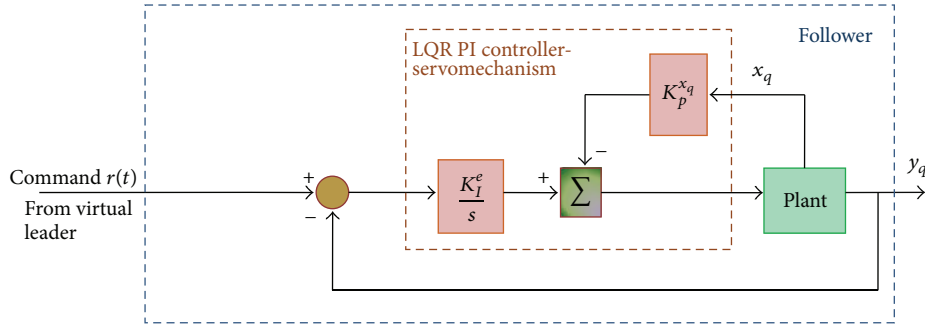FIGURE 6: LQR PI control scheme with reference model.

FIGURE 7: LQR PI servomechanism control scheme.

For some scenarios, we may wish for not to smooth out the command $r(t)$ through a reference model but rather to follow the command directly. For this, a modified version of the scheme shown in Figure 6 may be exploited, known as *LQR PI servomechanism*. This may be achieved by setting A_r, B_r, and C_r as null matrices and D_r as identity matrix. Then (4) yields $y_r = r(t)$. Block diagram showing interconnection for this control scheme is given in Figure 7. We need to consider that a system based on LQR PI servomechanism scheme is controllable as long as the following rank condition is true [21]:

$$\text{rank} \begin{bmatrix} A_q & B_q \\ C_q & D_q \end{bmatrix} = n_q + m_q. \qquad (20)$$

We tested the rank condition for our case and it came out to be 16, the full ranked. However, rank condition also implies that maximum number of regulated outputs may not exceed the number of control inputs to the system (four in the case of quadcopter).

4. Simulation Results and Analysis

4.1. Comparison of Control Schemes. Two types of command tracking schemes are implemented: *LQR PI control scheme* based on *reference model following* to track the commands quite smoothly and the *LQR PI servomechanism* for rapid command tracking. We have exploited both the schemes for two formations to meet the respective mission requirements. Formation 1 is managed through LQR PI servomechanism control scheme, which demonstrates quick response at
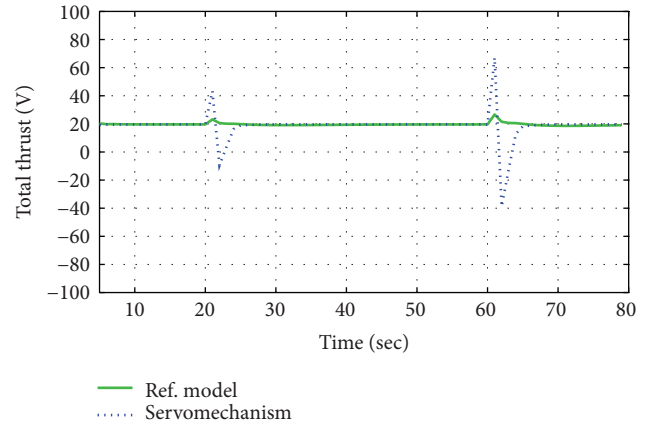

FIGURE 8: Comparison of control effort for two control schemes.

the cost of high control effort. This forces the agents to follow the desired trajectory vigorously as per the requirements of the mission, for example, photogrammetry. However, for Formation 2 we are rather interested in smooth movements so we implement LQR PI control scheme based on reference model following. Control efforts for both the control schemes are simulated in MATLAB environment for comparison, and the plot is shown in Figure 8. Tracking performance for two control schemes is demonstrated in Figure 9. For the sake of comparison and clarity in plots, only two agents, one from each formation, which receive the same tracking commands are considered here.
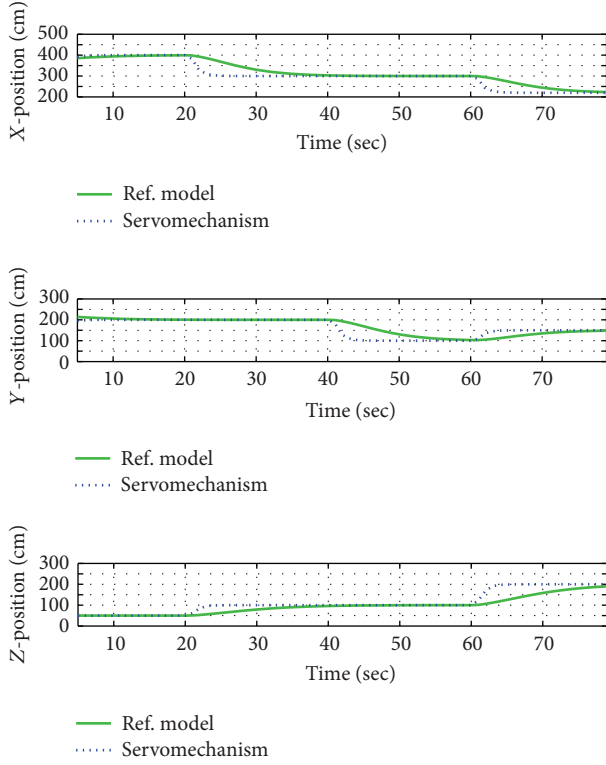
FIGURE 9: Comparison of tracking performance for two control schemes.

FIGURE 10: Quadcopters joining to make desired formation geometry.

FIGURE 11: Trajectory tracking by two formations.

4.2. Trajectory Tracking Performance. From safety point of view, most of the proposed algorithms are required to be simulated before actual flights may be undertaken. Simulations also help to authenticate the efficacy of the algorithm. The two types of control schemes defined above have been implemented in MATLAB/Simulink. The whole mission of fire extinguishment is divided into two phases. In phase 1 the agents join to form the required formation geometry (e.g., triangle or parallelogram), while in phase 2 the agents follow the trajectory commanded from the ground station through *mission planner*.

Commanded relative separation vectors for agents, defined as $r = [r_x, r_y, r_z]$, in formation 1 are $[-10, 0, 0]$, $[20, -20, 0]$, and $[20, 20, 0]$, while those in formation 2 are $[-50, 20, 0]$, $[30, 20, 0]$, $[-30, -20, 0]$, and $[50, -20, 0]$ with respect to their respective virtual leaders. In the first phase, agents depart from their initial arbitrary points in order to form the desired formation geometry. Simulation results to this effect for formation 2 are shown in Figure 10.

Tracking reference values are introduced from network operators to the quadcopters. Simulation results for the two formations in 3D view are shown in Figure 11. Initial and final positions of agents and corresponding virtual leaders (or references) are also shown to indicate the direction of motion. From the plots, it is evident that output of reference models is tracked quite smoothly by the agents in formation 2 while maintaining the desired mutual separations. The agents in formation 1 vigorously follow the tracking commands.
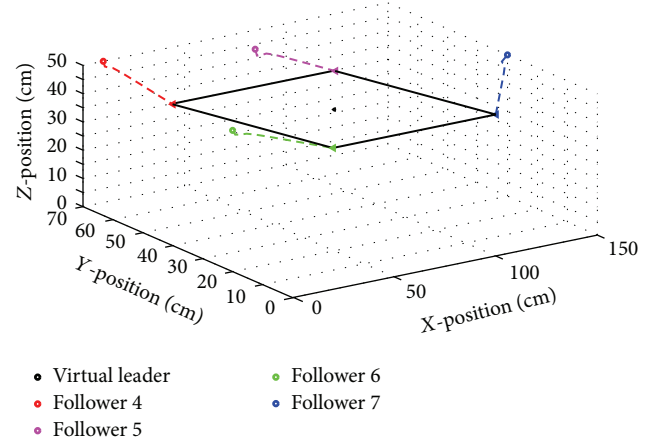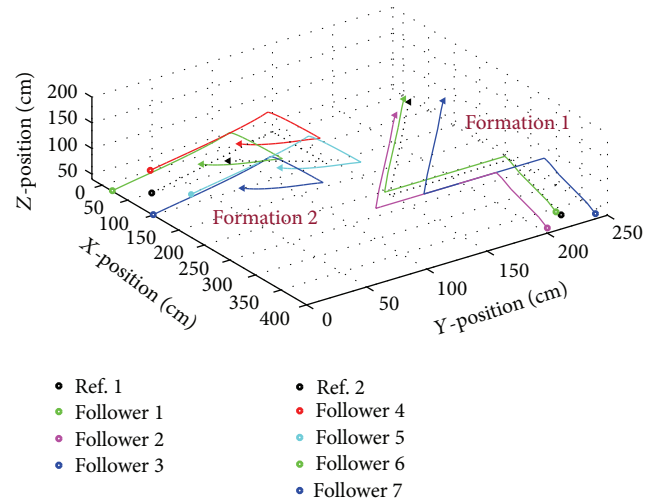
Now the tracking performance is demonstrated for sinusoidal command in order to cover maximum area underneath the agents carrying the fire extinguishers. Results to this effect are shown in Figure 12. However, for such commands, desired tracking (of same frequency) is achieved if rate of change of command signal is within the closed-loop system bandwidth [21]. Though we get the sinusoidal output but with a different amplitude and phase determined by the magnitude of the system transfer function. Desired amplitude may be achieved using a suitable precompensator. Thus, the control scheme is capable of variable set-point tracking as well. LQR PI control scheme based on model following is quite robust against different types of possible perturbations as shown in our previous study [16]. However, for the sake of brevity and clarity in plots, disturbances are not studied in this paper.

We now make the quadcopters to hover at desired location for specified time duration. Figure 13 shows formation 2 in hovering mode on a time scale, where the agents maintain their horizontal position.
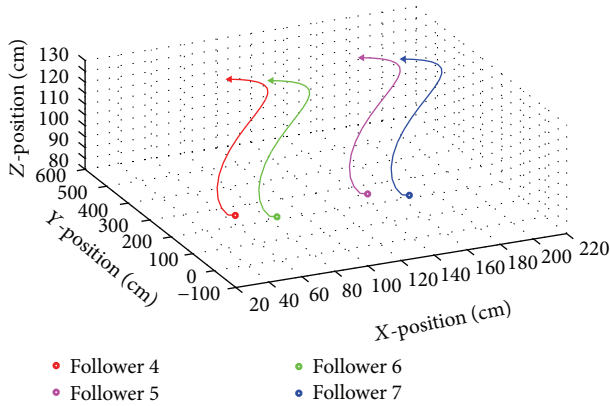
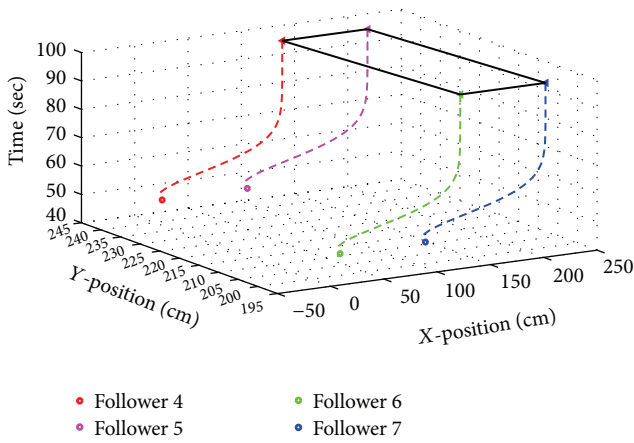FIGURE 12: Sinusoidal motion of formation 2.

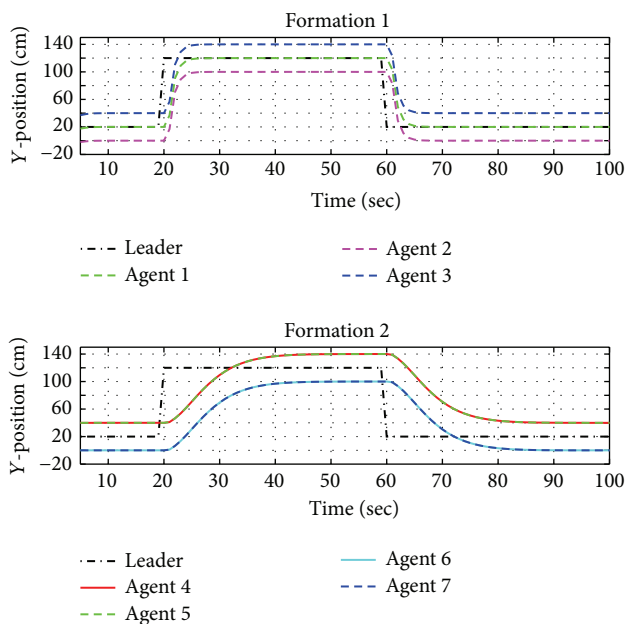FIGURE 13: Formation 2 in hovering mode.

FIGURE 14: Same trajectory tracking by two formations.

For further data comparison, two formations are commanded to track the same trajectory while maintaining their respective formation geometries and adhering to their respective control schemes. Simulation result for Y-position of both formation agents is plotted in Figure 14 that emphasizes the scalable property of the formation control system.

5. Conclusions and Future Works

Position controllers for two formations were implemented using the control schemes LQR PI based on model following and LQR PI servomechanism. These control schemes are compared in terms of convergence to desired tracking values and the control effort. LQR PI model following controller behaves well in terms of transient response and exhibits no overshoot, while LQR PI servomechanism reacts faster to the commands but at the cost of high control effort. With these two control schemes, quadcopters in formation are able to track the desired trajectory while maintaining defined relative 3D separations. Extensive simulations proved efficacy of the proposed schemes while giving promising results. Proposed architecture is scalable and can be expanded easily. Notion of virtual leader enables the scheme to be robust against any node failure. Load of information (command values) from ground station to the agents is quite low, as only position information is transmitted from ground station to the quadcopters. Working range of communication media needs to be considered that determines the maximum distance between ground station and the quadcopter formation.

The two presented control approaches for formation flights are simple from implementation point of view. However, it necessitates knowing the exact dynamic model and the states of aerial vehicles. As it is quite laborious to model all the dynamics of rotorcraft flying in close formation, adaptive and robust control techniques, like LQR PI, may be explored to their full potential. Advanced designs may be realized while introducing the network dynamics that is a new area of research.

Cluster reconfiguration of agents to change the formation geometry to any arbitrary shape during flight may also be undertaken in an efficient and simple way using the proposed scheme and is envisaged as a subsequent work. A suitable collision avoidance mechanism may also be introduced for safe operations. For future work, presented scheme may be implemented in real time using the formation flying test setup developed at Institute of Aerospace Information Technology, Würzburg University [24].

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

References

[1] Q. Ali, N. Gageik, and S. Montenegro, "A review on distributed control of cooperating mini UAVS," *International Journal of Artificial Intelligence & Applications*, vol. 5, no. 4, pp. 1–13, 2014.

[2] H. van Dyke Parunak, S. A. Brueckner, and J. J. Odell, "Swarming coordination of multiple UAV's for collaborative sensing," in *Proceedings of the 2nd AIAA 'Unmanned Unlimited' Conference and Workshop and Exhibit*, San Diego, Calif, USA, September 2003.

[3] S. Martini, M. Egerstedt, and A. Bicchi, "Controllability decompositions of networked systems through quotient graphs," in *Proceedings of the 47th IEEE Conference on Decision and Control (CDC '08)*, pp. 5244–5249, Cancun, Mexico, December 2008.

[4] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[5] R. Olfati-Saber and R. M. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *Proceedings of the 15th World Congress of the International Federation of Automatic Control (IFAC '02)*, pp. 242–248, Barcelona, Spain, July 2002.

[6] S. Tonetti, M. Hehn, S. Lupashin, and R. D'Andrea, "Distributed control of antenna array with formation of UAVs," in *Proceedings of the 18th IFAC World Congress*, pp. 7848–7853, Milano, Italy, September 2011.

[7] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, "Cooperative grasping and transport using multiple quadrotors," in *Distributed Autonomous Robotic Systems*, A. Martinoli, F. Mondada, N. Correll et al., Eds., vol. 83 of *Springer Tracts in Advanced Robotics*, pp. 545–558, Springer, Berlin, Germany, 2013.

[8] K. M. Sobel and E. Y. Shapiro, "A design methodology for pitch pointing flight control systems," *Journal of Guidance, Control, and Dynamics*, vol. 8, no. 2, pp. 181–187, 1985.

[9] D. Kim and Y. Kim, "Optimized feedforward design scheme unifying regulator and command generator tracker," *Journal of Guidance, Control, and Dynamics*, vol. 19, no. 4, pp. 899–904, 1996.

[10] J. A. Guerrero, I. Fantoni, S. Salazar, and R. Lozano, "Flight formation of multiple mini rotorcraft via coordination control," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 620–625, Anchorage, Alaska, USA, May 2010.

[11] A. Chapman and M. Mesbahi, "UAV swarms: models and effective interfaces," in *Handbook of Unmanned Aerial Vehicles*, pp. 1987–2019, Springer, Berlin, Germany, 2014.

[12] M. Bartels and H. Werner, "Cooperative and consensus-based approaches to formation control of autonomous vehicles," in *Proceedings of the 19th IFAC World Congress*, pp. 8079–8084, Cape Town, South Africa, 2014.

[13] O. Oetomo and B. T. Riyanto, "Distributed formation control using consensus for multiple humanoid robots," *Journal of Instrumentation, Automation and Systems*, vol. 1, no. 3, pp. 78–83, 2014.

[14] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: theories and applications," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 340–348, 2015.

[15] Z. Gosiewski and L. Ambroziak, "Formation flight control scheme for unmanned aerial vehicles," in *Robot Motion and Control 2011*, K. Kozłowski, Ed., vol. 422 of *Lecture Notes in Control and Information Sciences*, pp. 331–340, Springer, Berlin, Germany, 2012.

[16] Q. Ali and S. Montenegro, "Explicit model following distributed control scheme for formation flying of mini UAVs," *IEEE Access*, vol. 4, pp. 397–406, 2016.

[17] S. H. Breheny, R. D'Andrea, and J. C. Miller, "Using airborne vehicle-based antenna arrays to improve communications with UAV clusters," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 4, pp. 4158–4162, December 2003.

[18] P. Castillo, A. Dzul, and R. Lozano, "Real-time stabilization and tracking of a four-rotor mini rotorcraft," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 4, pp. 510–516, 2004.

[19] D. Lara, A. Sanchez, R. Lozano, and P. Castillo, "Real-time embedded control system for VTOL aircrafts: Application to stabilize a quad-rotor helicopter," in *Proceedings of the IEEE International Conference on Control Applications (CCA '06)*, pp. 2553–2558, Munich, Germany, October 2006.

[20] V. Kumar and N. Michael, "Opportunities and challenges with autonomous micro aerial vehicles," *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1279–1291, 2012.

[21] E. Lavretsky, "Robust and adaptive control methods for aerial vehicles," in *Handbook of Unmanned Aerial Vehicles*, pp. 675–710, Springer, 2015.

[22] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, John Wiley & Sons, New York, NY, USA, 2003.

[23] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-Agent Systems: Optimal and Adaptive Design Approaches*, Springer, 2013.

[24] Q. Ali and S. Montenegro, "A simple approach to quadrocopter formation flying test setup for education and development," in *Proceedings of the 9th International Technology, Education and Development Conference*, pp. 2776–2784, Madrid, Spain, 2015.