IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 27, Issue 3, Ser. 1 (May. – June. 2025), PP 38-45
www.iosrjournals.org

Rigidity Graph-Based Formation Control For Multi-Quadrotor Systems In Moving Target Tracking And Encirclement

Hoa T. T Nguyen[1], Minh T. Nguyen[1], Anh N.T. Dang[1], Ninh H.T Nguyen[1]
[1]Thai Nguyen University Of Technology, Viet Nam

Abstract:
This paper focuses on the design of a distributed formation control law for multi-quadrotor systems (MQS), utilizing rigidity graph theory and gradient-based formation control laws, and applies these methods to the task of tracking and encircling a moving target. The MQS is represented as an undirected graph that is infinitesimally and minimally rigid. The control law comprises a formation control protocol and a target tracking and encirclement mechanism, ensuring that the formation remains stable during mission execution. To enhance efficiency and simplify the design, the leader-follower strategy is utilized. In this approach, the target velocity is unknown to all quadrotors, but the leader can determine the relative position of the target and estimate its velocity, subsequently transmitting this information to the followers. The proposed control law is validated through simulations conducted in three-dimensional space using MATLAB software. The results demonstrate that the MQS can successfully establish and maintain the desired formation while tracking and encircling the moving target.

***Key Word**: Distributed formation control, multi-quadrotor systems, Rigidity graph theory, Gradient-based formation control laws, Moving target.*

Date of Submission: 22-04-2025 Date of Acceptance: 02-05-2025

I. Introduction

Multi-agent formation control involves the design of control inputs for agents to achieve and maintain a specific geometric configuration in space. In natural, social, and technological domains, large-scale complex systems with multiple agents present significant challenges for centralized formation control, often making its application difficult. Consequently, distributed formation control for multi-agent systems has been extensively researched due to its ease of implementation, strong self-organizational capabilities, and robustness. In engineering applications, multi-agent systems are frequently represented by groups of autonomous vehicles, such as unmanned aerial vehicles (UAVs), unmanned ground vehicles (UGVs), unmanned underwater vehicles (UUVs), etc., which are employed in tasks like tracking, surveillance, mapping, and disaster monitoring. Specifically, for missions such as tracking and encircling aerial targets with UAVs, the multi-quadcopter system (MQS), known for its maneuverability and flexibility, is the most suitable choice. Figure 1 illustrates the distributed formation of a MQS tracking and encircling a moving target, which is an intruding quadrotor.

Figure 1. The distributed formation of a MQS tracking and encircling a moving target

From a mathematical standpoint, the formation control technique is developed based on mathematical concepts from graph theory and consensus dynamics [1]. In this framework, graph theory serves as an effective tool for describing the formation shape of the MQS in space, as well as the sensing, communication, and control topologies between agents in a distributed system. For coordination among agents, each agent must exchange information with its neighboring agents to achieve consensus on common objectives. In formation control problems, the control variables can be absolute position, relative position, or the distance between quadrotors [2]. Rigid graph theory ensures that distance constraints between quadrotors in the desired formation are maintained through the rigidity matrix of the graph, thereby ensuring that they avoid collisions during the formation process. A significant advantage of using the distance as a control variable is that it can be determined through local deviation vectors and is independent of reference frames, unlike methods relying on consensus algorithms. In distance-based formation control, which utilizes relative displacements under misaligned orientations as sensing variables and distances as control variables, the most effective method is the gradient-based approach. Gradient-based formation control laws employ a potential function to generate local controllers for distributed agents. In this approach, the gradient of the potential function must lead the distributed formation controller, making the selection of an appropriate potential function crucial. If the potential function is defined based on the distance errors between neighboring quadrotors, each quadrotor can implement a control law to minimize the potential function, via the local coordinate frame. Moreover, selecting the appropriate control strategy for the task is crucial in the design of formation control for MQS. Formation control includes three main strategies: behavior-based strategy [3, 4], leader-follower strategy [5, 6], and virtual structure strategy [7, 8]. The leader-follower strategy is highly dependent on the accuracy and stability of the leader agent, but for target tracking tasks, particularly with aerial moving targets, this strategy shows its strength through its simple design process and ease of implementation [5, 9].

Motivated by recent advancements and research trends in formation control for MQS based on distance variables, this paper focuses on developing a distributed formation control algorithm grounded in rigid graph theory, applied to the task of tracking, and encircling a moving target while maintaining a predefined formation. The MQS is represented as a formation graph characterized by infinitesimal and minimal rigidity. The leader-follower strategy is employed, where all quadrotors lack direct information about the target's velocity. However, the leader can determine the relative position and estimate the target's velocity, subsequently transmitting this information to the other quadrotors in the formation. The formation control law is designed to stabilize the dynamics of inter-quadrotor distances, ensuring that the quadrotors maintain the desired formation. Consequently, the proposed control law incorporates both a formation stabilization component and a tracking and encircling component to effectively follow the moving target.

The structure of this paper is organized as follows: Section 2 provides a preliminary on the graph theory, graph rigidity, and the quadrotor model. Section 3 outlines the design process of the distributed formation control law for formation control and the tracking and encircling of the moving target. Section 4 presents MATLAB simulation results that validate the proposed control law, demonstrating the ability of the system to maintain the desired formation while tracking and encircling a moving target in three-dimensional space. Finally, Section 5 concludes with a discussion and the future directions for this area of research.

II. Preliminaries

Graph Theory

A graph is used to describe the communication relationship between agents. The basics of graph theory given in this section are adopted from [10]. This relationship between "n" agents can be denoted by a directed or undirected graph. In this study, we focus exclusively on undirected graphs and will henceforth omit the term "undirected" when referring to graph. Define a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{\mathcal{A}})$ where $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$ is the set of all vertices that represented for agents, $\mathcal{E} \subseteq \{(v_i, v_j): v_i, v_j \in \mathcal{V}, i \neq j\}$ is the set of all edges between the vertices, and $\boldsymbol{\mathcal{A}} = [a_{ij}] \in \mathbb{R}^{n\times n}$ is the weighted adjacency matrix with nonnegative entries and defined as

$$a_{ij} = \begin{cases} 1, & \text{if } (i,j) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases} \qquad a_{ij} = a_{ji},\, i \neq j\,,\, a_{ii} = 0 \tag{1}$$

The cardinality of $|\mathcal{V}| = n,$ and $|\mathcal{E}| = m$. A graph $\mathcal{G}$ is considered connected if there exists a path between every pair of vertices. A graph $\mathcal{G}$ is said to be a complete graph if every pair of distinct vertices is connected by an edge, i.e., $m = n(n-1)/2$. Let the degree matrix of a graph $\mathcal{G}$ be denoted as $\mathcal{D} = \text{diag}\,\{d_i\} \in \mathbb{R}^{n\times n}$, where the degree of vertex i is $d_i = \sum_{j\in\mathcal{N}_i} a_{ij}$. The neighbors of the i^{th}agent is defined as $\mathcal{N}_i = \{v_j | (v_i, v_j) \in \mathcal{E}\}$.

A framework is a realization of a graph at given points in Euclidean space. Specifically, if $p_i \in \mathbb{R}^d$ (d-dimensional Euclidean space) is the position values of agents i with respect to some fixed coordinate frame and $\boldsymbol{p} = [p_1, \dots, p_n]^T \in \mathbb{R}^{dn}$ is a configuration of n agents, then a framework F is a pair $F(\mathcal{G}, \boldsymbol{p})$.

Graph rigidity theory

Rigidity theory studies the conditions required for a framework to have a unique realization, where the formation constraints between neighboring agents are defined by some scalar or vector magnitude. When these constraints are represented as distances in an undirected graph, the theory is called graph rigidity. An effective method for determining whether a given framework is distance-rigid is the linear algebra approach, which involves examining the rank of the rigidity matrix. This matrix is derived from the conditions of infinitesimal and minimal rigidity, providing a mathematical tool to assess the system's rigidity.

From the length of the edges in a framework $F(\mathcal{G}, \boldsymbol{p})$, the edge function is defined as

$$\boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}) \triangleq \left[\ldots, \left\|p_i - p_j\right\|^2, \ldots\right]^T, \qquad (i,j) \in \mathcal{E} \tag{2}$$

In the case where two frameworks are congruent, then they are said to be isomorphic in $\mathbb{R}^d$, and the set of all frameworks that are isomorphic to F is denoted by the symbol $\text{Iso}(F)$. Additionally, if two frameworks are equivalent but not congruent, then they are said to be ambiguous, and the set of all ambiguities of framework F is denoted by $\text{Amb}(F)$. A subset of rigidity called infinitesimal rigidity [11], in which the first-order preservation of distances has remained during an infinitesimal motion. The rigidity matrix $\boldsymbol{\mathcal{R}}(\boldsymbol{p}) \in \mathbb{R}^{m \times dn}$ of the framework $F(\mathcal{G}, \boldsymbol{p})$ is defined as:

$$\boldsymbol{\mathcal{R}}(\boldsymbol{p}) = \frac{1}{2}\frac{\partial \boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p})}{\partial \boldsymbol{p}} \tag{3}$$

Based on the rigidity matrix $\boldsymbol{\mathcal{R}}(\boldsymbol{p})$, a framework $F = (\mathcal{G}, \boldsymbol{p})$ is infinitesimally rigid in $\mathbb{R}^d$ if and only if

$$rank\big(\boldsymbol{\mathcal{R}}(\boldsymbol{p})\big) = dn - \frac{d(d+1)}{2}. \tag{4}$$

A graph is minimally rigid if it is rigid, and the removal of any edge causes the graph to lose its rigidity [11]. We can check the condition for minimal rigidity of the graph by the condition of the following result. A rigid graph $\mathcal{G}$ is minimally rigid in $\mathbb{R}^d$ if and only if

$$m = dn - \frac{d(d+1)}{2}. \tag{5}$$

If $F(\mathcal{G}, \boldsymbol{p})$ is infinitesimally and minimally rigid, i.e., $m = rank\big(\boldsymbol{\mathcal{R}}(\boldsymbol{p})\big) = dn - \big(d(d+1)\big)/2$ then its rigidity matrix is full row rank, and $\big(\boldsymbol{\mathcal{R}}(\boldsymbol{p})\boldsymbol{\mathcal{R}}^T(\boldsymbol{p})\big)$ is the positive definite matrix.

Quadrotor model

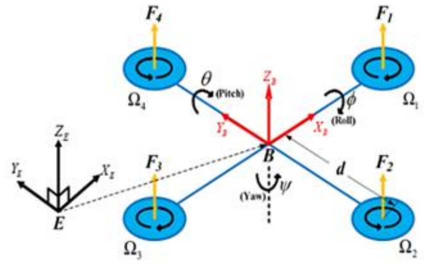

Figure 2. Quadrotor configuration frame scheme of i^{th} quadrotor

In this section, we analyze the dynamic model of the i^{th} quadrotor in a MQS consisting of n quadrotors. The quadrotor is an underactuated mechanical system with six DOFs and four control inputs. These six DOFs include translational motion in three directions and rotational motion around the three axes. Each of the quadrotor's propellers, driven by a motor, is mounted at the end of two cross-shaped frames. To describe the quadrotor's position and orientation, both the inertial frame and the body-fixed frame are utilized. The coordinate system for the quadrotor is illustrated in Figure 2, where the inertial frame is represented by $E = (X_E, Y_E, Z_E)$ and the body-fixed frame $B = (X_B, Y_B, Z_B)$. The inertial frame is based on the Earth with the origin coinciding with the origin of the body-fixed frame before the quadrotor take-off. The body-fixed frame is fixed with the quadrotor with its origin located at the center of mass of the vehicle.

Define the positions and velocities in frame E of the i^{th} quadrotor as $\boldsymbol{p}_i = \left[p_{xi}, p_{yi}, p_{zi}\right]^T$ and $\boldsymbol{v}_i = \left[v_{xi}, v_{yi}, v_{zi}\right]^T$ respectively. The attitude of the i^{th} quadrotor defined in frame E is described by three Euler angles $\boldsymbol{\eta}_i^E = [\phi_i, \theta_i, \psi_i]^T$ where ϕ_i, θ_i and ψ_i denote the angles of roll, pitch, and yaw, respectively. The angular velocity in frame E is expressed as $\dot{\boldsymbol{\eta}}_i^E = \left[\dot{\phi}_i, \dot{\theta}_i, \dot{\psi}_i\right]^T$.Then, the angular velocity $\omega_i^B = [p_i, q_i, r_i]^T$ of the i^{th} quadrotor with respect to frame B is written as:

$$\boldsymbol{\omega}_i^B = \begin{bmatrix} p_i \\ q_i \\ r_i \end{bmatrix} = \mathcal{T}_E^B \dot{\boldsymbol{\eta}}_i^E = \begin{bmatrix} 1 & 0 & -\sin\theta_i \\ 0 & \cos\phi_i & \cos\theta_i \sin\phi_i \\ 0 & -\sin\phi_i & \cos\theta_i\cos\phi_i \end{bmatrix} \begin{bmatrix} \dot{\phi}_i \\ \dot{\theta}_i \\ \dot{\psi}_i \end{bmatrix} \tag{6}$$

where $\mathcal{T}_E^B$ is the transformation matrix for the angular velocity from frame E to frame B.
The dynamics of i^{th} quadrotor with a translational motion and a rotational motion is given by as follows:

$$\begin{cases} \dot{\boldsymbol{p}}_i(t) = \boldsymbol{v}_i(t) \\ m_i\dot{\boldsymbol{v}}_i(t) = \boldsymbol{U}_i^F(t) - \boldsymbol{G}_i^E \\ \dot{\boldsymbol{\eta}}_i^E(t) = \mathcal{T}_B^E \boldsymbol{\omega}_i^B(t) \\ \boldsymbol{I}_i^B \dot{\boldsymbol{\omega}}_i^B(t) = -\boldsymbol{\omega}_i^B\left(I_i^B \times \boldsymbol{\omega}_i^B(t)\right) + \boldsymbol{\tau}_i(t) \end{cases} \tag{7}$$

where m_i is the mass of the i^{th} quadrotor, the virtual control input $\boldsymbol{U}_i^F = \boldsymbol{\mathcal{R}}_{i_B}^{\ E} T_i^B = T_i \boldsymbol{\mathcal{R}}_{i_B}^{\ E} \boldsymbol{e_3}$, $\boldsymbol{e}_3 = [0, \quad 0, \quad 1]^T$. $G_i^E = [0 \quad 0 \quad m_i g]^T$ is gravity, $\boldsymbol{\tau}_i \in \mathbb{R}^{3\times1}$ and $T_i \in \mathbb{R}^{1\times1}$ are the control torque and the total lift generated by the rotors of the i^{th} quadrotor.
The rotation matrix $\boldsymbol{\mathcal{R}}_{i_B}^{\ E}$ is described as follows:

$$\boldsymbol{\mathcal{R}}_{i_E}^{\ B} = \begin{bmatrix} \cos\theta_i \cos\psi_i & \cos\theta_i \sin\psi_i & -\sin\theta_i \\ \sin\phi_i \sin\theta_i \cos\psi_i - \cos\phi_i \sin\psi_i & \sin\phi_i \sin\theta_i \sin\psi_i + \cos\phi_i \cos\psi_i & \sin\phi_i \cos\theta_i \\ \cos\phi_i \sin\theta_i \cos\psi_i + \sin\phi_i \sin\psi_i & \cos\phi_i \sin\theta_i \sin\psi_i - \sin\phi_i \cos\psi_i & \cos\phi_i \cos\theta_i \end{bmatrix} \tag{8}$$

In general, a MQS emphasizes the consistency and stability of the position and velocity of quadrotors. Therefore, this study primarily focuses on translational motion and the design of a distributed formation controller for this motion. The translational motion of the i^{th} quadrotor can be described under the double-integrator model as follows:

$$\begin{cases} \dot{\boldsymbol{p}}_i(t) = \boldsymbol{v}_i(t) \\ \dot{\boldsymbol{v}}_i(t) = \boldsymbol{u}_i(t) \end{cases} \qquad i = 1,2,\dots,n \tag{9}$$

From (7) and (9), the control signal $\boldsymbol{u}_i = \left[u_{xi}, u_{yi}, u_{zi}\right]^T$ is expressed as follows:

$$\boldsymbol{u}_i(t) = \frac{\boldsymbol{U}_i^F}{m_i} - g\boldsymbol{e_3} \tag{10}$$

III. Distributed Formation Control Design

The desired formation is an infinitesimally and minimally rigid framework $F^* = (\mathcal{G}^*, \boldsymbol{p}^*)$, In which, $\mathcal{G}^* = (\mathcal{V}^*, \mathcal{E}^*)$, $\boldsymbol{p}^* = \left[p_1^{*T}, \dots, p_n^{*T}\right]^T$. The distance between quadrotors is $d_{ij}(t) = \left\|p_i(t) - p_j(t)\right\|$. The desired distance between quadrotors is $d_{ij}^* = \left\|p_i^* - p_j^*\right\| > 0, i,j \in \mathcal{V}^*$.

Formation control protocol

Design the first design task of this study to ensure that the quadrotors form and maintain a predefined geometric configuration in space. The control objective for this formation problem serves as the common and foundational objective for the other tasks. This objective is to design u_i such that:

$$d_{ij}(t) \to d_{ij}^* \text{ as } t \to \infty \;\; i,j \in \mathcal{V}^* \tag{11}$$

The control objective for this problem which forms and maintains formation of MQS is to design $u_i = f\left(p_i - p_j, d_{ij}^*\right)$, $i = 1,2,\dots,n$, $\forall j \in N_i$.
Eq. (9) is equivalent to:

$$\boldsymbol{h}_{\mathcal{G}}\left(\boldsymbol{p}(t)\right) \longrightarrow \boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}^*) \text{ as } t \longrightarrow \infty \tag{12}$$

Define the relative position z_{ij} as

$$z_{ij} = p_i - p_j \tag{13}$$

Let the distance-error e_{ij} is given by

$$e_{ij} = \left\|z_{ij}\right\| - d_{ij}^* = d_{ij} - d_{ij}^* \tag{14}$$

Let the error $\bar{e}_{ij}$ is given by

$$\bar{e}_{ij} = \left\|z_{ij}\right\|^2 - \left(d_{ij}^*\right)^2 = d_{ij}^2 - \left(d_{ij}^*\right)^2 \tag{15}$$

$\boldsymbol{z} = \left[\dots, z_{ij}^T, \dots\right]^T \in \mathbb{R}^{dm}$, $\boldsymbol{e} = \left[\dots, e_{ij}, \dots\right]^T \in \mathbb{R}^m$, $\bar{\boldsymbol{e}} = \left[\dots, \bar{e}_{ij}, \dots\right]^T \in \mathbb{R}^m$, $(i,j) \in \mathcal{E}$
From Eq. (2) and Eq. (15), we have that

$$\bar{\boldsymbol{e}} = \left[\dots, \bar{e}_{ij}, \dots\right]^T = \boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}) - \boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}^*) \tag{16}$$

Eq. (16) can be rewritten by

$$\bar{e}_{ij} = e_{ij}\left(e_{ij} + 2d_{ij}^*\right) \tag{17}$$

It is not difficult to see that $e_{ij} \geq -d_{ij}^*$ and $\bar{e}_{ij} = 0$ if and only if $e_{ij} = 0$.
Consider the potential function:

$$\boldsymbol{\mathcal{L}}(\boldsymbol{e}) = \frac{1}{4}\sum_{(i,j)\epsilon\mathcal{E}^*} \bar{e}_{ij}^2 = \frac{1}{4}\|\bar{\boldsymbol{e}}\|^2 = \frac{1}{4}\left\|\boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}) - \boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}^*)\right\|^2 \tag{18}$$

This function is positive definite in $\boldsymbol{e}$, continuously differentiable, and radially unbounded. The time derivative of $\boldsymbol{\mathcal{L}}(\boldsymbol{e})$, we have that

$$\dot{\boldsymbol{\mathcal{L}}} = \frac{1}{4}\frac{d}{dt}\left(\left\|\boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}) - \boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p}^*)\right\|^2\right) = \frac{1}{2}\bar{\boldsymbol{e}}^T \frac{d\left(\boldsymbol{h}_{\mathcal{G}}(\boldsymbol{p})\right)}{dt}$$
$$= \bar{\boldsymbol{e}}^T\boldsymbol{\mathcal{R}}(\boldsymbol{p})\boldsymbol{u}_f \quad (19)$$

where $\boldsymbol{u}_f = \left[\boldsymbol{u}_{f1}, \dots, \boldsymbol{u}_{fn}\right]^T \in \mathbb{R}^{dn}$.

Given the formation framework $F(t) = (\mathcal{G}^*, \boldsymbol{p}(t))$ and let the initial conditions be such that $e(0) = \mathcal{C}_1 \cap \mathcal{C}_2$, $\boldsymbol{e} \in \mathbb{R}^m$,

$$\mathcal{C}_1 = \{\Upsilon(F, F^*) \leq \varepsilon\} \quad (20)$$
$$\mathcal{C}_2 = \left\{\mathrm{dist}\left(\boldsymbol{p}, \mathrm{Iso}(F^*)\right) < dist\left(\boldsymbol{p}, \mathrm{Amb}(F^*)\right)\right\} \quad (21)$$

Taking the gradient of $\boldsymbol{\mathcal{L}}(\boldsymbol{e})$, the formation control law is designed as:

$$\boldsymbol{u}_f = -\nabla\left(\boldsymbol{\mathcal{L}}(\boldsymbol{e})\right) = -\left[\frac{\partial\mathcal{L}(e)}{\partial \boldsymbol{p}}\right]^T = -k_p\boldsymbol{\mathcal{R}}^T(\boldsymbol{p})\bar{\boldsymbol{e}} \quad (22)$$

where $k_p > 0$ is a control gain.

This formation control law makes $\dot{\boldsymbol{\mathcal{L}}}$ in Eq. (19) negative definite and will be embedded in the tracking control law. The condition in Eq. (20) is a sufficient constraint for the formation framework $F(t)$ to remain infinitesimally rigid. The condition in Eq. (21) guarantees that $F(t)$ is closer to a framework in $Iso(F^*)$ at $t = 0$ than to one in $\mathrm{Amb}(F^*)$ to prevent converging to an ambiguous framework. The formation control law for each quadrotor can be expressed as:

$$\boldsymbol{u}_{fi} = -k_p \sum_{j\epsilon N_i} z_{ij}\, \bar{e}_{ij} \quad (23)$$

Target tracking and encirclement mechanism

The second design task of this study is as follows: The quadrotors track and encircle a moving target with a predefined formation. In this problem, we employ the leader-followers strategy by taking the n^{th} quadrotor to be the leader while the other quadrotors are followers. The control protocol includes: Acquiring a desired formation F^*, the leader agent chasing the target, and the followers tracking and surrounding the leader while maintaining the desired formation F^*. Let the target position is denoted as $\boldsymbol{p}_T(t) \in \mathbb{R}^d$, and the second objective is then expressed as:

$$\boldsymbol{p}_T(t) \in \mathrm{conv}\{\boldsymbol{p}_1(t), \boldsymbol{p}_2(t), \dots, \boldsymbol{p}_{n-1}(t)\} \text{ as } t \to \infty \quad (24)$$

In the second problem, we assume that the target velocity $\boldsymbol{v}_T := \dot{\boldsymbol{p}}_T$ is unknown to all quadrotors. However, the leader (n^{th} quadrotor) can measure the target's relative position $\boldsymbol{z}_T = \boldsymbol{p}_T - \boldsymbol{p}_n$ and communicate this information to all followers. The control objective for tracking and encirclement the moving target with a desired formation is to design $\boldsymbol{u}_i = f\left(z_{ij}, d_{ij}^*, \boldsymbol{z}_T, \hat{\boldsymbol{v}}_T\right)$, where $\hat{\boldsymbol{v}}_T$ is the target velocity estimate value that generated by the continuous dynamic estimation mechanism [12]

$$\hat{\boldsymbol{v}}_T(t) = \int_0^t \left[k_1\boldsymbol{z}_T(\tau) + k_2 sgn\left(\boldsymbol{z}_T(\tau)\right)\right]d\tau \quad (25)$$

where $k_1, k_2 > 0$ are control gains.

Consider the formation framework $F(t) = (G^*, \boldsymbol{p}(t))$ and let the initial conditions be such that $\boldsymbol{e}(0) = \mathcal{C}_1 \cap \mathcal{C}_2$ given in Eq. (20) and Eq. (21).

The tracking formation control law $\boldsymbol{u} \in \mathbb{R}^{dn}$ is designed as

$$\boldsymbol{u} = \boldsymbol{u}_f + \boldsymbol{1}_n \otimes \boldsymbol{h} \quad (26)$$

where $\boldsymbol{u}_f \in \mathbb{R}^{dn}$ was defined in Eq. (22), the term $\boldsymbol{h} \in \mathbb{R}^d$ is given by

$$\boldsymbol{h} = k_1\boldsymbol{z}_T + \hat{\boldsymbol{v}}_T - \boldsymbol{u}_{fn} \quad (27)$$

The control law $\boldsymbol{u}$ in Eq. (26) has two components: $\boldsymbol{u}_f$ ensures the formation problem while $\boldsymbol{h}$ guarantees tracking and encircling a moving target. This makes $\boldsymbol{e} = 0$ exponentially stable and ensures Eq. (11) and Eq. (24) are satisfied.

From Eq. (26) and Eq. (27), the leader quadrotor control input $u_n \in \mathbb{R}^d$ is

$$\boldsymbol{u}_n = k_1\boldsymbol{z}_T + \hat{v}_T \quad (28)$$

and the follower i^{th} quadrotor control input $u_i \in \mathbb{R}^d$ is

$$\boldsymbol{u}_i = \boldsymbol{u}_{fi} + \boldsymbol{h}$$
$$= -k_p \sum_{j\epsilon N_i} z_{ij}\, \bar{e}_{ij} + k_1 z_T + \int_0^t \left[k_1\boldsymbol{z}_T(\tau) + k_2 sgn\left(\boldsymbol{z}_T(\tau)\right)\right]d\tau - \boldsymbol{u}_{fn} \quad (29)$$

where $\boldsymbol{u}_{fi} = -k_p \sum_{j\epsilon N_i} z_{ij}\, \bar{e}_{ij}$ $(i = 1, \dots, n-1)$ and $\boldsymbol{u}_{fn} = -k_p \sum_{j\epsilon N_n} z_{nj}\, \bar{e}_{nj}$

It can be observed that the follower control law is distributed, as it depends on its relative position to neighboring quadrotors, the target's relative position, and the formation control of the leader.

IV. Simulation Results

The experiment involved a MQS with 6 quadrotors: quadrotor 6 acts as the leader, responsible for tracking a moving target, while the 5 quadrotors function as followers, tasked with encircling the leader and maintaining the desired formation. In the desired formation, the five vertices of the pentagon represent the followers and the vertex at the center represents the leader. The desired formation was designed infinitesimally and minimally rigid framework. From Eq. (5), we have that the minimal number of edges $m = 12$. Accordingly, the desired topology was established up by the adjacency matrix $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{6\times 6}$ was derived from Eq. (1). The desired formation of MQS in three-dimensional space is shown in Figure 3. The rigidity matrix $\boldsymbol{\mathcal{R}}(\boldsymbol{p}) \in \mathbb{R}^{12\times 18}$ is derived from Eq. (3) and applied to the formation control laws.

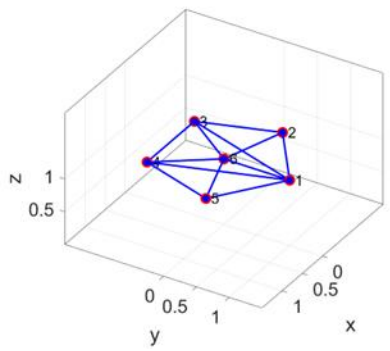

Figure 3. *The desired formation in three-dimensional space*

The desired inter-follower distances of followers are:

$$d_{12}^* = d_{15}^* = d_{23}^* = d_{34}^* = d_{45}^* = 2\sin\frac{\pi}{5} \;;\; d_{13}^* = d_{14}^* = \sqrt{2(1+\cos(\pi/5))} \quad (30)$$

The desired leader-follower distances are

$$d_{16}^* = d_{26}^* = d_{36}^* = d_{46}^* = d_{56}^* = 1 \quad (31)$$

The initial conditions of all quadrotors are randomly selected by $p_i(0) = p_i^* + (\mathrm{rand}(0,1,0) - I_3 0.5)$. $v_i(0) = 2[\mathrm{rand}(0,1,0) - 0.5I_3]$ The control gains: $k_p = 1$, $k_1 = 2$, and $k_2 = 2$. The velocity of the target was set up $v_T = [0, 1, sin\, t]$, this value is not provided for all four-rotor aircraft in the control law. The target initial position $p_T(0) = [0,2,0]$.

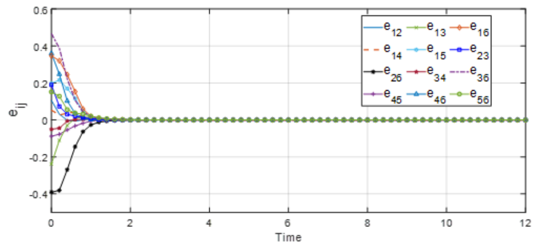

Figure 4. *Distance errors $e_{ij}(t)$*

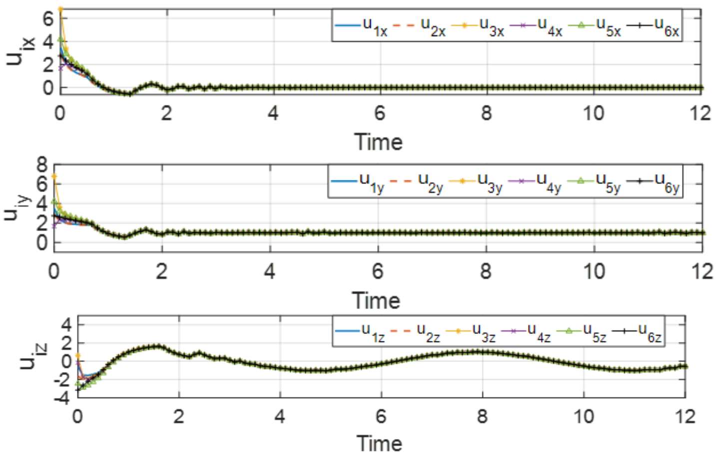

Figure 5. *Control inputs $u_{ix}(t)$, $u_{iy}(t)$ and $u_{iz}(t)$*

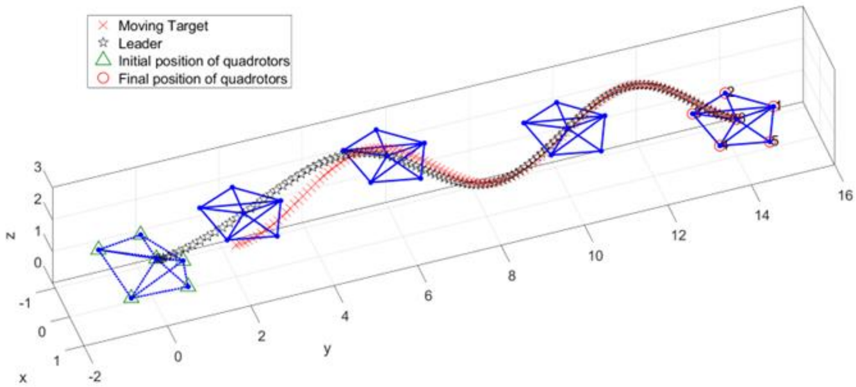

Figure 6. *The formation F(t) tracks and encircles the moving target while maintaining the desired formation*

The simulation results presented in Figure 4 show that the distance errors $e_{ij}(t)$, converge to zero. This indicates that the MQS successfully achieves and maintains the desired formation throughout the process of tracking and encircling the moving target. Figure 5 illustrates that the control inputs of each quadrotor, $u_i = [u_{ix}, u_{iy}, u_{iz}]$ converge to $v_T = [0,1, sin\ t]$, even though the quadrotors in the formation do not have direct access to the target's velocity information. This result highlights the effectiveness of the proposed control algorithm in estimating and tracking the target's trajectory, with the quadrotors adjusting their control inputs to align with the target's motion profile.

The results shown in Figure 6 illustrate the process of formation, tracking, and encircling a moving target using the proposed control laws in three-dimensional space. The leader follows the trajectory of the moving target, while the followers track and encircle the target according to the desired formation. Throughout the entire movement, the formation remains intact, demonstrating that the proposed control laws effectively preserve the formation without any distortion during the process.

V. Conclusion

This study introduces the distributed formation control law in the design for multi-quadrotor systems based on rigid graph theory, applied to tracking, and encircling a moving target. The formation is represented as an undirected graph with differential distance rigidity and minimal rigidity. Consequently, the control law is developed based on the rigidity matrix to drive the distances between quadrotors toward the desired formation distances. The control law consists of two components: the formation control protocol and the target tracking and encircling control mechanism. The first control component is highly distributed, as it depends solely on the quadrotor's information and the information from neighboring quadrotors. In contrast, the second control component requires information transmitted from the leader to the followers; however, it still maintains the properties of a distributed group controller since the leader is also considered a neighbor of the other quadrotors. The proposed control algorithm has been validated for its effectiveness through simulation results in a three-dimensional space using MATLAB.

Future research in this area could explore the integration of dynamic and time-varying environments into formation control algorithms, enhancing the system's adaptability to real-world scenarios. Additionally, incorporating communication constraints and limited sensing capabilities could further improve the robustness of MQS.

Acknowledgements

The authors would like to thank Thai Nguyen University of Technology (TNUT), Vietnam for the support. (http://www.tnut.edu.vn/).

References

[1]. Ahn, H. S. Formation Control. Springer International Publishing, 2020.
[2]. Oh, Kwang-Kyo, Myoung-Chul Park, And Hyo-Sung Ahn. "A Survey Of Multi-Agent Formation Control." Automatica 53, Pp. 424-440, 2015.
[3]. F. Michaud, M. Nicolescu, "Behavior-Based Systems" Springer Handbook Of Robotics, Pp.307–328, 2016.
[4]. G. Lee And D. Chwa, "Decentralized Behavior-Based Formation Control Of Multiple Robots Considering Obstacle Avoidance," Intelligent Service Robotics, Vol. 11, Pp. 127–138, 2018.
[5]. H. T. Do, H. T. Hua, M. T. Nguyen, C. V. Nguyen, H. T. Nguyen, H. T. Nguyen, And N. T. Nguyen, "Formation Control Algorithms For Multiple-Uavs: A Comprehensive Survey," EAI Endorsed Transactions On Industrial Networks And Intelligent Systems, Vol. 8, No. 27, Pp. E3–E3, 2021.
[6]. V. Rold˜Ao, R. Cunha, D. Cabecinhas, C. Silvestre, And P. Oliveira, "A Leader-Following Trajectory Generator With Application To Quadrotor Formation Flight," Robotics And Autonomous Systems, Vol. 62, No. 10, Pp. 1597–1609, 2014.

[7]. C. K. Peterson And J. Barton, "Virtual Structure Formations Of Cooperating Uavs Using Wind Compensation Command Generation And Generalized Velocity Obstacles," In 2015 IEEE Aerospace Conference, IEEE, Pp. 1–7, 2015.
[8]. D. Zhou, Z. Wang, And M. Schwager, "Agile Coordination And Assistive Collision Avoidance For Quadrotor Swarms Using Virtual Structures," IEEE Transactions On Robotics, Vol. 34, No. 4, Pp. 916–923, 2018.
[9]. H. T. Do, H. T. Hua, H. T. Nguyen, M. T. Nguyen, And H. T. Tran, "Cooperative Tracking Framework For Multiple Unmanned Aerial Vehicles (Uavs)," In Advances In Engineering Research And Application: Proceedings Of The International Conference On Engineering Research And Applications, ICERA 2021, Springer, Pp. 276–285, 2022
[10]. Agnarsson, Geir, And Raymond Greenlaw. Graph Theory: Modeling, Applications, And Algorithms. Prentice-Hall, Inc., 2006.
[11]. IZMESTIEV, Ivan. Infinitesimal Rigidity Of Frameworks And Surfaces. Lectures On Infinitesimal Rigidity, 2009.
[12]. Qu, Zhihua, And Jian-Xin Xu. "Model-Based Learning Controls And Their Comparisons Using Lyapunov Direct Method." Asian Journal Of Control 4.1 (2002): 99-110.