Fault-tolerant cooperative navigation of networked UAV swarms for forest fire monitoring

Junyan Hu [a], Hanlin Niu [b,*], Joaquin Carrasco [b], Barry Lennox [b], Farshad Arvin [b]

[a] *Department of Computer Science, University College London, London, WC1E 6BT, UK*
[b] *Department of Electrical and Electronic Engineering, The University of Manchester, Manchester, M13 9PL, UK*

A B S T R A C T

Coordination of unmanned aerial vehicle (UAV) swarms has received significant attention due to its wide practical applications including search and rescue, cooperative exploration and target surveillance. Motivated by the flexibility of the UAVs and the recent advancement of graph-based cooperative control strategies, this paper aims to develop a fault-tolerant cooperation framework for networked UAVs with applications to forest fire monitoring. Firstly, a cooperative navigation strategy based on network graph theory is proposed to coordinate all the connected UAVs in a swarm in the presence of unknown disturbances. The stability of the aerial swarm system is guaranteed using the Lyapunov approach. In case of damage to the actuators of some of the UAVs during the mission, a decentralized task reassignment algorithm is then applied, which makes the UAV swarm more robust to uncertainties. Finally, a novel geometry-based collision avoidance approach using onboard sensory information is proposed to avoid potential collisions during the mission. The effectiveness and feasibility of the proposed framework are verified initially by simulations and then using real-world flight tests in outdoor environments.

1. Introduction

Due to the advancement of communication, computing and sensing techniques over the past two decades, navigation and coordination of unmanned aerial vehicle (UAV) swarms have attracted increasing attention from both academia and industry [1]. By utilizing the cooperative control strategies, a large number of UAVs connected via a communication network can be steered to achieve a common goal. When properly designed, the collaboration of the networked UAVs can be ensured by using neighboring state information via onboard sensors and communication modules, which offers more flexibility and cost-effectiveness over a single sophisticated UAV in solving complex and challenging tasks. Compared to wheeled mobile robots, the high flexibility and maneuverability of UAVs make them ideal platforms to perform autonomous tasks in extreme environments [2]. UAV swarms have been successfully implemented in many real-world applications, including search and

hunting [3–5], cooperative transportation [6], autonomous exploration and mapping [7], target surveillance [8], etc. For example, the literature [9] lays a significant contribution in developing a clustering-based path planning and task allocation method for multi-UAV area coverage. In recent years, severe forest fires around the world have stimulated research into the development of advanced UAV navigation strategies for forest fire monitoring [10].

Pioneer research on cooperative forest fire surveillance was conducted in [11], where a team of micro UAVs with limited communication and sensing range were coordinated to cooperatively track the propagation of large forest fires. In order to achieve the goal, a real-time algorithm was proposed to estimate the perimeter of fires and a distributed fire monitoring algorithm was then designed for position tracking. In [12], a Kalman filter-based methodology was proposed to estimate the wildfire propagation behavior and fire front contour with online measurements of UAVs. Besides, an uncertainty function was developed to describe the performance of the estimation results. A min-time max-coverage issue in sweep coverage was considered in [13], where a set of UAVs was dispatched to efficiently patrol the targets in the given fire area to achieve maximum coverage in minimum time. A distributed control framework was designed in [14] for a team of UAVs that can closely monitor a wildfire in open space and precisely track its de-

* Corresponding author.
E-mail addresses: Junyan.Hu@ucl.ac.uk (J. Hu), Hanlin.Niu@manchester.ac.uk (H. Niu), Joaquin.Carrasco@manchester.ac.uk (J. Carrasco), Barry.Lennox@manchester.ac.uk (B. Lennox), Farshad.Arvin@manchester.ac.uk (F. Arvin).

velopment. However, to the best of the authors' knowledge, most of the aforementioned works do not cover the robustness of the UAV system against uncertain disturbances in the environments and loss of team members due to partial or full damage caused by fires.

To achieve an efficient and robust navigation performance, various guidance strategies have been developed by researchers in recent years. In [15], an autonomous path planning algorithm using a tangent intersection and guidance method was proposed for UAVs. Taking the UAV kinematic constraints into consideration, a feasible collision-free path can be obtained via smooth waypoints. In another study [16], a road following vision-based guidance system was developed for UAVs in GPS-denied environments. To validate the proposed road tracking systems, actual test flights were conducted using an RGB camera and hyperspectral sensor. In [17], an extended signal-correction observer was designed to reject large sensor errors and to measure the system uncertainty in the presence of stochastic non-Gaussian noise. This method was then implemented in an aircraft's navigation system. Unfortunately, the aforementioned methods can only be applied to a single UAV and the cooperation problems of multiple UAVs in the swarm have not been addressed.

When there exist multiple UAVs in a cooperative task, the design of the navigation system becomes more challenging as the cooperation of the UAVs must be coordinated properly to avoid conflicts and achieve higher efficiency. To solve this issue, a number of multi-robot coordination strategies have been proposed for various applications in recent years. In [18], a voronoi-based path planning method was proposed to deal with multi-agent cooperative exploration tasks in unknown environments. The navigation system included a high-level waypoint generation layer and a low-level target tracking layer based on deep reinforcement learning techniques. An adaptive approach to achieve dynamic formation was proposed in [19] for a team of tri-rotor UAVs. A distributed guidance law was proposed in [20] for multiple rotary-wing UAVs to intercept several intruding targets. The proposed distributed guidance law was via the integration of the classic parallel navigation and velocity feedback. Optimal and adaptive formation control of multi-robot systems and UAV swarms were explored in [21] and [22], respectively. In [23], a group mobile computing approach was developed for maritime search and rescue tasks, where unmanned aerial and surface vehicles were used to form a cognitive mobile computing network and reinforcement learning was used to plan a search path. However, most of the aforementioned works do not cover the safety of the UAV system subjected to switching network topology due to loss of communications and potential collision risks during the cooperation tasks.

So far, we have discussed some recent advances in the UAV navigation system and the cooperative control design. How to efficiently coordinate a swarm of UAVs when undertaking cooperation tasks remains an unsolved problem. There are three major problems that should be urgently resolved: (i) how to design a decentralized navigation strategy for UAVs using neighboring information via the communication network? (ii) How to deal with practical limitations such as unknown disturbances, loss of team members and obstacle avoidance? (iii) How to implement the proposed algorithm in a real UAV system? To address these questions, in this article, we propose a fault-tolerant cooperative navigation framework for UAV swarms that is able to perform target-tracking missions. To the best of authors' knowledge, such a unified cooperative coordination framework for networked UAVs has not been established in the literature. The main contributions of this article can be summarized as follows:

- A robust guidance law is proposed for the networked UAVs with switching interaction topologies to form a desired geo-
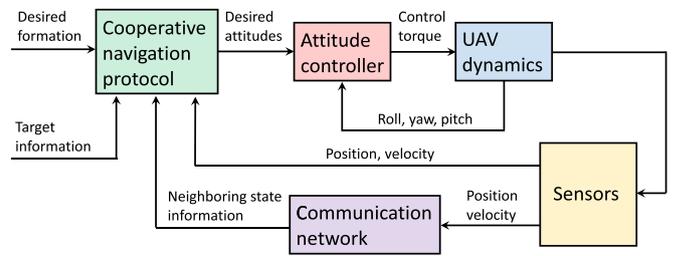

Fig. 1. The two-layer control architecture of the UAV system.

metric pattern around the target. Based on local information, using measurements from neighboring states, all the UAVs are able to achieve a circular formation to enable monitoring the full range of the target activity.
- In case of the loss of one or more UAVs from damage or recharging, a task reassignment and formation reconstruction algorithm based on the distributed Hungarian algorithm is then developed to ensure the robustness of the UAV swarm.
- A geometry-based collision detection and avoidance algorithm is proposed to avoid potential collisions during the mission. Furthermore, the proposed results are validated by real-world flight tests in outdoor environments.

2. Problem statement

2.1. Preliminaries on graph theory

Consider a communication network topology $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ with some UAVs and associated communication links. A link started at the ith UAV and ended at the jth UAV is denoted by (i, j), which means information can flow from UAV i to UAV j and thus UAV j is a neighbor of UAV i. a_{ij} is the weight of link (j, i) and $a_{ij} = 1$ if $(j, i) \in \mathcal{E}$. Let $D = \text{diag}\{d_i\}$ with $d_i = \sum_{j=1}^{N} a_{ij}$ and the Laplacian matrix L of $\mathcal{G}$ is then designed by $L = D - \mathcal{A}$. If UAV i can obtain the target's information directly (labeled as '0'), then a link $(0, i)$ is generated between them with the weight $g_i = 1$. The pinning matrix is defined as $G = \text{diag}(g_i)$.

During the cooperation mission, the communication topology of the UAV swarm system may need to be switched. For all $p \in \mathbb{N}_+$, the interaction topology changes at the switching sequence t_p and remains fixed during the time interval ς_0. Let $[t_p, t_{p+1})$ be an infinite sequence of uniformly bounded non-overlapping time interval, with $t_1 = 0$, $0 < \varsigma_0 \le t_{p+1} - t_p \le \varsigma_1$. Let $\eta : [0, +\infty) \to \{1, \ldots, p\}$ be a switching signal whose value at time t is the index of the topology. The associated Laplacian matrix and pinning matrix of the graph at $\eta(t)$ are described by $L_{\eta(t)}$ and $G_{\eta(t)}$, respectively.

2.2. Dynamic model of UAV swarm

Since the attitude dynamics of each UAV has a smaller time constant relative to the trajectory, by implementing the two-time-scale separation principle [24], the UAV can be coordinated by an outer/inner loop architecture as shown in Fig. 1. In the inner loop, the attitude of each UAV is stabilized by the attitude controller. Based on the state information obtained from the onboard/global sensors and the neighboring information via communication networks, the desired trajectory of each UAV can be designed in the outer loop.

In this paper, we particularly focus on the outer loop navigation strategy design. Besides, in the cooperation task, it is assumed that all the UAVs fly at the same altitude. Hence, the dynamics of each UAV can then be simplified as a double-integrator system in two-dimensional space following [25–27]. Considering that
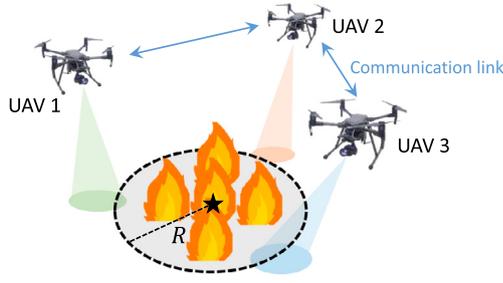
Fig. 2. An example shows a circular formation with three networked quadrotor UAVs tracking a fire target with radius of R.

some uncertainties may exist in the environments (e.g., wind disturbances), we can use the following equations to describe the UAV swarm

$$\begin{cases} \dot{p}_i(t) = v_i(t) \\ \dot{v}_i(t) = u_i(t) + d_i(t) \end{cases} \quad \forall i \in \{1, 2, \ldots, N\}, \quad (1)$$

where $p_i = [p_{xi}, p_{yi}]^T$ and $v_i = [v_{xi}, v_{yi}]^T$ denote the position vector and velocity vector of the UAV i, respectively. $u_i = [u_{xi}, u_{yi}]^T$ is the control input to be designed. $d_i = [d_{xi}, d_{yi}]^T$ is an unknown time-varying input disturbance signal, which can be parameterized with a set of base functions as

$$d_i = \kappa_i \theta_i(t), \qquad \forall i \in \{1, 2, \ldots, N\}, \quad (2)$$

where $\theta_i(t)$ is the base function vector and κ_i is an unknown constant matrix.

The feedback linearized dynamic (1) can also be described in the following state-space form

$$\dot{x}_i = A x_i + B(u_i + d_i) \qquad \forall i \in \{1, 2, \ldots, N\}, \quad (3)$$

where the matrices

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

and $x_i = [p_{xi}, v_{xi}, p_{yi}, v_{yi}]^T$. It can be readily verified that the pair (A, B) is stabilizable.

2.3. Problem description

In this work, we aim to design a decentralized cooperative circumnavigation strategy for application to networked UAVs swarms during target tracking missions. A motivating example is illustrated in Fig. 2, where a forest fire monitoring task can be performed by a collaborative UAV swarm using onboard cameras. Suppose that the fire area with radius of R is very large, which cannot be monitored by a single UAV due to the limited field of view. To address this issue, multiple networked UAVs can be deployed to attain a time-varying circular formation surrounding the fire for monitoring the full range of its activity. Each UAV should be able to communicate with its neighbors to share obtained information, such that the mission can be accomplished via a distributed manner. Hence, this paper aims to develop a cooperative framework that can be used in many similar real-world applications. To show the usefulness of such collective behaviors and improve readability, we adopt this example as one of the potential applications of the proposed navigation strategy throughout this paper.

The dynamic model of the target (i.e., the location of the fire) can be described by the following state-equation for analysis

$$\dot{x}_0 = A x_0 + B u_0, \quad (4)$$

where $x_0 = [p_{x0}, v_{x0}, p_{y0}, v_{y0}]^T$ and $u_0 = [u_{x0}, u_{y0}]^T$ are the state and exogenous input of the target reflecting its uncertain maneuvers.

Considering the uncertain movements of the target, the exogenous input u_0 can be described by the follow function

$$u_0(t) = \kappa_0 \theta_0(t), \quad (5)$$

where $\theta_0(t)$ is the base function vector and κ_0 is an unknown constant matrix.

Remark 1. The parameterized model shown in (2) and (5) is motivated by the reference [28,29], where the unknown exogenous input of the target and the time-varying disturbances of the UAVs are linearly parameterized by some base functions. By using this technique, each UAV identifies its disturbance and the input of the target, thus the effect of such unknown dynamics can be eliminated by the adaptive control protocol. Note that the selection of the base functions is not unique in different real-world applications, which needs to be fixed by the engineers depending on the specific mission and the environmental conditions.

This article mainly solves the following three problems: (i) how to design a coordination strategy to achieve the desired circular formation around the fire while avoiding potential collisions; (ii) how to construct a decentralized task reassignment algorithm to handle unexpected faults during the mission; and (iii) how to apply the proposed approach to the real-world flight tests in the outdoor environments.

3. Fault-tolerant cooperative navigation design

3.1. Coordination algorithm for UAVs

In this subsection, a cooperative circumnavigation algorithm is proposed for the UAV swarm. For each UAV in the team, only neighboring information is required to achieve a desired circular formation surrounding the target.

The dynamic circular formation for the UAVs can be specified by

$$f_i(t) = \begin{bmatrix} R \sin\left(wt + \frac{2(i-1)\pi}{N}\right) \\ wR \cos\left(wt + \frac{2(i-1)\pi}{N}\right) \\ R \cos\left(wt + \frac{2(i-1)\pi}{N}\right) \\ -wR \sin\left(wt + \frac{2(i-1)\pi}{N}\right) \end{bmatrix}, \quad (6)$$

where R is the radius of the target and w is the desired angular velocity for monitoring the full range of target conditions. Let $f(t) = \left[f_1^T(t), f_2^T(t), \ldots, f_N^T(t)\right]^T$ be the global formation vector.

Considering that the unknown matrices κ_0 and κ_i are not available to the UAVs, they can be estimated by the ith UAV by $\hat{\kappa}_{i0}$ and $\hat{\kappa}_i$, respectively. Inspired by [30], an adaptive guidance protocol with time-varying coupling gains is designed by:

$$\begin{cases} u_i = c_i K \xi_i + \gamma_i + \hat{\kappa}_{i0} \theta_0 - \hat{\kappa}_i \theta_i \\ \dot{c}_i = \rho_i \xi_i^T \Gamma \xi_i \\ \dot{\hat{\kappa}}_{i0} = -\tau_i \Xi \xi_i \theta_0^T \\ \dot{\hat{\kappa}}_i = \sigma_i \Xi \xi_i \theta_i^T \end{cases} \quad (7)$$

where $\xi_i = \sum_{j=1}^{N} a_{ij}\left((x_i - f_i) - (x_j - f_j)\right) + g_i\left((x_i - f_i) - x_0\right)$ is the formation tracking error, ρ_i, τ_i and σ_i are positive scalars. K, Γ and Ξ are constant protocol gains to be determined. γ_i is a continuously differentiable function to be designed later.

Since B hs have full rank, there exists the pseudo inverse

$$\tilde{B} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

such that $\tilde{B}B = I_2$.

To satisfy some of the aforementioned objectives, we propose the following Theorem 1 to address the robust formation tracking issues for networked UAV swarms with target's uncertain maneuver and unknown disturbances.

Theorem 1. *The cooperative navigation task is achieved in the UAV swarm by connected switching interaction topologies and applying the adaptive coordination protocol given in (7) with $K = -R^{-1}B^T P$, $\Gamma = PBR^{-1}B^T P$, $\Xi = (PB)^T$ and $\gamma_i = \tilde{B}\dot{f}_i - \tilde{B}Af_i$, where $P > 0$ is the solution of the algebraic Riccati (ARE) equation*

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{8}$$

for given $Q > 0$ and $R > 0$.

Proof. Define the local formation tracking error of each UAV as $\delta_i = x_i - f_i - x_0$. We have

$$\dot{\delta}_i = A\delta_i + B(u_i + d_i - u_0) + Af_i - \dot{f}_i \tag{9}$$

Let $\delta = [\delta_1^T, \delta_2^T, \dots, \delta_N^T]^T$. Then by using Kronecker product techniques, $\xi = [\xi_1^T, \dots, \xi_N^T]^T$ can be described by

$$\xi = \big((L_{\eta(t)} + G_{\eta(t)}) \otimes I_2\big)\delta. \tag{10}$$

Let $\tilde{\kappa}_{j0} = \hat{\kappa}_{j0} - \kappa_0$ and $\tilde{\kappa}_j = \hat{\kappa}_j - \kappa_j$ for all $j \in \{1, \dots, N\}$. The closed-loop error dynamics δ using the robust control law (7) can be described in a structured form as

$$\begin{aligned}
\dot{\delta} =& \big(I_N \otimes A + \hat{C}(L_{\eta(t)} + G_{\eta(t)}) \otimes BK\big)\delta \\
&+ \big(I_N \otimes A\big)f + (I_N \otimes B)(\tilde{\kappa}_0\Theta_0 - \tilde{\kappa}\Theta) \\
&- \big(I_N \otimes I_2\big)\dot{f} + \big(I_N \otimes B\big)\gamma,
\end{aligned} \tag{11}$$

where $\hat{C} = \text{diag}(c_1, \dots, c_N)$, $\tilde{\kappa}_0 = \text{diag}(\tilde{\kappa}_{10}, \dots, \tilde{\kappa}_{N0})$, $\tilde{\kappa} = \text{diag}(\tilde{\kappa}_1, \dots, \tilde{\kappa}_N)$, $\Theta_0 = \text{col}(\theta_0, \dots, \theta_0)$, $\Theta = \text{col}(\theta_1, \dots, \theta_N)$ and $\gamma = [\gamma_1^T, \gamma_2^T, \dots, \gamma_N^T]^T$.

Consider the following Lyapunov function candidate,

$$\begin{aligned}
V =& \delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes P\big)\delta + \sum_{i=1}^{N} \frac{1}{\rho_i}(c_i - \alpha)^2 \\
&+ \sum_{i=1}^{N} \text{tr}(\frac{1}{\tau_i}\tilde{\kappa}_{i0}^T\tilde{\kappa}_{i0}) + \sum_{i=1}^{N} \text{tr}(\frac{1}{\sigma_i}\tilde{\kappa}_i^T\tilde{\kappa}_i),
\end{aligned} \tag{12}$$

where α is a positive scalar to be determined later.

Then, the time derivative of V along the trajectory of (11) can be obtained by

$$\begin{aligned}
\dot{V} =& 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes PA\big)\delta + \sum_{i=1}^{N} 2(c_i - \alpha)\xi_i^T \Gamma \xi_i \\
&+ 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes PB\big)(\tilde{\kappa}_0\Theta_0 - \tilde{\kappa}\Theta) \\
&+ 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)})\hat{C}(L_{\eta(t)} + G_{\eta(t)}) \otimes PBK\big)\delta \\
&+ 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes PA\big)f \\
&- 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes I_2\big)\dot{f} \\
&+ 2\sum_{i=1}^{N} \text{tr}(\frac{1}{\tau_i}\tilde{\kappa}_{i0}^T\dot{\tilde{\kappa}}_{i0}) + 2\sum_{i=1}^{N} \text{tr}(\frac{1}{\sigma_i}\tilde{\kappa}_i^T\dot{\tilde{\kappa}}_i) \\
&+ 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes PB\big)\gamma.
\end{aligned} \tag{13}$$

Note that

$$2\delta^T \big((L_{\eta(t)} + G_{\eta(t)})\hat{C}(L_{\eta(t)} + G_{\eta(t)}) \otimes PBK\big)\delta$$
$$= -2\sum_{i=1}^{N} c_i\xi_i^T PBR^{-1}B^T P\xi_i \tag{14}$$

and

$$\sum_{i=1}^{N} 2c_i\xi_i^T PBR^{-1}B^T P\xi_i = 2\sum_{i=1}^{N} c_i\xi_i^T PBR^{-1}B^T P\xi_i. \tag{15}$$

Consider that

$$\begin{aligned}
Af_i &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} R\sin(wt + \frac{2(i-1)\pi}{N}) \\ wR\cos(wt + \frac{2(i-1)\pi}{N}) \\ R\cos(wt + \frac{2(i-1)\pi}{N}) \\ -wR\sin(wt + \frac{2(i-1)\pi}{N}) \end{bmatrix} \\
&= \begin{bmatrix} wR\cos(wt + \frac{2(i-1)\pi}{N}) \\ 0 \\ -wR\sin(wt + \frac{2(i-1)\pi}{N}) \\ 0 \end{bmatrix},
\end{aligned} \tag{16}$$

$$\dot{f}_i = \begin{bmatrix} wR\cos(wt + \frac{2(i-1)\pi}{N}) \\ -w^2R\sin(wt + \frac{2(i-1)\pi}{N}) \\ -wR\sin(wt + \frac{2(i-1)\pi}{N}) \\ -w^2R\cos(wt + \frac{2(i-1)\pi}{N}) \end{bmatrix}, \tag{17}$$

$$\begin{aligned}
B\gamma_i &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}(\dot{f}_i - Af_i) \\
&= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -w^2R\sin(wt + \frac{2(i-1)\pi}{N}) \\ 0 \\ -w^2R\cos(wt + \frac{2(i-1)\pi}{N}) \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ -w^2R\sin(wt + \frac{2(i-1)\pi}{N}) \\ 0 \\ -w^2R\cos(wt + \frac{2(i-1)\pi}{N}) \end{bmatrix}.
\end{aligned} \tag{18}$$

It can be readily shown that

$$Af_i - \dot{f}_i + B\gamma_i = 0, \tag{19}$$

which can then be rewritten in the following form

$$(I_N \otimes A)f - (I_N \otimes I_2)\dot{f} + (I_N \otimes B)\gamma = 0. \tag{20}$$

Pre-multiplying both sides of (20) by $(L_{\eta(t)} + G_{\eta(t)}) \otimes P$, we obtain

$$\begin{aligned}
&((L_{\eta(t)} + G_{\eta(t)}) \otimes PA)f - ((L_{\eta(t)} + G_{\eta(t)}) \otimes P)\dot{f} \\
&+ ((L_{\eta(t)} + G_{\eta(t)}) \otimes PB)\gamma = 0.
\end{aligned} \tag{21}$$

Substituting (14), (15) and (21) into (13), we get

$$\begin{aligned}
\dot{V} =& \delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes (PA + A^T P) \\
&- 2\alpha(L_{\eta(t)} + G_{\eta(t)})^2 \otimes PBR^{-1}B^T P\big)\delta \\
&+ 2\delta^T \big((L_{\eta(t)} + G_{\eta(t)}) \otimes PB\big)(\tilde{\kappa}_0\Theta_0 - \tilde{\kappa}\Theta) \\
&+ 2\sum_{i=1}^{N} \text{tr}(\frac{1}{\tau_i}\tilde{\kappa}_{i0}^T\dot{\tilde{\kappa}}_{i0}) + 2\sum_{i=1}^{N} \text{tr}(\frac{1}{\sigma_i}\tilde{\kappa}_i^T\dot{\tilde{\kappa}}_i).
\end{aligned} \tag{22}$$

According to the properties $\text{tr}(X^T) = \text{tr}(X)$ and $\text{tr}(XY) = \text{tr}(YX)$ where X and Y are two matrices with appropriate dimensions, we have

$$\delta^T\big((L_{\eta(t)} + G_{\eta(t)}) \otimes PB\big)(\tilde{\kappa}_0\Theta_0 - \tilde{\kappa}\Theta)$$
$$= \sum_{i=1}^{N}\big(\sum_{j=1}^{N}\hat{L}_{ij}\delta_j^T\big)(PB)\tilde{\kappa}_{i0}\theta_0 - \sum_{i=1}^{N}\big(\sum_{j=1}^{N}\hat{L}_{ij}\delta_j^T\big)(PB)\tilde{\kappa}_i\theta_i$$
$$= \sum_{i=1}^{N}\text{tr}(\tilde{\kappa}_{i0}(PB)^T\xi_i\theta_0^T) - \sum_{i=1}^{N}\text{tr}(\tilde{\kappa}_i(PB)^T\xi_i\theta_i^T), \tag{23}$$

where $\hat{L}_{ij}$ is the element on the ith row and j^{th} column of the matrix $(L_{\eta(t)} + G_{\eta(t)})$.

Selecting the following adaptive terms

$$\dot{\tilde{\kappa}}_{i0} = -\tau_i(PB)^T\xi_i\theta_0^T, \tag{24}$$
$$\dot{\tilde{\kappa}}_i = \sigma_i(PB)^T\xi_i\theta_i^T. \tag{25}$$

Substituting (24) and (25) into (22) yields

$$\dot{V} = \delta^T\big((L_{\eta(t)} + G_{\eta(t)}) \otimes (PA + A^TP)$$
$$- 2\alpha(L_{\eta(t)} + G_{\eta(t)})^2 \otimes PBR^{-1}B^TP\big)\delta. \tag{26}$$

Because all the eigenvalues of the symmetric matrix $(L_{\eta(t)} + G_{\eta(t)})$ have positive real parts [31], there exists a non-singular U such that $U^T(L_{\eta(t)} + G_{\eta(t)})U$ is in the diagonal form $J = \text{diag}(\lambda_1, \ldots, \lambda_N)$, where $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$ are the eigenvalues of $(L_{\eta(t)} + G_{\eta(t)})$. Let $\Phi = [\Phi_1^T, \ldots, \Phi_N^T]^T = (U^T \otimes I_2)\delta$, then from (26) we can obtain

$$\dot{V} = \delta^T\big((L_{\eta(t)} + G_{\eta(t)}) \otimes (PA + A^TP)$$
$$- 2\alpha(L_{\eta(t)} + G_{\eta(t)})^2 \otimes PBR^{-1}B^TP\big)\delta$$
$$= \Phi^T\big(J \otimes (PA + A^TP) - 2\alpha J^2 \otimes \Gamma\big)\Phi$$
$$= \sum_{i=1}^{N}\lambda_i\Phi_i^T(PA + A^TP - 2\alpha\lambda_i\Gamma)\Phi_i. \tag{27}$$

Selecting $\alpha \geq \frac{1}{\lambda_1}$, the expression of $\dot{V}$ can be simplified to

$$\dot{V} \leq \sum_{i=1}^{N}\lambda_i\Phi_i^T(PA + A^TP - PBR^{-1}B^TP)\Phi_i. \tag{28}$$

This implies $\dot{V} \leq 0$ since $(PA + A^TP - PBR^{-1}B^TP) = -Q < 0$ via (8) and furthermore, $\dot{V} = 0$ when $\Phi = 0$. Following LaSalle's invariance principle [32], (11) is asymptotically stable. Hence, $\lim_{t\to\infty}\Phi(t) = 0$, and

$$\lim_{t\to\infty}\delta(t) = 0, \tag{29}$$

since Φ and δ are related via non-singular transformation. This is to say, the dynamic formation tracking of the UAV swarm is achieved.

Remark 2. The proof of the convergence of the swarm system under switching topologies follows several standard approaches such as [25] and [33], where it is assumed that the Laplacian matrix $L_{\eta(t)}$ and pinning matrix $G_{\eta(t)}$ are constants during the time interval. In this study, we mainly focus on the controller design for guaranteeing the stability of the UAV swarm with proper dwell time. Please refer to [34] for more discussions regarding the stability analysis of switching systems.

According to Theorem 1, a systematic procedure to construct the guidance law u_i is provided in Algorithm 1.

Algorithm 1 Procedure to design the control protocol for the UAV swarm.

1: **for** each UAV $i \in \{1, \ldots, N\}$ **do**
2:    set the desired formation reference $f_i(t)$;
3:    choose positive constants ρ_i, τ_i and σ_i;
4:    find $P > 0$ by solving the ARE (8) for given $Q > 0$ and $R > 0$;
5:    compute the controller gain matrices K, Γ and Ξ, and the smooth function γ_i using $P > 0$;
6:    update c_i, $\hat{\kappa}_{i0}$ and $\hat{\kappa}_i$;
7:    construct the adaptive guidance protocol u_i given in (7);
8: **end for**

3.2. Task reassignment and formation reconfiguration

In the cooperative forest fire detection and monitoring tasks, the UAVs should be operated at a safe height over the flames of the fire to ensure their safety. However, the flames may reach the UAVs in certain situations, which may cause partial or full damage. Furthermore, in some long-term cooperative tasks, one or more UAV(s) may leave the team to recharge, due to their limited battery energy. Hence, designing a fault-tolerant algorithm able to deal with redundancy and actuator faults occurring in one or more UAVs is important.

Based on the fault-tolerant detection design [35], the following equation reflects the value of the loss of effectiveness of the ith UAV:

$$0 \leq \Omega_i \leq 1, \quad \forall i \in \{1, \ldots, N\}. \tag{30}$$

Let $\hat{\Omega}$ be a threshold value of the detected faults. Note that $\hat{\Omega}$ can be set according to the types of the UAVs or the requirements of the tasks. Based on the value of Ω_i, the following cases may occur:

- If $\Omega_i = 0$ $\forall i$, then all UAVs are fault-free. Hence, the swarm continues with its predefined cooperative tracking mission.
- If there exists $0 < \Omega_i \leq \hat{\Omega}$, then one or more UAVs are subject to internal faults, but are still able to complete the given task with degraded performance. Hence, the UAV swarm will continue the task.
- If there exists $\hat{\Omega} < \Omega_i \leq 1$, then one or more UAVs are suffering from actuator faults and thus cannot accomplish the mission due to the severity of the faults. In this case, the remaining, healthy UAVs must reconfigure their formation based on the new situation.

In this work, we mainly focus on the third case. Assume that a fault is detected in one of the UAVs at time t_f, then the faulty UAV interrupts its communication link with the other UAVs and leaves the collaborative swarm to land safely. The dynamic circular formation for the remaining UAVs can be altered by

$$f_j(t) = \begin{bmatrix} R\sin(wt + \frac{2(j-1)\pi}{N-1}) \\ wR\cos(wt + \frac{2(j-1)\pi}{N-1}) \\ R\cos(wt + \frac{2(j-1)\pi}{N-1}) \\ -wR\sin(wt + \frac{2(j-1)\pi}{N-1}) \end{bmatrix}, \quad \forall t > t_f, \tag{31}$$

for all $j \in \{1, \ldots, N-1\}$.

The target position for the remaining UAVs can then be given by

$$T_j = \begin{bmatrix} T_{xj} \\ T_{yj} \end{bmatrix} = \begin{bmatrix} p_{x0} + R\sin(wt + \frac{2(j-1)\pi}{N-1}) \\ p_{y0} + R\cos(wt + \frac{2(j-1)\pi}{N-1}) \end{bmatrix}. \tag{32}$$

Based on the current positions of the UAVs and the target locations, an optimal task allocation problem can be formulated, where each UAV is assigned one of the target slots T_j based on the travel

costs and hence selects a suitable dynamic formation vector from the list. The task allocation algorithm is triggered every time a UAV leaves the team, and an optimal solution can be calculated based on local information through the network.

For designing the task allocation algorithm, the Hungarian algorithm [36] can be used to reduce the complexity of finding the optimal assignment from combinatorial to polynomial in time. However, such a centralized algorithm has some limitations on scalability and robustness. Therefore, motivated by [37], in this study, we use a decentralized Hungarian algorithm for the formation task reallocation as shown in Algorithm 2.

Algorithm 2 Decentralized Hungarian algorithm for task reassignment.

1: **repeat**
2:     **for** each UAV $i \in \{1, \ldots, N-1\}$ **do**
    // Initialization
3:         form a cost matrix C^i, where each element C^i_{rj} denotes the distance between UAV r to target j at t_f
4:         **for** $r \in \{1, \ldots, N-1\}, j \in \{1, \ldots, N-1\}$ **do**
5:             **if** $r = i$ **then**
6:                 $C^i_{rj} \leftarrow \sqrt{(p_{xi} - T_{xj})^2 + (p_{yi} - T_{yj})^2}$
7:             **else**
8:                 $C^i_{rj} \leftarrow \infty$
9:             **end if**
10:         **end for**
    // Task allocation
11:         apply Hungarian Algorithm [36]
12:         connect with neighbor UAV k and receive C^k
13:         update C^i following:
14:         **if** $C^k_{rj} \neq \infty$ **then**
15:             $C^i_{rj} \leftarrow C^k_{rj}$
16:         **end if**
17:         back to step 11
18:     **end for**
19: **until** an optimal assignment is achieved

In Algorithm 2, each UAV scans its own communication network and obtains its own position and the locations of the formation targets. Then based on the received information, the UAV i creates a cost matrix C^i, where the element at the ith row and the jth column reflects the cost of allocating the UAV i to the target location j. The decentralized Hungarian algorithm iterates between two main stages. The centralized Hungarian algorithm is firstly implemented in each UAV. The objective of this stage is to guarantee a target location is assigned to each UAV during the mission. In the second stage, UAVs connect with their neighbors to exchange cost matrices and update their local cost matrix accordingly. When the initial local data of a UAV is globally transmitted to all the other UAVs via the connected network, all the members in the team will eventually have the same cost matrix.

Note that the number of steps that it takes for the convergence depends on the network connectivity. That is to say, the more complex the graph topology is, the shorter time the decisions of the remaining healthy UAVs will achieve consensus. However, considering the bandwidth requirement and the communication costs, a trade-off between the convergence time and the complexity of the network should be determined based on the applications and available resources.

3.3. Collision avoidance algorithm

During the cooperation tasks, UAVs in the swarm should avoid potential collision risks using the onboard sensory information. A geometry-based collision avoidance algorithm is proposed to navigate each UAV to ensure that they avoid both static and dynamic obstacles.

3.3.1. Collision detection strategy

When a UAV is following a predefined path, it calculates the potential collision risk using a collision detection strategy. Each obstacle is covered by a virtual safe circle. The obstacle is denoted by o and its speed is represented by $\overrightarrow{v_o}$. A UAV is denoted by u with its speed $\overrightarrow{v_u}$. The UAV position is denoted as (p_{xu}, p_{yu}). Obstacles position is represented by (p_{xo}, p_{yo}). The relative speed between the UAV and obstacle is denoted by $\overrightarrow{v_r}$. UAV speed $\overrightarrow{v_u}$, obstacle speed $\overrightarrow{v_o}$ and relative speed $\overrightarrow{v_r}$ form the triangle, as shown in Fig. 3(a). Then we have

$$\overrightarrow{v_r} = \overrightarrow{v_u} - \overrightarrow{v_o}. \tag{33}$$

The angle between $\overrightarrow{v_r}$ and line uo is denoted by θ. The distance between obstacle o and the extension line of vector $\overrightarrow{v_r}$ is represented by r_{mo}, which is referred to as the Closest Approach Distance (CAD). The tangent line is denoted by un. n represents the tangent point. The distance between o and n can be denoted by r_{no}, which is also the radius of the safety circle that is a user-defined parameter. The distance between an obstacle and UAV is represented by r_{uo} and this obstacle can be considered a collision risk when

$$r_{mo} < r_{no}, \tag{34}$$

where

$$r_{mo} = r_{uo} \sin(\theta), \tag{35}$$
$$r_{uo} = \sqrt{(p_{xu} - p_{xo})^2 + (p_{yu} - p_{yo})^2}. \tag{36}$$

The distance between a UAV and Closest Point of Approach (CPA) m is denoted by r_{um}. The Time to Closest Point of Approach (TCPA) t_{tcpa} can be calculated by

$$r_{um} = r_{uo} \cos(\theta), \tag{37}$$
$$t_{tcpa} = \frac{r_{um}}{|\overrightarrow{v_r}|}. \tag{38}$$

When collision risk occurs and t_{tcpa} is less than a user-defined time threshold, the collision resolution will be triggered.

3.3.2. Collision resolution strategy

The tangent point of a safety circle is chosen as the collision resolution waypoint. The tangent point (x, y) can be calculated by solving the following equations [38]

$$(x - x_o)^2 + (y - y_o)^2 = r^2, \tag{39}$$
$$(y - y_o)(y - y_u) = -(x - x_o)(x - x_u), \tag{40}$$

where UAV position is denoted by (x_u, y_u), obstacle position is represented by (x_o, y_o), and the safety circle radius is r. Two tangent points are denoted using p and q. The right side tangent point q is used as a collision resolution waypoint.

3.3.3. Waypoint guidance algorithm

In order to navigate a UAV to the collision resolution waypoint precisely under external disturbances, such as wind, a Proportional Navigation (PN) guidance algorithm [39] is implemented to manoeuvre the UAV to the collision resolution waypoint by regulating the lateral acceleration of the UAV, $\overrightarrow{a_u}$ that can be calculated using

$$\overrightarrow{a_u} = \varrho_r \dot{\theta}_r \overrightarrow{v_r}, \tag{41}$$

where ϱ_r denotes a user-defined proportional navigation constant rate. $\dot{\theta}_r$ represents the rotating rate of relative velocity $\overrightarrow{v_r}$. The angular velocity of UAV $\dot{\theta}_u$ can be calculated by
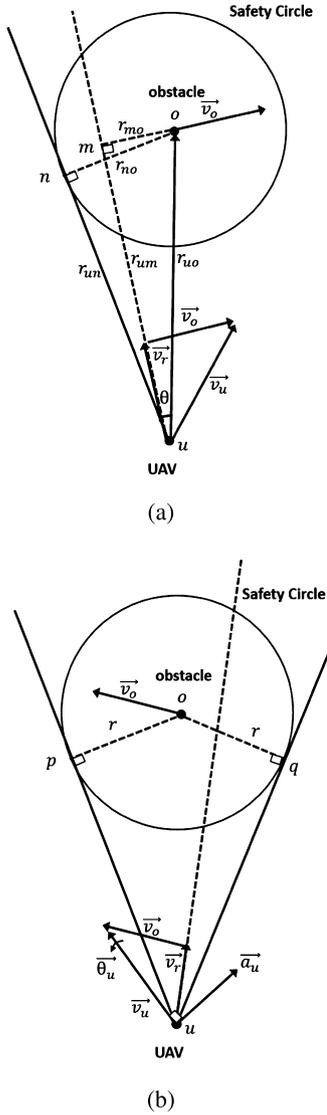
(a)

(b)

Fig. 3. (a) Collision detection. (b) Collision resolution.

$$\dot{\theta}_u = \frac{a_u}{v_u}. \tag{42}$$

The pseudocode of the collision avoidance algorithm is provided in Algorithm 3.

Algorithm 3 Geometry-based collision avoidance algorithm.

1: **repeat**
2:    **for** each UAV $i \in \{1, \ldots, N\}$ **do**
3:       Check collision risk using relative speed and position between each drone and obstacle via (33)-(36)
4:       **if** Collision risk occurs **then**
5:          Check Time to Closest Point of Approach (TCPA) by using (37) and (38)
6:          **if** TCPA is less than a user-defined threshold **then**
7:             Calculate collision resolution waypoint q using (39) and (40) and calculate angular velocity command $\dot{\theta}_u$ of UAV using (41) and (42)
8:          **end if**
9:       **end if**
10:    **end for**
11: **until** mission is completed
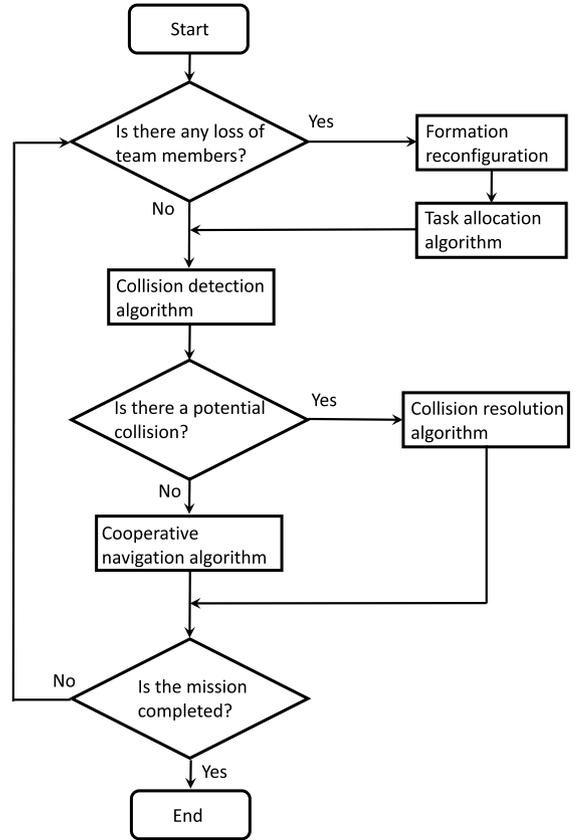
Fig. 4. The flow diagram of the proposed coordination scheme.

3.4. Unified collaboration framework

Following the results proposed in Section 3.1-3.3, an algorithm processing procedure diagram is given in Fig. 4.

The proposed cooperation framework for networked UAVs has the potential to be extended and applied to some more complex tasks. Note that more powerful and state-of-the-art swarm intelligence algorithms can also be integrated with the proposed framework to guarantee safe navigation in extreme environments. In that scenario, based on task characteristics, the low-level algorithms implemented in the UAVs may be different, but the proposed framework used as a high-level coordination strategy will remain the same.

4. Simulation case study: a forest fire monitoring mission

In this section, a simulated forest fire monitoring mission is provided to verify the effectiveness of the proposed fault-tolerant cooperative guidance strategy.

A forest fire detection and monitoring mission was configured using a swarm of six UAVs. Different types of empirical models have been proposed to describe the spatial behavior of fire. In the simulation, the spiral model proposed in [40] was adopted:

$$\begin{cases} \dot{p}_0(t) = v_0(t) \\ \dot{v}_0(t) = \begin{bmatrix} -2a\sin(t) - at\cos(t) \\ 2b\cos(t) - bt\sin(t) \end{bmatrix} \end{cases}$$

where the parameters $a = 0.1$ and $b = 0.2$ reflect the fire spread rates. Furthermore, the following unknown disturbances were added to the dynamics of the UAVs

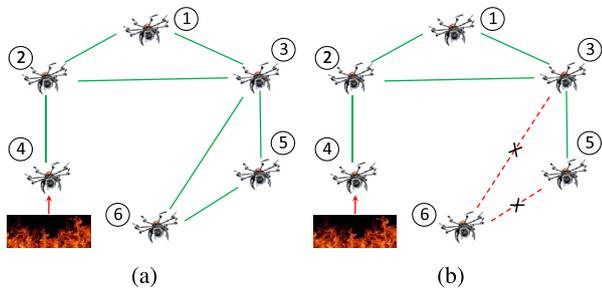$$d_i(t) = (-1)^i \begin{bmatrix} 8\sin(2t) \\ 0.1t\cos(2t) \end{bmatrix}.$$

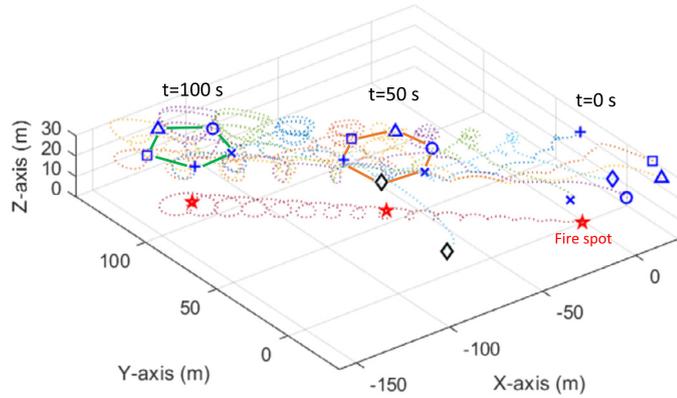**Fig. 5.** Communication topology among the UAVs. (a) Before 50 s, and (b) after 50 s.

Fig. 6. Position trajectories of the UAVs in the 3D space forming a dynamic circular formation around the moving fire spot. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

Fig. 7. Time variation of the formation tracking error. (a) Approach in the current paper. (b) Approach in [41].

At the beginning of the mission, all the UAVs were connected by a sparse communication network as shown in Fig. 5(a). Based on the proposed fire detection algorithm and the coordination strategy, the UAV swarm was expected to form a dynamic hexagon formation around the moving fire spot. However, at $t = 50$ s, UAV 6 experienced a hardware fault and hence left the collaborative UAV swarm to land safely. At the same time, it terminated its communication with the other UAVs. The updated communication topology is illustrated in Fig. 5(b). Due to the unexpected fault, the remaining UAVs began a task reallocation and formation reconfiguration operation using the decentralized Hungarian algorithm and quickly attained a new pentagon formation around the fire. The trajectories of the UAVs are shown in Fig. 6, where the positions of the UAVs are represented by different blue symbols respectively. At $t = 50$ s, one of the UAVs stopped functioning due to a sudden fault, which is marked by the black diamond symbol. The formation tracking error of each UAV is provided in Fig. 7(a), where some additional tracking errors can be observed at $t = 50$ s due to formation reconstruction. It can be observed that the formation tracking error can be quickly eliminated in the presence of actuator faults, the uncertain movement of the fire and loss of a team member.

To indicate the superior robust performance of the proposed scheme under unknown disturbances, the same mission was performed using the coordinated protocol proposed in [41] for comparison. The corresponding formation tracking error is illustrated in Fig. 7(b), from which it can be seen that the tracking error diverges due to the effect of the injected unknown disturbances and the target's uncertain movement. From all these figures, it can be concluded that the fire detection and monitoring mission can be effectively accomplished under the proposed fault-tolerant cooperative guidance strategy.
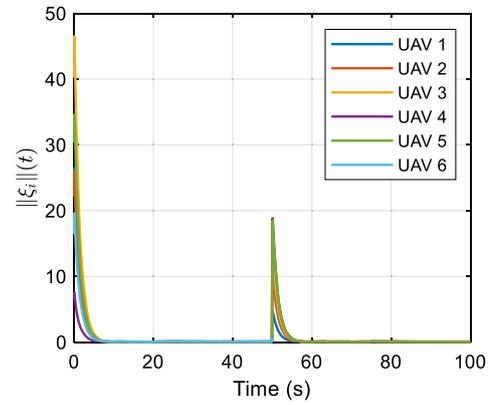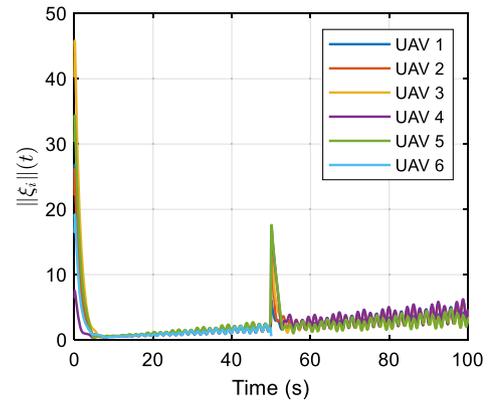
5. Experimental results

5.1. Experimental setup

The proposed cooperative circumnavigation algorithm was validated using three Parrot Bebop drones, as shown in Fig. 8(a). All the drones used the Robot Operating System (ROS), communicating through the WiFi network, with the laptop serving as the ROS master. Bebop autonomy was used as the firmware driver and it used GPS, an Inertial Measurement Unit (IMU), a sonar sensor and an optical flow camera to estimate the attitude and speed of the UAV, as shown in Fig. 8(b) and Fig. 8(c). The ROS driver integrated the visual-inertial velocity estimates, reported by Bebop's firmware driver, to calculate the odometery that contains both position and velocity information. The laptop was set as the ROS master and each drone was configured as the ROS client. Each drone updated its odometery, speed and yaw angle at a specified frequency (i.e., 2 Hz). The ROS master subscribed these topics and calculated the speed and yaw angle command for each drone. These commands were published through ROS driver and each drone subscribed its corresponding command and executed them. The video of the experiments can be found at https:// www.youtube.com/watch?v=6fnPfqqVOvs.

5.2. Experiment 1: Outdoor flight tests with switching network topology

In this section, a simplified circular formation tracking experiment was conducted to verify the feasibility of the proposed robust navigation strategy in a real-world environment.
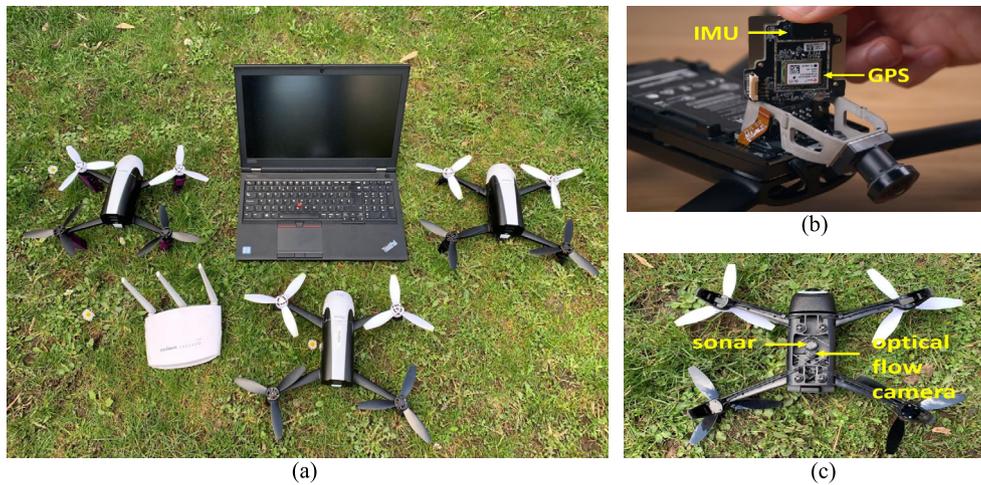
Fig. 8. (a) Multi-drone formation experimental platform. (b) IMU and GPS sensors. (c) Sonar and Optical flow camera.
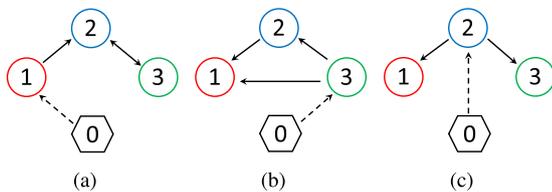

Fig. 9. Switching interaction topologies, where the virtual target is labeled as '0'.
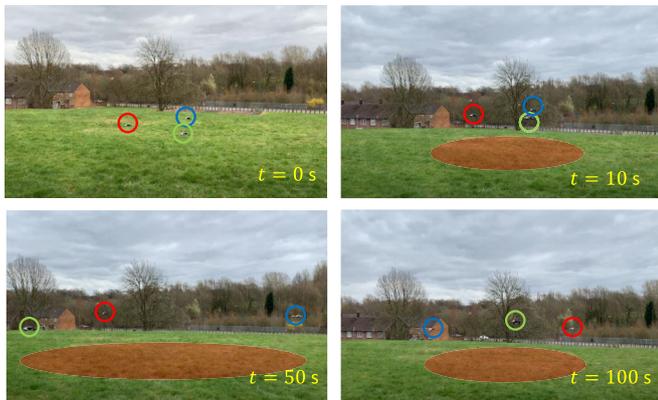

Fig. 10. Snapshots at different time instants during the outdoor flight tests.
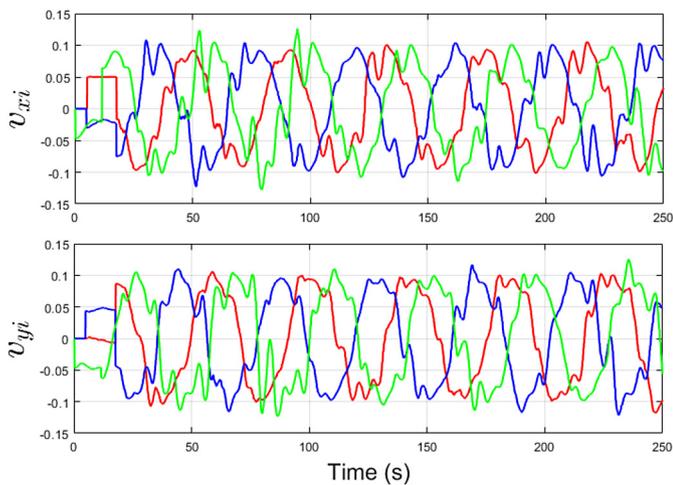

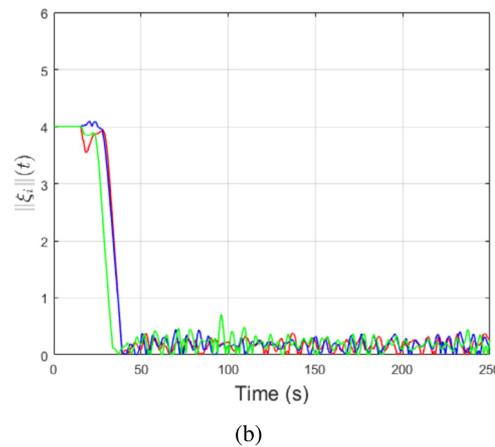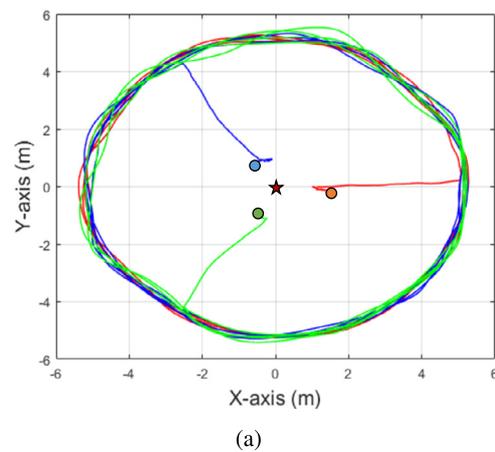**Fig. 11.** Velocities of the UAVs along the X-axis and the Y-axis.

Fig. 12. (a) 2D trajectories of three drones. (b) Time variation of the formation tracking error of three Parrot Bebop drones.

The drones were controlled to fly around the center $(0, 0)$ and the flight radius was set as $R = 5$ m. The interaction topology among the three UAVs is shown in Fig. 9(a)-(c), where the communication topology is switched every 50 s in sequence. The progress during the hardware experiment on cooperative navigation is illustrated using several real-time snapshots shown in Fig. 10. The velocities v_{xi} and v_{yi} of the three drones, as recorded during $t = 250$ s of flight, are represented by green, blue and red lines in Fig. 11. Since the desired formation is time-varying, some periodic motions can be observed in the velocity responses. The 2D trajectories are
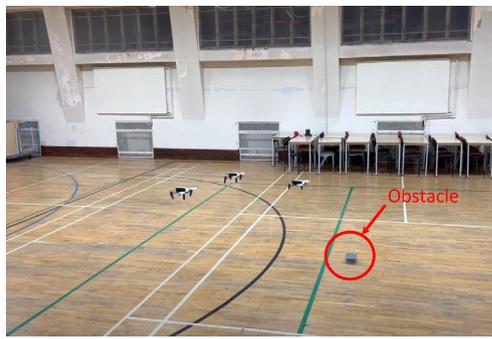
(a)
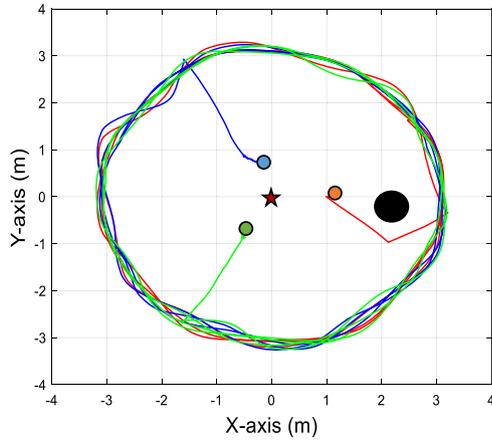
(b)

Fig. 13. (a) The environment of the experiment in the presence of an obstacle (b) 2D trajectories of three drones with collision avoidance.


(a)   (b)

Fig. 14. Experiment 3 shows that a triangular formation is first attained by three drones, which is then switched to a line formation after a sudden loss of one drone.

shown in Fig. 12(a). The time variation of the formation tracking error is shown in Fig. 12(b). From all these figures, it can be concluded that the formation tracking mission is accomplished by the networked drones using the proposed cooperative navigation strategy in the presence of disturbances such as wind perturbation and actuator noises.

5.3. Experiment 2: Robustness with obstacle avoidance

To test the performance of the proposed collaboration scheme in the presence of potential obstacles, another experiment has been conducted. Similar to the setup of the Experiment 1, we used three drones to form a time-varying circular formation around a static target. As shown in Fig. 13(a), an obstacle was added to the environment. Since each drone was not equipped with advanced sensors to detect such obstacles, the position and the size of the obstacle were known in prior in the controller design. Because the control law relies only on the local information, the present experimental setup is suitable for validating the feasibility of the proposed collision avoidance scheme. The trajectories of the drones
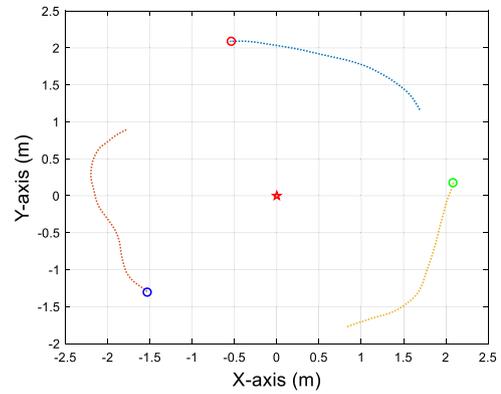
(a)
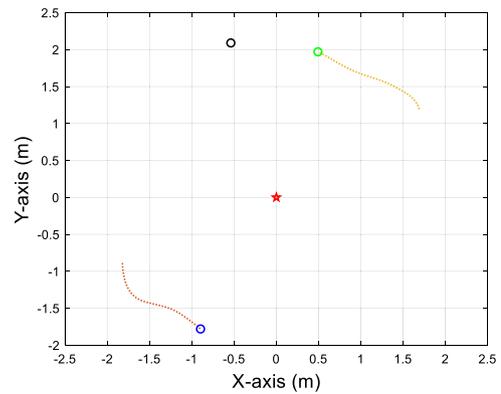
(b)

Fig. 15. Position trajectories of the drones during the formation tracking mission in Experiment 3. (a) Before the sudden fault. (b) After the sudden fault.

in the mission is shown in Fig. 13(b). Compare Fig. 13(b) with Fig. 12(a), it can be easily seen that all the drones have successfully avoided the potential collisions while reaching the desired destination.

5.4. Experiment 3: Fault-tolerance with loss of member

This experiment was performed with three drones to test the fault-tolerance of the proposed cooperative navigation scheme to a sudden loss of drones. The experiment started with the goal of achieving a triangular formation by the drones with respect to the given virtual target. Fig. 14(a) shows that the desired triangular formation was attained by those three drones surrounding the virtual target. Fig. 14(b) illustrates a situation when one of the drones (marked by the black circle) stopped functioning due to a sudden fault and the remaining two drones were able to tackle the situation and attained a new line formation with respect to the given target. Fig. 15(a) and Fig. 15(b) complement the results depicted in Fig. 14(a) and Fig. 14(b) by showing the spatial variation of the position trajectories of the drones during the experiment.

5.5. Experiment 4: Formation tracking with a moving target

This experiment was carried out using a group of three drones to test the formation trajectory tracking with a moving target. Fig. 16(a)–(c) presents the hardware experimental results on achieving the formation surrounding a time-varying virtual target (marked by the red star). The initial and final positions of the drones at each time instant are represented by diamond and circle symbols, respectively. The trajectories of the drones during the mission are shown by dotted lines. From all these figures, it can
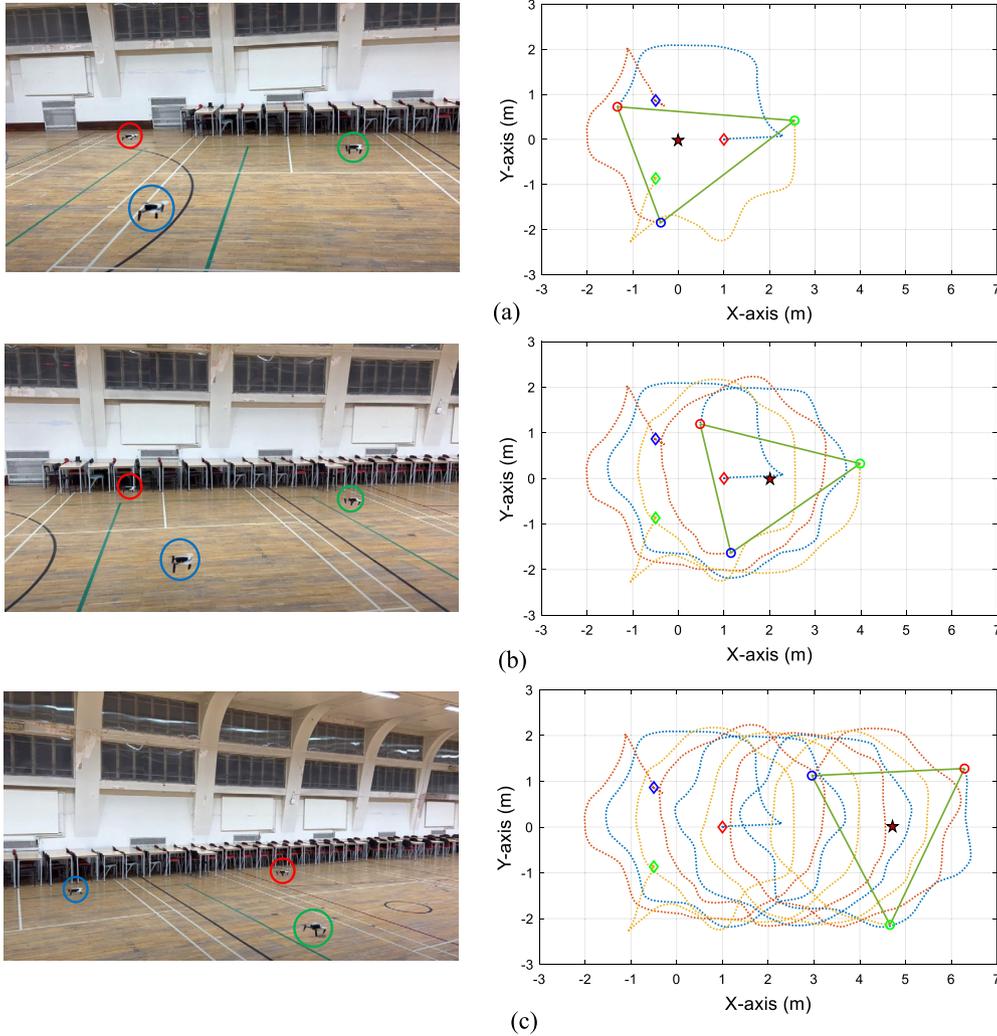
Fig. 16. Progress of the formation tracking mission being achieved by a team of three networked drones with a virtual moving target during Experiment 4. (a) $t = 25$ s. (b) $t = 100$ s. (c) $t = 180$ s.

be concluded that the proposed cooperative navigation scheme is effective while tracking a moving target.

6. Conclusions

In this paper, a cooperative navigation strategy was proposed for networked unmanned aerial vehicles (UAVs). Firstly, a novel adaptive circular formation control protocol was developed for UAV swarms, which considered the effect of unknown disturbances. A novel task reassignment algorithm and a formation reconfiguration method were also designed to handle actuator faults that occurred in one or more UAVs during the mission. Finally, a geometry based collision avoidance algorithm was proposed for each UAV to avoid potential collision risks, using onboard sensory information. A forest fire monitoring simulation case study was conducted to indicate one of the potential applications of the proposed navigation strategy. Finally, an outdoor flight experiment and several indoor flight experiments were conducted to validate the effectiveness and feasibility of the proposed navigation strategy.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] D. Li, K. Cao, L. Kong, H. Yu, Fully distributed cooperative circumnavigation of networked unmanned aerial vehicles, IEEE/ASME Trans. Mechatron. 26 (2) (2021) 709–718.

[2] Z. Zhen, Y. Chen, L. Wen, B. Han, An intelligent cooperative mission planning scheme of UAV swarm in uncertain dynamic environment, Aerosp. Sci. Technol. 100 (2020) 105826.

[3] Z. Zhen, D. Xing, C. Gao, Cooperative search-attack mission planning for multi-UAV based on intelligent self-organized algorithm, Aerosp. Sci. Technol. 76 (2018) 402–411.

[4] J. Yu, X. Dong, Q. Li, Z. Ren, Distributed cooperative encirclement hunting guidance for multiple flight vehicles system, Aerosp. Sci. Technol. 95 (2019) 105475.

[5] J. Hu, P. Bhowmick, I. Jang, F. Arvin, A. Lanzon, A decentralized cluster formation containment framework for multirobot systems, IEEE Trans. Robot. 37 (6) (2021) 1936–1955.

[6] Y. Liu, F. Zhang, P. Huang, X. Zhang, Analysis, planning and control for cooperative transportation of tethered multi-rotor UAVs, Aerosp. Sci. Technol. 113 (2021) 106673.

[7] Z. Terze, V. Pandža, M. Kasalo, D. Zlatar, Discrete mechanics and optimal control optimization of flapping wing dynamics for mars exploration, Aerosp. Sci. Technol. 106 (2020) 106131.

[8] Y. Liu, H. Liu, Y. Tian, C. Sun, Reinforcement learning based two-level control framework of UAV swarm for cooperative persistent surveillance in an unknown urban area, Aerosp. Sci. Technol. 98 (2020) 105671.

[9] J. Chen, C. Du, Y. Zhang, P. Han, W. Wei, A clustering-based coverage path planning method for autonomous heterogeneous UAVs, IEEE Trans. Intell. Transp. Syst. (2021) 1–11.

[10] K.A. Ghamry, Y. Zhang, Fault-tolerant cooperative control of multiple UAVs for forest fire detection and tracking mission, in: 2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol), IEEE, 2016, pp. 133–138.

[11] D.W. Casbeer, D.B. Kingston, R.W. Beard, T.W. McLain, Cooperative forest fire surveillance using a team of small unmanned air vehicles, Int. J. Syst. Sci. 37 (6) (2006) 351–360.

[12] Z. Lin, H.H. Liu, M. Wotton, Kalman filter-based large-scale wildfire monitoring with a system of uavs, IEEE Trans. Ind. Electron. 66 (1) (2019) 606–615.

[13] J. Li, Y. Xiong, J. She, M. Wu, A path planning method for sweep coverage with multiple uavs, IEEE Int. Things J. 7 (9) (2020) 8967–8978.

[14] H.X. Pham, H.M. La, D. Feil-Seifer, M.C. Deans, A distributed control framework of multiple unmanned aerial vehicles for dynamic wildfire tracking, IEEE Trans. Syst. Man Cybern. Syst. 50 (4) (2020) 1537–1548.

[15] H. Liu, X. Li, M. Fan, G. Wu, W. Pedrycz, P.N. Suganthan, An autonomous path planning method for unmanned aerial vehicle based on a tangent intersection and target guidance strategy, IEEE Trans. Intell. Transp. Syst. (2020) 1–13, in press.

[16] R. Hartley, B. Kamgar-Parsi, C. Narber, Using roads for autonomous air vehicle guidance, IEEE Trans. Intell. Transp. Syst. 19 (12) (2018) 3840–3849.

[17] X. Wang, W. Wang, Extended signal-correction observer and application to aircraft navigation, IEEE Trans. Ind. Electron. 67 (4) (2019) 3149–3156.

[18] J. Hu, H. Niu, J. Carrasco, B. Lennox, F. Arvin, Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning, IEEE Trans. Veh. Technol. 69 (12) (2020) 14413–14423.

[19] J. Hu, A.E. Turgut, B. Lennox, F. Arvin, Robust formation coordination of robot swarms with nonlinear dynamics and unknown disturbances: design and experiments, IEEE Trans. Circuits Syst. II, Express Briefs 69 (1) (2022) 114–118.

[20] B. Zhu, A.H.B. Zaini, L. Xie, Distributed guidance for interception by using multiple rotary-wing unmanned aerial vehicles, IEEE Trans. Ind. Electron. 64 (7) (2017) 5648–5656.

[21] J. Hu, P. Bhowmick, A. Lanzon, Group coordinated control of networked mobile robots with applications to object transportation, IEEE Trans. Veh. Technol. 70 (8) (2021) 8269–8274.

[22] S. Li, X. Fang, A modified adaptive formation of UAV swarm by pigeon flock behavior within local visual field, Aerosp. Sci. Technol. 114 (2021) 106736.

[23] T. Yang, Z. Jiang, R. Sun, N. Cheng, H. Feng, Maritime search and rescue based on group mobile computing for unmanned aerial vehicles and unmanned surface vehicles, IEEE Trans. Ind. Inform. 16 (12) (2020) 7700–7708.

[24] İ. Bayezit, B. Fidan, Distributed cohesive motion control of flight vehicle formations, IEEE Trans. Ind. Electron. 60 (12) (2012) 5763–5772.

[25] X. Dong, Y. Zhou, Z. Ren, Y. Zhong, Time-varying formation tracking for second-order multi-agent systems subjected to switching topologies with application to quadrotor formation flying, IEEE Trans. Ind. Electron. 64 (6) (2016) 5014–5024.

[26] J. Wang, M. Xin, Integrated optimal formation control of multiple unmanned aerial vehicles, IEEE Trans. Control Syst. Technol. 21 (5) (2012) 1731–1744.

[27] X. Fu, J. Pan, H. Wang, X. Gao, A formation maintenance and reconstruction method of UAV swarm based on distributed control, Aerosp. Sci. Technol. (2020) 105981.

[28] J. Hu, W.X. Zheng, Adaptive tracking control of leader–follower systems with unknown dynamics and partial measurements, Automatica 50 (5) (2014) 1416–1423.

[29] H. Yu, X. Xia, Adaptive consensus of multi-agents in networks with jointly connected topologies, Automatica 48 (8) (2012) 1783–1790.

[30] J. Sun, Z. Geng, Adaptive consensus tracking for linear multi-agent systems with heterogeneous unknown nonlinear dynamics, Int. J. Robust Nonlinear Control 26 (1) (2016) 154–173.

[31] Z. Li, Z. Duan, G. Chen, L. Huang, Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint, IEEE Trans. Circuits Syst. I, Regul. Pap. 57 (1) (2009) 213–224.

[32] H.K. Khalil, Nonlinear Systems, Prentice-Hall, New Jersey, USA, 1996.

[33] H. Liu, Y. Wang, F.L. Lewis, K.P. Valavanis, Robust formation tracking control for multiple quadrotors subject to switching topologies, IEEE Trans. Control Netw. Syst. 7 (3) (2020) 1319–1329.

[34] D. Liberzon, Switching in Systems and Control, Springer Science & Business Media, 2003.

[35] M.A. Kamel, X. Yu, Y. Zhang, Real-time fault-tolerant formation control of multiple wmrs based on hybrid ga-pso algorithm, IEEE Trans. Autom. Sci. Eng. 18 (3) (2021) 1263–1276.

[36] H.W. Kuhn, The hungarian method for the assignment problem, Nav. Res. Logist. Q. 2 (1–2) (1955) 83–97.

[37] S. Ismail, L. Sun, Decentralized hungarian-based approach for fast and scalable task allocation, in: International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, 2017, pp. 23–28.

[38] A. Savvaris, H. Niu, H. Oh, A. Tsourdos, Development of collision avoidance algorithms for the c-enduro usv, IFAC Proc. Vol. 47 (3) (2014) 12174–12181.

[39] H. Niu, A. Savvaris, A. Tsourdos, USV geometric collision avoidance algorithm for multiple marine vehicles, in: OCEANS 2017-Anchorage, IEEE, 2017, pp. 1–10.

[40] G. Perry, Current approaches to modelling the spread of wildland fire: a review, Prog. Phys. Geogr. 22 (2) (1998) 222–245.

[41] J. Hu, P. Bhowmick, A. Lanzon, Distributed adaptive time-varying group formation tracking for multiagent systems with multiple leaders on directed graphs, IEEE Trans. Control Netw. Syst. 7 (1) (2020) 140–150.