

A k -hop Constrained Reachability Based Proactive Connectivity Maintaining Mechanism of UAV Swarm Networks

Huibin Wang^{1,2*}, Ming Chen³, Xianglin Wei⁴

¹ College of Computer and Information Engineering, Chuzhou University, China

² Command and Control Engineering College, Army Engineering University of PLA, China

³ College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China

⁴ 63rd Research Institute, National University of Defense Technology, China
wanghuibin@chzu.edu.cn, mingchennj@163.com, wei_xianglin@163.com

Abstract

Harsh or hostile environments may lead to node failure and connectivity degradation of a UAV swarm system. In order to maintain or restore the connectivity of a network in case of node failure, this paper proposes a mechanism to adjust the network topology to resist the impact caused by node failure. Firstly, a network model of a UAV swarm network based on k -hop constrained reachability is proposed. Secondly, a k -hop constrained reachability based proactive connectivity maintaining mechanism of UAV swarm network is presented. In this mechanism, each node identifies the network abnormality distributed according to k -hop reachability, and reports the observed abnormality to the master node; then, a virtual edge-based topology reconstruction algorithm is put forward for the master node to derive a topology reconstruction solution in a centralized way; afterwards, the solution is delivered to the slave nodes to reconfigure the network topology in parallel. Thirdly, a quantitative method is introduced to optimize the total travel distance of nodes, and a spanning tree-based method is designed to maintain the connectivity during the topology transformation process. Both theoretical analysis and simulation results have shown that: on the one hand, the proposed mechanism are effective in maintaining a UAV swarm's connectivity in case of node failure; on the other hand, the proposed mechanism outperforms existing mechanisms in terms of fault tolerance, connectivity, and total travel distance, and it's less affected by the failure rate.

Keywords: UAV swarm network, Network reconstruction, Connectivity maintaining, Algebraic connectivity

1 Introduction

Unmanned Aerial Vehicle (UAV) swarms composed of miniaturized UAV nodes have gained significant attention in recent years due to their unique advantages, including cost efficiency, strong scalability, and high survivability. These advantages stem from the collaborative efforts of UAVs, which rely on their underlying UAV swarm network (USNET). However, maintaining connectivity in USNETs can be challenging, especially in unattended or hostile

environments where UAV swarms are typically deployed. In such harsh conditions, swarm nodes may get damaged, and wireless links among them may fail due to limited power supply, malicious damage, and other factors. When multiple nodes fail in a UAV swarm, the structure of the swarm may become compromised, leading to reduced survivability [1]. Therefore, it is critical to develop mechanisms to maintain USNET connectivity in the event of node failures.

Existing connectivity maintenance strategies under node failure could be divided into two categories: reactive and proactive [2]. A reactive mechanism aims to restore connectivity after detecting network segmentation. Typical reactive proposals include repositioning nodes [3-5], Data Mule [6-7], and cooperative communication [8]. In contrast, a proactive mechanism strives to provision resources both at setup and during normal operation to prevent network segmentation when node failure happens. Specifically, existing connectivity maintenance methods can be categorized into four groups: node deployment optimization [9-10], backup node designation [11], dynamic communication range adjustment [12], and controllable node mobility [13-16]. However, existing proactive proposals could not fully solve the connectivity concerns of USNET in two aspects: on the one hand, existing solutions cannot well adapt to the characteristics of USNET, such as time-varying topology, limited payload, controllable movement in 3-D space, and high unit price; on the other hand, there is no easy-to-calculate metric that can quantify local network connectivity.

In this backdrop, to solve the above-mentioned two problems, this paper puts forward a k -hop constrained reachability based proactive connectivity maintaining mechanism (PCM) for maintaining a USNET's connectivity. At first, a novel metric is defined, named k -hop constrained local algebraic connectivity, to characterize the connectivity of a USNET to facilitate failure detection. A failure event will be reported to the master node of the swarm once a node's k -hop constrained local algebraic connectivity is lower than a threshold. Secondly, the master node of the swarm will initiate the topology reconstruction process, in which a reconstructed topology will be calculated through establishing several virtual links between isolated or weak-connected parts caused by failure. Thirdly, to minimize all nodes' total travel distance, an optimization problem is

*Corresponding Author: Huibin Wang; E-mail: wanghuibin@chzu.edu.cn

formulated and a travel distance optimization method based on geometric is proposed. The main contributions of this paper are as follows:

- A distributed abnormal identification algorithm based on k -hop reachability is proposed. For this purpose, a novel metric, named k -hop constrained local algebraic connectivity, is defined that could better profile the connectivity status of a USNET than traditional global algebraic connectivity.
- A virtual edge-based topology reconstruction algorithm is put forward, in which the nodes connected by virtual edges will move towards each other due to the attractive force on the edge, to refine the lower bound of minimum local algebraic connectivity. The traditional potential-field based connectivity maintaining algorithm improve connectivity by shortening the distance between the nodes connected by physical link, it may fall into local optimum due to the minimum inter-node distance constraint. The method of constructing virtual edges can make the nodes that originally exceed the communication range approach each other, thereby solving the local optimum problem.
- An optimization method based on geometric is introduced to shorten the total travel distance during network transformation; and a spanning tree-based connectivity maintaining method is designed to maintain the network connectivity during the transformation process.

The remaining of this paper is organized as follows. Section 2 summarizes related work. Section 3 formulates the connectivity maintenance problem. Section 4 details the proposed PCM mechanism and related algorithms. In Section 5, extensive simulation experiments are presented to verify the proposed PCM mechanism and evaluate its performance. Finally, Section 6 presents the conclusion.

2 Problem Formulation

This section firstly introduces the USNET model and defines three running statuses of a USNET, then the connectivity maintenance problem is formulated.

2.1 USNET Model

A USNET at a particular time t could be described as a time-varying undirected graph $G(t) = (U(t), E(t))$, where $U(t) = \{u_i \mid i = 1, 2, \dots, n\}$ denotes n UAV nodes, $E(t) = \{e_{ij} \mid u_i \in U(t), u_j \in U(t)\}$ refers to the bidirectional wireless links between nodes u_i and u_j in the network. Each node could perceive the connection status of all its wireless links with its neighboring nodes, and get its own location through GPS. A node is appointed as the master node, denoted as u_{master} , and the remaining nodes are slave nodes. The master node could perceive the state of the network, including network topology and connectivity.

Let $q_i(t) \in \mathbb{R}^3$ and $v_i(t) \in \mathbb{R}^3$ denote the position and velocity of node u_i at time t , respectively. Let $d_{ij}(t) = \|q_i(t) - q_j(t)\|$ denote the Eulerian distance between u_i and u_j .

Assumed that all nodes have the same transmission range R ; therefore, an edge $e_{ij} \in E(t)$ exists if and only if $d_{ij}(t) \leq R$.

Definition 1. (*k-hop neighboring set*): Let $N_i^k(t)$ be the set of k -hop neighboring nodes of node u_i at time t . That is,

$$N_i^k(t) = \begin{cases} \{u_j \mid u_j \in U(t), e_{ij} \in E(t)\}, & k=1 \\ \bigcup_{u_j \in N_i^{k-1}(t)} N_j^1(t), & k > 1 \end{cases} \quad (1)$$

Let $N_i^+(t) = \{u_i\} \cup N_i^1(t)$ denote the set of 1-hop neighboring nodes of node u_i that contain itself.

Let $A_G(t)$ denote the weighted adjacency matrix of the time-varying undirected graph $G(t)$. That is,

$$[A_G(t)]_{ij} = \begin{cases} f(d_{ij}(t)), & \text{if } u_j \in N_i^+(t) \\ 0, & \text{otherwise} \end{cases}, \quad (2)$$

where $f(d_{ij}(t)) > 0$ is a sigmoid function about the distance $d_{ij}(t)$. Let $L_G(t)$ denote the weighted Laplacian matrix of the time-varying undirected graph $G(t)$. That is,

$$[L_G(t)]_{ij} = \begin{cases} \sum_{u_j \in N_i(t)} [A_G(t)]_{ij}, & \text{if } i = j \\ -[A_G(t)]_{ij}, & \text{otherwise} \end{cases}. \quad (3)$$

$L_G(t)$ is a symmetric positive semi-definite matrix. Let $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ be the eigenvalues of $L_G(t)$, the second smallest eigenvalue λ_2 is referred to as algebraic connectivity. Let $\lambda_G(t)$ denote the algebraic connectivity of the graph $G(t)$, $\lambda_G(t) = 0$ indicates that the network is disconnected at time t , i.e., the network is partitioned into at least two disjoint subnets.

Algebraic connectivity could profile the global connectivity of a USNET, and is suitable for centralized computing. However, the calculation of algebraic connectivity needs to obtain global topology information, which leads to high communication overhead. Besides, algebraic connectivity is strongly related to the number of nodes. For instance, the algebraic connectivity of a ring topology decreases with the number of nodes, that is, a uniform algebraic connectivity value cannot be used to describe networks of different scales with the same topology. We propose a new metric, named k -hop local algebraic connectivity, which is suitable for distributed computing and could provide a relatively uniform metric that is weakly related to the number of nodes.

Let $c_{ij}(t)$ be a Boolean variable representing whether or not u_i and u_j are neighbors. That is,

$$c_{i,j}(t) = \begin{cases} 1, & \text{if } e_{ij} \in E(t) \\ 0, & \text{otherwise} \end{cases}. \quad (4)$$

Definition 2. (*path between u_i and u_j except u_i*): For $\forall u_i, u_j \in U(t) \setminus \{u_i\}$, let p_{ij}^i be the path between u_i and u_j except u_i , that is, $\exists u_{p-1}, u_{p-2}, \dots, u_{p-k-1} \in U(t) \setminus \{u_i\}$, such that,

$$\begin{cases} \prod_{j=0}^{k-1} \mathcal{C}_{p_{-j}, p_{-j+1}}(t) = 1 \\ p_{ij}^l = \{u_{p_{-0}}, u_{p_{-1}}, \dots, u_{p_{-k}}\} \end{cases} \quad (5)$$

where $u_{p_{-0}}$ denotes u_i , and $u_{p_{-k}}$ denotes u_j , $\|p_{ij}^l\|$ denote the length of the path p_{ij}^l .

Let $\Theta_{ij}^l = \{p_{ij}^l\}$ denote the set of all paths between u_i and u_j except u_i , and $\eta_{ij}^l = \min\|p_{ij}^l\|$ denote the length of the shortest path in Θ_{ij}^l .

Definition 3. (k -hop constrained reachability between u_i and u_j except u_i): For $\forall u_i, u_j \in U(t) \setminus \{u_i\}$, $i \neq j$, let $[\gamma_{ij}^l]_k$ denote the k -hop constrained reachability between u_i and u_j except u_i . That is,

$$[\gamma_{ij}^l]_k = \begin{cases} \frac{1}{\eta_{ij}^l}, & \Theta_{ij}^l \neq \emptyset \text{ and } \eta_{ij}^l \leq k \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

Let SG_l denote the subgraph corresponding to the nodes set $N_l^+(t)$, the definition of the k -hop constrained weighted adjacency matrix $A_{SG_l}^k(t)$ corresponding to SG_l is:

$$[A_{SG_l}^k(t)]_{ij} = \begin{cases} [\gamma_{ij}^l]_k \cdot f(d_{ij}(t)), & i \neq j \\ 0, & i = j \\ f(d_{ij}(t)), & \text{otherwise} \end{cases} \quad (7)$$

Let $L_{SG_l}^k(t)$ denote k -hop constrained Laplacian matrix of u_i , which corresponds to the k -hop constrained weighted adjacent matrix $A_{SG_l}^k(t)$, the method for calculating the value of its elements is the same as (3).

Definition 4. (k -hop constrained local algebraic connectivity): Let $\lambda_i^k(t)$ denote k -hop constrained local algebraic connectivity of u_i , it is defined as the second smallest eigenvalue of $L_{SG_l}^k(t)$.

This paper uses k -hop constrained local algebraic connectivity to characterize the local connectivity of nodes. Based on this metric, we could classify a USNET's running state into three types: normal, abnormal, and outage, which are defined as follows.

Definition 5. (*Normal state*): A USNET is said to be in normal state at time t if it satisfies the following conditions:

- Any node $u_i \in U(t)$ in the network can reach all other nodes $u_j \in U(t)$, i.e., $\lambda_G(t) > 0$;
- k -hop constrained local algebraic connectivity of $\forall u_i \in U(t)$ is greater than a specific threshold ζ , i.e., $\min_{\forall u_i \in U(t)} \lambda_i^k(t) \geq \zeta$.

Definition 6. (*Abnormal state*): A USNET is said to be in abnormal state at time t if it satisfies the following conditions:

- Any node $u_i \in U(t)$ in the network can reach all other nodes $u_j \in U(t)$, i.e., $\lambda_G(t) > 0$;
- k -hop constrained local algebraic connectivity of $\forall u_i \in U(t)$ is less than a specific threshold ζ , i.e., $\min_{\forall u_i \in U(t)} \lambda_i^k(t) < \zeta$.

The nodes with a k -hop constraint local algebraic connectivity lower than the threshold are labeled as abnormal nodes.

Definition 7. (*Outage state*): A USNET is said to be in outage state at time t if there exists any surviving node $u_i \in U(t)$ cannot reach any other surviving nodes $u_j \in U(t)$, i.e., $\lambda_G(t) = 0$.

Figure 1 shows the state transition diagram of a USNET.

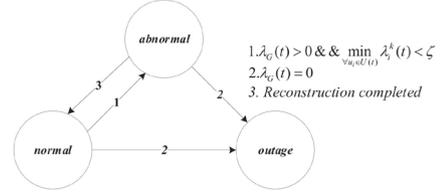


Figure 1. The state transition diagram of a USNET

Assuming that the failure events of a USNET are discrete, independent, and mutually exclusive; and the number of failure events constitutes a Poisson process. In other words, the probability of node failure events occurring k times within time $[t, t + \tau]$ is:

$$P[(N(t + \tau) - N(t)) = k] = \frac{e^{-\lambda\tau} (\lambda\tau)^k}{k!}, k = 0, 1, \dots \quad (8)$$

According to the characteristics of the Poisson process, the arrival time interval of node failure events is an independent and identically distributed exponential random variable with the mean value $1/\lambda$, which will be referred to as the failure rate of the nodes r_{fail} in the following analysis. The occurrence of a node failure event means that one of the surviving nodes fails with equal probability.

A USNET is in the normal state when it is initialized. When a node fail event happens, if the network is still connected and there is at least one node's k -hop constrained local algebraic connectivity is less than a certain threshold ζ , i.e., $\lambda_G(t) > 0 \ \&\& \ \min_{\forall u_i \in U(t)} \lambda_i^k(t) < \zeta$, then the USNET enters the abnormal state. If the network is disconnected, i.e., $\lambda_G(t) = 0$, the USNET enters the outage state. A self-reconstruction process is called in abnormal state, and the USNET will switch back to the normal state after the reconstruction is completed. If a node fails during the reconstruction process, the process will be aborted. Whether to restart the self-reconstruction process depends on the network's connectivity status. The reconstruction problem at the outage state can refer to [17]. This paper focuses on the self-reconstruction process in the abnormal state.

2.2 Problem Formulation

Figure 2 shows an example of a USNET's different states. For convenience, the value of k is set to 2 in this paper,

sigmoid function $f(d_{ij}(t)) = 1 / \left(1 + e^{\frac{20(d_{ij}(t) - 90)}{R} - 10} \right)$, $R = 200m$, $\zeta = 1$.

At time t_0 , $\lambda_G(t) = 0.83 > 0$, $\min_{\forall u_i \in U(t)} \lambda_i^2(t) = 1.51 \geq \zeta$, thus

the USNET shown in Figure 2(a) is in normal state. Node 2 fails when $t = t_1$, $\lambda_G(t) = 0.58 > 0$, but $\min_{\forall u_i \in U(t)} \lambda_i^2(t) = 0.97 < \zeta$, the USNET shown in Figure 2(b) switches to the abnormal state. In this case, Node 4 is an abnormal node, whose 2-hop constrained local algebraic connectivity is 0.97, which is less than the threshold. Node 4 is also a cut-vertex of the network. Then node 4 fails when $t = t_2$, which results in the USNET to be partitioned into two disjoint subnets ($\lambda_G(t) = 0$), then it enters the outage state, as shown in Figure 2(c).

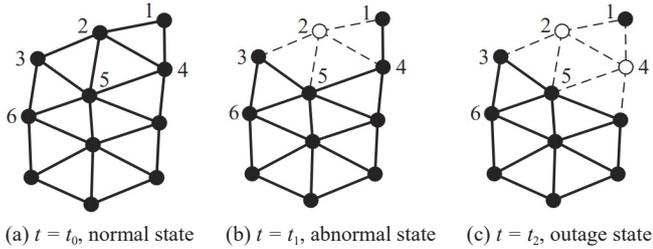


Figure 2. An example of a USNET's different states

In order to restore the USNET's state from abnormal to the normal state, it is necessary to use the node's maneuverability to move the surrounding relevant nodes (such as node 1 or 3) to the appropriate positions (such as the position of node 2). This could greatly reduce the probability of the USNET entering the outage state.

The kinematic model of the node u_i , $i \in \{1, 2, \dots, n\}$ is as follows:

$$\begin{cases} \dot{q}_i(t) = v_i(t) \\ 0 \leq \|v_i(t)\| \leq V_{\max} \end{cases}, \quad (9)$$

where $v_i(t)$ is the moving speed of the UAVs at time t , and $V_{\max} \in \mathbb{R}$ denote the maximum flight speed of a UAV.

The three main metrics to measure the goodness of a topology reconstruction method include: topology reconstruction time $T_{reconstruct}$, algebraic connectivity $\lambda_G(t)$, and total travel distance of nodes TD . $T_{reconstruct}$ refers to the time from the occurrence of a node failure event to the completion of topology reconstruction. The algebraic connectivity $\lambda_G(t)$ is used to measure the robustness of the network before and after reconstruction; and the total travel distance is used to reflect the energy efficiency of the node movement process. Let $Q'(t) = \{q'_i(t) \mid u_i \in U(t)\}$ denote the set of target locations of all nodes $u_i \in U(t)$ at time t , where $q'_i(t) \in \mathbb{R}^3$ denote the target location of node u_i , then $TD = \sum_{u_i \in U(t)} \|q'_i(t) - q_i(t)\|$. The longer the travel distance, the greater the energy consumption and thus the lower the energy efficiency.

Then, the reconstruction of a USNET's topology could be modeled as an optimization problem that minimizes all nodes' total travel distance while restoring the network's connectivity, as shown in (10):

$$Q'(t) = \arg \min_{Q'(t)} \sum_{u_i \in U(t)} \|q'_i(t) - q_i(t)\| \quad (10)$$

$$\begin{cases} \min_{\forall u_i \in U(t)} \lambda_i^k(t) \geq \zeta & (11) \\ \forall u_i \in U(t), u_j \in U(t), d_{ij}(t) \geq R_{\min} & (12) \\ \lambda_G(t) > 0 \text{ while reconstruct the topology} & (13) \end{cases}$$

Exp. (11) describes **minimum local algebraic connectivity constraint**, which requires that the k -hop constrained local algebraic connectivity of all nodes in the target topology is greater than ζ . Exp. (12) describes **minimum inter-node distance constraint**, which requires that the distance between any two neighboring nodes in the target topology is no less than $R_{\min} \in (0, R)$, so as to avoid potential collision or interference. Exp. (13) describes **keep-connected constraint**, which requires the USNET keep connected during the reconstruction process.

3 k -hop Constrained Reachability based Proactive Connectivity Maintaining Mechanism

This section details the k -hop constrained reachability based proactive connectivity maintaining (PCM) mechanism.

3.1 Basic Idea

When a node failure event happens, PCM mechanism autonomously adjusts the network topology to maintain the connectivity of the network utilizing each node's (i.e., a UAV's) decision-making ability and maneuverability. To be specific, the following three steps will be conducted:

- Distributed abnormal state identification: each node periodically perceives information of k -hop neighboring nodes and identify abnormalities in time.
- Centralized topology reconstruction: the identified network abnormalities are reported to the master node, which calculates the network topology reconstruction plan that satisfies the connectivity requirements.
- Parallel topology transformation: transform the abnormal topology into the new topology according to the network reconstruction plan.

Next, we will detail each step in the following sections.

3.2 Distributed Abnormal-state Identification

Each node detects network abnormalities through calculating whether its k -hop constrained local algebraic connectivity is less than a certain threshold. Specifically, a node $u_i \in U(t)$ periodically detects whether the link between itself and its neighboring node $u_j \in N_1^i(t)$ is broken; if the link is disconnected, u_i will start the following process:

- Exchange information with its 1-hop neighboring nodes to obtain their respectively IDs, locations, neighboring nodes of its 2-hop neighboring nodes;
- Calculate its 2-hop constrained local algebraic connectivity $\lambda_i^2(t)$;
- If $\lambda_i^2(t) < \zeta$, u_i judges that the network is in abnormal state, and reports its decision to the master node u_{master} for network reconstruction.

This process is formally illustrated in Algorithm 1, named distributed abnormal identification (DAI) algorithm.

Algorithm 1. Distributed abnormal identification (DAI) algorithm

Input: ζ
Output: Whether the network is abnormal or not

```

1:  if  $\exists u_i \in N_i^1(t)$ ,  $e_{ij}$  is disconnected then
2:    communicate with  $u_j \in N_i^1(t)$  twice to get the information of  $N_i^2(t)$ 
3:    for  $u_j \in N_i^1(t)$  do
4:      for  $u_k \in N_i^1(t)$  do
5:        if  $j=k$  then
6:           $[A_{SG_i}^2(t)]_{jk} \leftarrow 0$ 
7:        else if  $j \neq l$  and  $k \neq l$  then
8:          if  $\exists e_{jk} \in E(t)$  then
9:             $[\gamma'_{jk}]_2 \leftarrow 1$ 
10:         else if  $N_j^1(t) \cup N_k^1(t) \setminus \{u_i\} \neq \emptyset$  then
11:            $[\gamma'_{jk}]_2 \leftarrow 1/2$ 
12:         else
13:            $[\gamma'_{jk}]_2 \leftarrow 0$ 
14:         end if
15:          $[A_{SG_i}^2(t)]_{jk} \leftarrow [\gamma'_{jk}]_2 \cdot f(d_{jk}(t))$ 
16:       else
17:          $[A_{SG_i}^2(t)]_{jk} \leftarrow f(d_{jk}(t))$ 
18:       end if
19:        $[L_{SG_i}^2(t)]_{ij} \leftarrow [L_{SG_i}^2(t)]_{ij} + [A_{SG_i}^2(t)]_{jk}$ 
20:     end for
21:   end for
22:    $L_{SG_i}^2(t) \leftarrow [L_{SG_i}^2(t) - A_{SG_i}^2(t)]$ 
23:    $\lambda_i^2(t) \leftarrow$  second smallest eigenvalue of  $L_{SG_i}^2(t)$ 
24:   if  $\lambda_i^2(t) < \zeta$  then
25:     inform  $u_{master}$  that current system is in an abnormal state
26:   end if
27: end if

```

3.3 Centralized Topology Reconstruction

After receiving one or more abnormal reports from slave nodes, the master node u_{master} will immediately poll all nodes to get the global topology information, and then calculates the network topology reconstruction plan.

3.3.1 Topology Reconstruction Problem

The key to solving the topology reconstruction problem is to calculate the target position of all nodes to make the 2-hop constrained local algebraic connectivity of the abnormal nodes reach the threshold. We solve the problem through improving the global algebraic connectivity of the network, since increasing the global algebraic connectivity can improve the lower bound of the 2-hop constrained local algebraic connectivity. The maximization of algebraic connectivity is an NP hard problem [18]. There are two methods to improve the algebraic connectivity of the network: increasing the edges in the network, and shortening the distance between nodes.

PFM mechanism, which is designed by Zavlanos et al., has good performance in maintaining connectivity [14]. However, in the USNET environment, the shrinking process of the network will be restricted by the Minimum inter-node distance constraint. This may make PFM be trapped in the local optimum when the abnormal nodes (such as cut-vertex) still exist in the network. If there are cut-vertices in the network, the failure of the cut-vertex will cause the network to be partitioned into multiple disjoint subnets, which makes the network unable to work normally.

This may cause it to trap in local optimum when the abnormal nodes (such as cut-vertex) still exist in the network. If there are cut-vertices in the network, the failure of the cut-vertex will cause the network to be partitioned into multiple disjoint subnets, which makes the network unable to work normally. Figure 3 illustrates two cases of this problem. In Figure 3, after the initial topology is transformed by the PFM mechanism, there still exist cut-vertices in the target topology or the target topology cannot satisfy the specific connectivity requirements. The initial topology in Figure 3(a) is a regular hexagon composed of six nodes, when the distance between nodes is reduced to the minimum inter node distance, the nodes cannot get closer to each other. However, the topology is still a regular hexagon, and the connectivity of the target topology may not satisfy the specific connectivity requirements. In the initial topology in Figure 3(b), all nodes are located on a straight line. When the distance between nodes is reduced to the minimum inter-node distance, all nodes with degree 2 (such as node 2 and node 3) are cut-vertex.

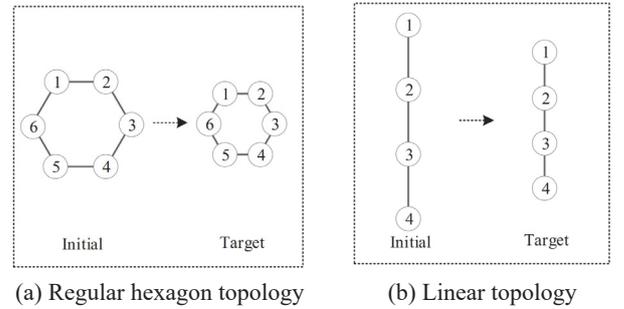


Figure 3. Problems of PFM mechanism

To solve this problem, we propose an algorithm to improve the algebraic connectivity of the network, which combines the methods of increasing the edges in the network and shortening the distance between nodes.

3.3.2 Topology Reconstruction Mechanism

In our topology reconstruction mechanism, our solution for the aforementioned problems contains two parts:

On the one hand, several virtual edges are introduced into the network to solve the problem illustrated in Figure 3(a). A virtual edge is added between two adjacent nodes beyond each other's communication range, so that they can attract each other until their distance is less than the communication range, i.e., R , so as to increase their algebraic connectivity. As shown in Figure 4(a).

On the other hand, as for the problem illustrated in Figure 3(b), a 3-dimensional plane is constructed which contains

the straight line and has the smallest angle with xOy plane. If the number of nodes is greater than 3, we construct an axis with the same direction as the normal vector of the plane at the 1-hop neighboring node of the node with degree 1, then rotate the node with degree 1 around the axis by 120 degrees. As shown in Figure 4(b). If the number of nodes is 3, only one of the nodes with degree 1 is selected to perform the aforementioned rotation operation.

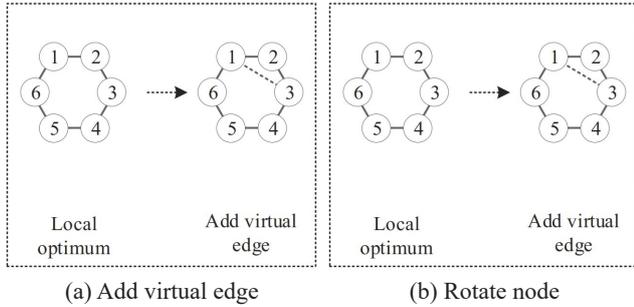


Figure 4. Solutions for the problems of PFM mechanism

After adding these two emendations, the refined topology reconstruction process is conducted as follows.

Firstly, PFM mechanism is applied for improving the algebraic connectivity of the network. When $u_i \in U(t)$ moves according to the control law shown in (14), the network will shrink toward the center, and the algebraic connectivity will be improved.

$$v_c^i(t) = \frac{a \cdot \text{tr} \left[M^{-1}(t) \frac{d}{dq_i(t)} M(t) \right]}{(\det(M(t)))^a}, \quad (14)$$

where a is a positive constant. $M(t) = P^T L_G(t) P$, $P = [p_1 p_2 \dots p_n]$, is an $n \times (n-1)$ matrix. $p_i^T p_j = 0$ for all $i, j = 1, \dots, n$, and $p_i^T \mathbf{1} = 0$ for all $i = 1, \dots, n$, where $\mathbf{1}$ is an n -dimensional vector with all entries equal to 1.

When the distance between nodes is less than the minimum inter-node distance, the repulsion force will be generated between nodes, which is described by the potential function $\phi(t)$ defined in (15).

$$\phi(t) = \frac{1}{2} \sum_{u_i \in U(t)} \sum_{u_j \in U(t), i \neq j} \phi_{ij}^i(t). \quad (15)$$

The potential function $\phi_{ij}(t)$ describes the repulsive force between nodes u_i and u_j :

$$\phi_{ij}(t) = \begin{cases} (R_{\min} - d_{ij})^2, & \text{if } d_{ij} \in (0, R_{\min}) \\ 0, & \text{otherwise} \end{cases}. \quad (16)$$

The control law of the repulsive force on $u_i \in U(t)$ is:

$$v_a^i(t) = -\frac{1}{2} \sum_{u_j \in U(t), i \neq j} \frac{d}{dq_i(t)} \phi_{ij}(t). \quad (17)$$

The above method may be trapped in a local optimum when the algebraic connectivity doesn't reach the threshold. The proposed PCM mechanism overcomes this by adding a virtual edge between nodes beyond the communication range to escape local optimum. Let the candidate virtual edge set be $VE = \{ve_{ij} \mid u_i \in U(t), u_j \in U(t)\}$, when u_i and u_j satisfy the following three conditions, ve_{ij} is added to the candidate virtual edge set VE :

- (a) u_i and u_j are not adjacent, i.e., $u_j \in U(t) \setminus N_i^+(t)$;
- (b) No other nodes are included on the line segment connecting u_i and u_j , i.e., $\forall u_k \in U(t) \setminus \{u_i, u_j\}, d_{ij}(t) \neq d_{ik}(t) + d_{kj}(t)$;
- (c) The line segment connecting u_i and u_j does not intersect with any other edge in the graph $G(t)$, i.e., for $\forall e_{ab} \in E(t)$, the following equation is satisfied.

$$\begin{cases} [(q_a(t) - q_i(t)) \times (q_b(t) - q_j(t))] \\ \times [(q_a(t) - q_j(t)) \times (q_b(t) - q_i(t))] = 0 \\ [(q_a(t) - q_i(t)) \times (q_b(t) - q_j(t))] \\ \cdot [(q_b(t) - q_i(t)) \times (q_j(t) - q_i(t))] < 0 \\ [(q_i(t) - q_a(t)) \times (q_b(t) - q_a(t))] \\ \cdot [(q_j(t) - q_a(t)) \times (q_b(t) - q_a(t))] < 0. \end{cases} \quad (18)$$

If the candidate virtual edge set $VE \neq \emptyset$, let $deg(u_i)$ denote u_i 's degree, $VE' \subseteq VE$ denote the candidate virtual edge set of nodes with degree 1, i.e., $VE' = \{ve_{ij} \mid ve_{ij} \in VE \text{ and } (deg(u_i) = 0 \text{ or } deg(u_j) = 0)\}$. When selecting the added virtual edges, the edges in the set VE' have the higher priority. If $VE' \neq \emptyset$, the shortest edge in VE' is chosen as the virtual edge; otherwise, the shortest edge in VE is selected as the virtual edge. Let ve_{ij} denote the selected virtual edge to be added, $ve_{ij} = (VE' \neq \emptyset ? \arg \min_{ve_{ab} \in VE'} d_{ab} : \arg \min_{ve_{ab} \in VE} d_{ab})$. Then, set the distance d_{ij} corresponding to ve_{ij} be the latest transmission range, and update the elements of the weighted adjacency matrix. In addition to the existing links, we set the value of the adjacency matrix element corresponding to the virtual edge ve_{ij} to $f(d_{ij}(t))$; then, update the value of the weighted Laplacian matrix, and recalculate the velocity of each node according to (14). Since the distance of the virtual edge ve_{ij} is relatively long, the attractive force between u_i and u_j is small, an amplification coefficient ρ is introduced to increase the attractive force generated by the virtual edge ve_{ij} and speeds up the convergence. $\rho > 1$ is a constant, and the attractive forces $v_c^i(t)$ and $v_c^j(t)$ are multiplied by ρ .

If the candidate virtual edge set $VE \neq \emptyset$, then we judge whether all nodes in the network locate on a straight line. If all the nodes in the network are located on a straight line, i.e., for $\forall u_i, u_j, u_k \in U(t), i \neq j \neq k, [q_k(t) - q_i(t)] \times [q_j(t) - q_i(t)] = \mathbf{0}$, we construct a plane which contains the straight line and has the smallest angle with the xOy plane. Let $u_i \in U(t), u_j \in U(t)$ denote the nodes with degree 1, (x_i, y_i, z_i) and (x_j, y_j, z_j) denote the 3-dimensional coordinate (n_x, n_y, n_z) s of nodes u_i and u_j respectively, then the constructed plane equation is as follows:

$$(y_j - y_i)x + [(x_i - x_j) + \kappa(z_j - z_i)]y + \kappa(y_i - y_j)z + x_j y_i - x_i y_j + \kappa(y_j z_i - y_i z_j) = 0, \quad (19)$$

where the value of κ is shown:

$$\kappa = \frac{(y_j - y_i)^2 + (x_i - x_j)^2}{(x_j - x_i)(z_j - z_i)}. \quad (20)$$

Let $\overline{(n_x, n_y, n_z)}$ denote the unit normal vector of the plane represented by (19), let $u_a \in U(t)$ denote the neighboring nodes of $u_i \in U(t)$ with degree 1, we construct an axis with the same direction as the normal vector at the coordinates of the node u_a , and rotate u_i 's coordinate by 120 degrees around the axis. Let $q'_i(t)$ denote u_i 's new coordinate after rotation, it will be:

$$q'_i(t) = [q_i(t) - q_a(t)]N + q_a(t), \quad (21)$$

where the matrix $N \in \mathbb{R}^{3 \times 3}$ is defined as:

$$N = \begin{bmatrix} \frac{3}{2}n_x^2 - \frac{1}{2} & \frac{3}{2}n_x n_y + \frac{\sqrt{3}}{2}n_z & \frac{3}{2}n_x n_z - \frac{\sqrt{3}}{2}n_y \\ \frac{3}{2}n_x n_y - \frac{\sqrt{3}}{2}n_z & \frac{3}{2}n_y^2 - \frac{1}{2} & \frac{3}{2}n_y n_z + \frac{\sqrt{3}}{2}n_x \\ \frac{3}{2}n_x n_z + \frac{\sqrt{3}}{2}n_y & \frac{3}{2}n_y n_z - \frac{\sqrt{3}}{2}n_x & \frac{3}{2}n_z^2 - \frac{1}{2} \end{bmatrix}. \quad (22)$$

When the number of nodes is 3, let $u_i, u_j \in U(t)$ denote the nodes with degree 1, select a node with the smaller ID from u_i and u_j to perform the above rotation operation.

When the iterative process is trapped in the local optimum, the algebraic connectivity can be improved by adding virtual edges or rotating nodes until 2-hop constrained local algebraic connectivity of $\forall u_i \in U(t)$ is greater than a specific threshold, i.e., $\min_{\forall u_i \in U(t)} \lambda_i^2(t) \geq \zeta$. When the algorithm is trapped in local optimum, if the candidate virtual edge set $VE \neq \emptyset$ and all nodes are not located on the same straight line, the algorithm fails.

3.3.3 Virtual Edge-based Topology Reconstruction

Algorithm

The virtual edge-based topology reconfiguration (VET) algorithm is illustrated in Algorithm 2.

Algorithm 2. Virtual edge-based topology reconstruction (VET) algorithm

Input: $U(t), q_i(t), \zeta, L_G(t), R, R_{\min}$

Output: $Q'(t)$

```

1:  detM ← -1; for  $\forall u_i \in U(t), q'_i(t) \leftarrow q_i(t)$ 
2:  while true do
3:    flag ← false
4:    for  $u_i \in U(t)$  do
5:      if  $\lambda_i^2(t) < \zeta$  then
```

```

6:      flag ← false, break
7:    end if
8:  end for
9:  if !flag then
10:    reconstruction finished, break
11:  end if
12:   $M(t) \leftarrow P^T L_G(T) P, lastDetM \leftarrow detM$ 
13:   $detM \leftarrow$  determinant of  $M(t)$ 
14:  if  $detM \leq lastDetM$  then
15:     $VE \leftarrow$  virtual edges satisfy the requirements
16:    if  $VE \neq \emptyset$  then
17:       $ve_{ij} \leftarrow VE' \neq \emptyset? \arg \min_{ve_{ab} \in VE} d_{ab}$ 
18:      :  $\arg \min_{ve_{ab} \in VE} d_{ab}$ 
19:      update  $L_G(t)$  and  $M(t)$  based on  $d_{ij}$ 
20:      else if all nodes are collinear then
21:        rotate node with degree 1 according to (21)
22:      continue
23:    else
24:      algorithm failed, break
25:    end if
26:  for  $u_i \in U(t)$  do
27:     $v'_c(t) \leftarrow \frac{a \cdot tr \left[ M^{-1}(t) \frac{d}{dq'_i(t)} M(t) \right]}{(det(M(t)))^a}$ 
28:    if  $u_i$  is one of the vertices of the chosen virtual edge then
29:       $v'_c(t) \leftarrow \rho * v'_c(t)$ 
30:    end if
31:     $v'_a(t) \leftarrow -\frac{1}{2} \sum_{u_j \in U(t), i \neq j} \frac{d}{dq'_i(t)} \phi_{ij}(t)$ 
32:     $q'_i(t) \leftarrow q'_i(t) + v'_c(t) + v'_a(t)$ 
33:  end for
34: end while
```

3.4 Parallel Topology Transformation

After obtaining the target topology, the master node delivers the reconstruction solution to the slave nodes, and all nodes move to their respective destination positions accordingly. To optimize this transformation process, two aspects need consideration:

- Optimization of the total travel distance. The shape of the target topology is determined in the process of topology reconstruction. The location and orientation of the target topology are optimized to minimize the total travel distance of nodes during the transformation from the initial topology to the target topology, thereby reducing energy consumption in the process.
- Connectivity maintenance. Each node's moving speed is calculated to ensure the connectivity of the network during the reconstruction process.

3.4.1 Optimization of the Travel Distance

To minimize energy consumption during topology reconstruction, optimizing each node's moving trajectory is essential. Figure 5 demonstrates an example of travel distance optimization. Figure 5(a) depicts the initial and target topology shapes. The optimization approach is as follows: Firstly, align the target topology with the initial topology's geometric center, and use it as the coordinate origin, as shown in Figure 5(b). Secondly, the target topology is rotated around the geometric center with an angle of $\alpha \in [0, 2\pi]$, and the problem is reduced to the calculation of the optimal α to minimize the travel distance of topology reconstruction. Since the UAV node will be limited by the flight altitude when rotating, we restrict the rotation could only be conducted around the z -axis, so that the altitude of each node remains unchanged, as shown in Figure 5(c).

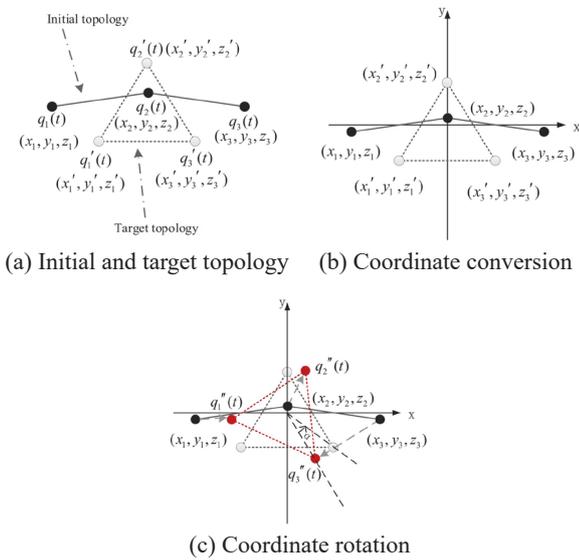


Figure 5. Coordinate conversion and rotation

Let (x_i, y_i, z_i) and (x'_i, y'_i, z'_i) denote u_i 's initial and the target coordinates respectively, $u_i \in U(t)$. Then, the coordinates of the initial topology's and the target topology's geometric centers will be $o_{ic} = (\frac{\sum_{u_i \in U(t)} x_i}{\|U(T)\|}, \frac{\sum_{u_i \in U(t)} y_i}{\|U(T)\|}, \frac{\sum_{u_i \in U(t)} z_i}{\|U(T)\|})$

and $o_{ic} = (\frac{\sum_{u_i \in U(t)} x'_i}{\|U(T)\|}, \frac{\sum_{u_i \in U(t)} y'_i}{\|U(T)\|}, \frac{\sum_{u_i \in U(t)} z'_i}{\|U(T)\|})$ respectively. We

treat the o_{ic} and o_{ic} to be coordinate origin, let $(\hat{x}_i, \hat{y}_i, \hat{z}_i)$ and $(\hat{x}'_i, \hat{y}'_i, \hat{z}'_i)$ denote the coordinates of the initial position and target position of the node $u_i \in U(t)$ in the new coordinate system respectively, then we can get $(\hat{x}_i, \hat{y}_i, \hat{z}_i) = q_i(t) - o_{ic}$, $(\hat{x}'_i, \hat{y}'_i, \hat{z}'_i) = q'_i(t) - o_{ic}$.

The coordinates $q''_i(t)$ obtained by rotating the target position of the node $u_i \in U(t)$ around the z -axis are shown as follows.

$$q''_i(t) = (q'_i(t) - o_{ic}) \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (23)$$

The value of the total travel distance TD is:

$$\begin{aligned} TD &= \sum_{u_i \in U(t)} \|q''_i(t) - (q_i(t) - o_{ic})\| \\ &= \sum_{u_i \in U(t)} \left[(\hat{x}'_i \cos \alpha - \hat{y}'_i \sin \alpha - \hat{x}_i)^2 \right. \\ &\quad \left. + (\hat{x}'_i \sin \alpha + \hat{y}'_i \cos \alpha - \hat{y}_i)^2 + (\hat{z}'_i - \hat{z}_i)^2 \right]^{\frac{1}{2}} \\ &= \sum_{u_i \in U(t)} \left[\hat{x}'_i{}^2 + \hat{y}'_i{}^2 + \hat{x}_i{}^2 + \hat{y}_i{}^2 + (\hat{z}'_i - \hat{z}_i)^2 \right] \\ &\quad - 2 \left[(\hat{x}_i \hat{x}'_i + \hat{y}_i \hat{y}'_i) \cos \alpha + (\hat{x}_i \hat{y}'_i - \hat{x}'_i \hat{y}_i) \sin \alpha \right]^{\frac{1}{2}}. \quad (24) \end{aligned}$$

The calculation of the minimum value of the angle α in (24) is to solve a single-variable bounded nonlinear function minimization problem. The angle α can be obtained by the method of combining golden section search and successive parabolic interpolation [19]. Let the coordinates of the optimized target position of the node u_i be $q^d_i(t)$, then $q^d_i(t) = q''_i(t) + o_{ic} = ((x'_i - \frac{\sum_{u_i \in U(t)} x'_i}{\|U(t)\|}) \cos \alpha - (y'_i - \frac{\sum_{u_i \in U(t)} y'_i}{\|U(t)\|}) \sin \alpha + \frac{\sum_{u_i \in U(t)} x_i}{\|U(t)\|}, (x'_i - \frac{\sum_{u_i \in U(t)} x'_i}{\|U(t)\|}) \sin \alpha + (y'_i - \frac{\sum_{u_i \in U(t)} y'_i}{\|U(t)\|}) \cos \alpha + \frac{\sum_{u_i \in U(t)} y_i}{\|U(t)\|}, z'_i - \frac{\sum_{u_i \in U(t)} z'_i}{\|U(t)\|} + \frac{\sum_{u_i \in U(t)} z_i}{\|U(t)\|})$. Now the shape, position, and orientation of the target topology have been determined. Next, we will discuss how to transform from the initial topology to the target topology.

3.4.2 Connectivity Maintenance

In the process of topology reconstruction, the master node needs to determine the flight speed of each slave node, so that the node will not exceed the maximum flight speed, and satisfy the keep connected constraint in the transformation process.

After determining the initial and target positions for a node, its longest moving distance (d_{\max}) during the topology reconstruction process can be calculated by:

$$d_{\max} = \max_{u_i \in U(t)} \|q^d_i(t) - q_i(t)\|. \quad (25)$$

The fastest topology reconstruction time $T_{reconstruct} = d_{\max} / V_{\max}$, where V_{\max} is the maximum flight speed of the UAV.

Theorem 1. *If there is at least one identical spanning tree between the initial and the target network topologies and all nodes arrive at their target locations with the same movement time t_{travel} , the links in the same spanning tree will not disconnect during the movement, i.e., the keep connected constraint is satisfied.*

Proof: To facilitate the calculation, we could select any one of the same edges in the identical spanning tree, and set the initial coordinates of the nodes u_1 and u_2 , i.e., two ends of the selected link, be $(0, 0, 0)$ and $(a, 0, 0)$, respectively, $a \in (0, R)$. In the target network topology, the coordinates of u_1 and u_2 are (x_1, y_1, z_1) , (x_2, y_2, z_2) , as shown in Figure 6.

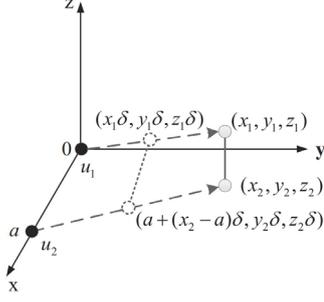


Figure 6. Link connectivity during topology transformation

If u_1 and u_2 take the same time t_{travel} to reach their respective target position, their velocities will be $(\frac{x_1}{t_{travel}}, \frac{y_1}{t_{travel}}, \frac{z_1}{t_{travel}})$ and $(\frac{x_2-a}{t_{travel}}, \frac{y_2}{t_{travel}}, \frac{z_2}{t_{travel}})$, respectively. During the moving process, after u_1 and u_2 move for a time period t , $t \in [0, t_{travel}]$, their respective positions are $(\frac{x_1 t}{t_{travel}}, \frac{y_1 t}{t_{travel}}, \frac{z_1 t}{t_{travel}})$ and $(a + \frac{(x_2-a)t}{t_{travel}}, \frac{y_2 t}{t_{travel}}, \frac{z_2 t}{t_{travel}})$. The distance d_{12} between u_1 and u_2 is:

$$\begin{aligned} d_{12} &= \sqrt{(a + (x_2 - a)\delta - x_1\delta)^2 + (y_2\delta - y_1\delta)^2 + (z_2\delta - z_1\delta)^2} \\ &= \left[\delta^2((x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2) \right. \\ &\quad \left. + (1 - \delta)^2 a^2 + 2\delta(1 - \delta)a(x_2 - x_1) \right]^{\frac{1}{2}}, \end{aligned} \quad (26)$$

where $\delta = t / t_{travel}$. It's easy to know that $(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 \leq R^2$ and $a(x_2 - x_1) \leq R^2$. Substituting these inequalities into (26), we have:

$$d_{12} \leq \sqrt{\delta^2 R^2 + (1 - \delta)^2 R^2 + 2\delta(1 - \delta)R^2} = R. \quad (27)$$

Therefore, if all nodes move to their target positions with the same movement time, the distance between the nodes of any link in the identical spanning tree will not exceed the transmission range R during the movement; in other words, the link will not be disconnected. This completes the proof.

In order to satisfy keep connected constraint, VET algorithm needs to be modified. The basic idea of the modification is to calculate the minimum spanning tree MST of the initial topology before planning the target topology. Let $B_i^j(t)$ denote the set of node u_i and all its descendants in the MST rooted at node u_i . In generating the target topology, if the repositioning of u_i will lead to the disconnection of and edge $e_{ij} \in MST$ connecting u_i and u_j , all the nodes in $B_i^j(t)$ are moved in cascade, and their displacement is consistent with the displacement of u_i , so that the edge e_{ij} in the MST will not be broken, and at least one identical spanning tree exists in the initial topology and the target topology. According to Theorem 1, as long as the node moves from the initial

position to the target position with the same movement time, the keep connected constraint can be satisfied. The modified virtual edge-based topology reconstruction (MVET) algorithm is shown in Algorithm 3.

Algorithm 3. Modified virtual edge-based topology reconstruction (MVET) algorithm

Input: $G(t)$, $U(t)$, $q_i(t)$, ζ , $L_G(t)$, R , R_{min}

Output: $Q'(t)$

```

1:   $detM \leftarrow -1$ ; for  $\forall u_i \in U(t)$ ,  $q_i'(t) \leftarrow q_i(t)$ 
2:   $MST \leftarrow$  minimum spanning tree of  $G(t)$ 
3:  while true do
4:    same as the Step 3-25 in VET algorithm
5:    for  $u_i \in U(t)$  do
6:       $v_c^i(t) \leftarrow \frac{a \cdot tr \left[ M^{-1}(t) \frac{d}{dq_i'(t)} M(t) \right]}{(det(M(t)))^a}$ 
7:       $v_a^i(t) \leftarrow -\frac{1}{2} \sum_{u_j \in U(t), i \neq j} \frac{d}{dq_i'(t)} \phi_{ij}(t)$ 
8:       $q \leftarrow v_c^i(t) + v_a^i(t)$ 
9:       $q_i'(t) \leftarrow q_i'(t) + q$ 
10:     for  $e_{ij} \in MST$  do
11:       if  $\|q_i'(t) - q_j'(t)\| > R$  then
12:          $B_i^j(t) \leftarrow \{u_j \text{ and all its descendants in } u_i\text{-rooted } MST\}$ 
13:         for  $u_k \in B_i^j(t)$  do
14:            $q_k'(t) \leftarrow q_k'(t) + q$ 
15:         end for
16:       break
17:     end if
18:   end for
19: end while
20: end while

```

4 Simulation and Results Analysis

The proposed PCM was compared with PFM [14] in algebraic connectivity, total moving distance, and fault-tolerance. All simulations are conducted on MATLAB. Table 1 summarized the simulation parameters used in the simulations.

Table 1. Simulation parameters

Parameter	Value
Transmission range (R)	200 m
Maximum flight speed V_{max}	20 m/s
Node density (D)	35 nodes/km ²
Minimum inter-node distance R_{min}	150 m
Threshold of 2-hop constrained local algebraic connectivity (ζ)	1.0
Altitude	200 m
Sigmoid function $f(d_{ij}(t))$	$1 / \left(1 + e^{\frac{20(d_{ij}(t)-90)}{R}-10} \right)$
Amplification factor (ρ)	20

Firstly, the validity of DAI algorithm and the relationship between local algebraic connectivity and global algebraic connectivity are tested. According to the uniform distribution, four test topologies are randomly generated in a square area with a side length of 1000 *m* at a specific node density *D* (case 1-4 in Figure 7). The generated topology needs to be connected and the minimum distance between nodes is not less than *R*_{min}. Node failure event starts at 40 seconds, and one node is randomly selected every 5 seconds as the failed one. When it is decided that the network enters the abnormal state, the test is terminated, and the test result is shown in Figure 7.

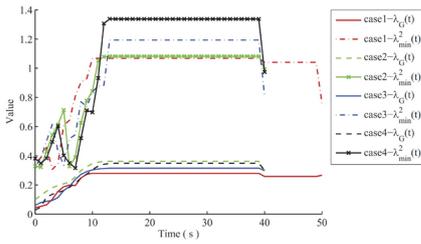


Figure 7. The validity of DAI algorithm and the relationship between $\lambda_{\min}^2(t)$ and $\lambda_G(t)$

Let $\lambda_{\min}^2(t) = \min_{u_i \in U(t)} \lambda_i^2(t)$ denote the smallest 2-hop constrained local algebraic connectivity among all nodes. As can be seen from Figure 7, the minimum local algebraic connectivity of the initial topology $\lambda_{\min}^2(t) < \zeta$, and DAI algorithm can detect that the current USNET is in abnormal state, triggering the reconstruction process, which takes approximately 10 seconds to complete. In the process of topological transformation, the distance between some nodes may be less than *R*_{min}, so the minimum local algebraic connectivity $\lambda_{\min}^2(t)$ and global algebraic connectivity $\lambda_G(t)$ will fluctuate, but the overall trend is increasing. When the time $t \geq 40s$, the minimum local algebraic connectivity $\lambda_{\min}^2(t)$ and the global algebraic connectivity $\lambda_G(t)$ decrease with the occurrence of node failure events. When 1-3 node failure events occur, DAI algorithm identifies that the USNET enters abnormal state. In general, the minimum local algebraic connectivity $\lambda_{\min}^2(t)$ and the global algebraic connectivity $\lambda_G(t)$ have the same trend. DAI algorithm can quickly identify network abnormalities, which verifies its validity.

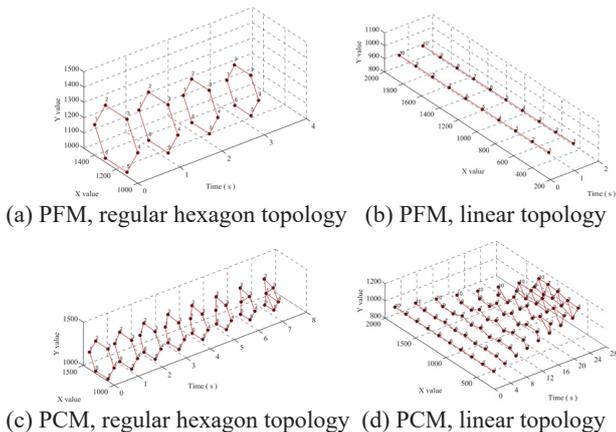


Figure 8. Comparison of the PFM and PCM mechanisms in terms of topology reconstruction

Next, we test whether PCM mechanism could solve the problems faces PFM mechanism as shown in Figure 3. Figure 8(a) and Figure 8(c) show the process of reconstructing regular hexagon topology by PFM and PCM mechanism, respectively. The topology structure reconstructed by PFM mechanism is still a regular hexagon and does not reach the connectivity threshold; in contrast, PCM mechanism completes the reconstruction in about 7 seconds. Figure 8(b) and Figure 8(d) show the process of reconstructing linear topology. The results show that PFM mechanism is trapped in the local optimum in less than 1 second, and the network topology is still a straight line. On the contrary, PCM mechanism stops at about 28 seconds, and there is no cut-vertex in the transformed topology.

Figure 9 illustrates the changes of $\lambda_{\min}^2(t)$ and $\lambda_G(t)$ in the topology reconstruction process using PFM and PCM mechanisms. It shows that $\lambda_{\min}^2(t)$ and $\lambda_G(t)$ are increasing in general. PFM mechanism stops iteration when $\lambda_{\min}^2(t)$ does not reach the threshold ζ , and fails to restore the network to normal state. PCM mechanism solves the local optimum problem of the PFM mechanism, it stops when $\lambda_{\min}^2(t)$ reaches the threshold ζ . Figure 9 also shows that PCM is better than PFM mechanism in global algebraic connectivity $\lambda_G(t)$.

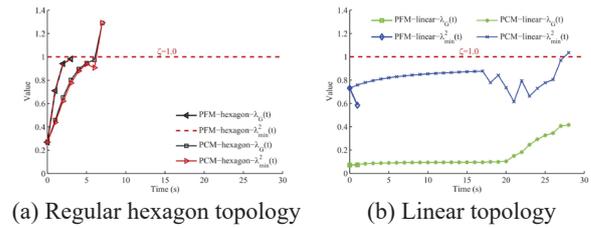
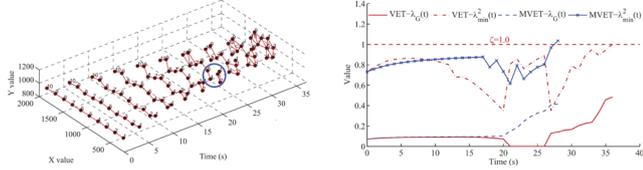


Figure 9. Comparison of the PFM and PCM mechanisms in terms of $\lambda_{\min}^2(t)$ and $\lambda_G(t)$

The third experiment tests the validity of MVET algorithm. For the convenience of presentation, a linear topology with 10 nodes is utilized, so there is only one spanning tree in the initial topology. Both VET and MVET algorithms are employed to generate the target topology. The changes in the spanning tree during the topology transformation are plotted to verify if the MVET algorithm can maintain the spanning tree without disconnection. Figure 10(a) shows the process of reconstructing linear topology using VET algorithm. Compared with the initial topology, the edges (1, 2) and (3, 4) in the spanning tree are disconnected. Therefore, the condition in Theorem 1 that there is at least one identical spanning tree between the initial network topology and the target network topology is not satisfied. It can be seen from the topology transformation process that the network is partitioned into two subnets during the period of 21 *s* to 26 *s* (as shown in the blue circle). Thus, the correctness of Theorem 1 is verified. Figure 10(b) illustrates the changes of $\lambda_{\min}^2(t)$ and $\lambda_G(t)$ in the topology reconstruction process using VET algorithm and MVET algorithm. It can be seen that VET algorithm can also generate the target topology satisfying the specific connectivity requirements, but it cannot guarantee to satisfy the keep connected constraint during the topology transformation process. Figure 10(b) tells us that $\lambda_G(t) = 0$ during the period of 21 *s* to 26 *s*; in other words, the

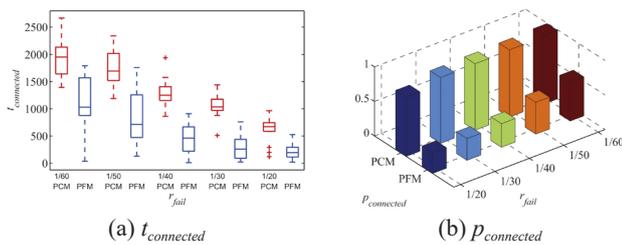
network is disconnected during this period. MVET algorithm can ensure that $\lambda_G(t) > 0$ during the topology transformation process, and the network remains connected and satisfies the keep connected constraint. Therefore, the validity of MVET algorithm is verified.



(a) Process of reconstructing linear topology using VET algorithm (b) Changes of $\lambda_{\min}^2(t)$ and $\lambda_G(t)$ using VET and MVET algorithm

Figure 10. Comparison of the reconstruction process using VET and MVET algorithm

The fourth experiment compares the fault-tolerant ability of PFM and PCM mechanisms. We randomly generate multiple test topologies in a square area with a specific node density, and the number of failure events constitutes a Poisson process (as described in section 2.1, r_{fail} denotes the mean value of the arrival time interval of node failure events). Let $t_{connected}$ denote the time from the network start-up to network outage, i.e., the time that the network remains connected. Let $p_{connected}$ denote the probability that the network remains connected. Figure 11 compares the values of $t_{connected}$ and $p_{connected}$ for PFM and PCM mechanisms under varying failure rate $r_{fail} \in \{1/60, 1/50, 1/40, 1/30, 1/20\}$. Figure 11(a) shows that as the failure rate increases, the time that the network remains connected reduces. Compared with PFM mechanism, PCM mechanism can prolong the time that the network remains connected by roughly 1.23 times. Figure 11(b) shows that with the increase of the failure rate, the probability that the network remains connected decreases. When the failure rate $r_{fail} \geq 1/30$, the proposed PCM mechanism can make the probability that the network remains connected exceeds 95%, while the PFM mechanism can only reach 56%. When the failure rate $r_{fail} = 1/20$, the probability $p_{connected}$ of PCM mechanism is reduced to about 85%. This is because the topology transformation process needs time. The shorter the time interval between node failure events, the more likely the node failure event will occur when the network topology transformation is not completed, which will increase the probability of the network entering the outage state. In summary, the proposed PCM mechanism outperforms the PFM mechanism in fault tolerance.



(a) $t_{connected}$ (b) $p_{connected}$

Figure 11. Comparison of fault-tolerance between PCM and PFM mechanisms

Node movement is an energy-intensive operation, so the total travel distance of nodes is an important metric to measure the performance of connectivity maintaining mechanism. Figure 12 shows the average total travel distance of reconstructing network with PFM and PCM mechanisms with different failure rate r_{fail} . It can be seen from Figure 12 that the average total travel distance of PCM mechanism is relatively stable, because PCM mechanism is less affected by node failure events and has good fault-tolerant ability. The average total travel distance of PFM mechanism decreases with the increase of the failure rate. This is because the network is easy to enter the outage state with the increase of failure rate, which leads to the decrease of total travel distance of nodes. When $r_{fail} \leq 1/40$, the total travel distance of PCM mechanism is significantly smaller than that of PFM mechanism. Compared with PFM mechanism, the total travel distance of PCM mechanism is reduced by 28% on average, and PCM mechanism is less affected by the failure rate.

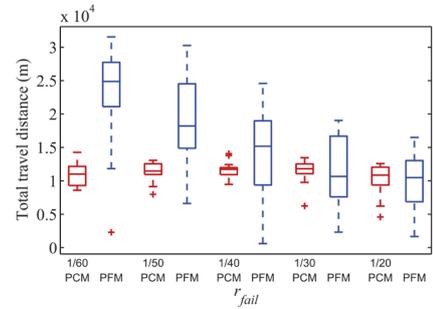


Figure 12. Comparison of total travel distance between PCM and PFM mechanisms

5 Conclusion

Harsh deploy environment or malicious damage make UAV network vulnerable to damage or failure. This paper proposes a k -hop constrained reachability based proactive mechanism to restore the network in case of node failure. This paper first formulates the connectivity maintaining problem and node failure model of the USNET. To the author’s best knowledge, the concept of local algebraic connectivity is proposed and utilized for the first time to profile the local connectivity status of a USNET. Secondly, a distributed network abnormality identification algorithm is proposed, which identifies network abnormalities according to the local topology information of neighboring nodes. Thirdly, the concept of virtual edge is put forward to solve the local optimum problem of the existing potential field method in improving algebraic connectivity; and a novel virtual edge-based topology reconstruction algorithm is designed to generate the target topology. Fourthly, a quantitative method is introduced to optimize the travel distance of the topology transformation process; and a spanning tree-based method is proposed to maintain the connectivity during the topology transformation process. A series of simulations are conducted to evaluate the proposed algorithms. Simulation results show that the proposed mechanism solves the local optimum problem in the existing potential field method, and outperforms the existing mechanisms in terms of fault

tolerance, connectivity, and total travel distance. In the future work, our focus will shift towards the real-world validation of the proposed mechanism, thereby ensuring its practical effectiveness. Furthermore, we intend to delve into the application of multi-agent reinforcement learning techniques, aiming to amplify the algorithm's adaptability and performance.

Acknowledgement

This work was supported by Research Foundation of Education Bureau of Anhui Province, China under Grant KJ2020A0726.

References

- [1] O. M. Al-Kofahi, A. E. Kamal, Survivability strategies in multihop wireless networks, *IEEE Wireless Communications*, Vol. 17, No. 5, pp. 71-80, October, 2010.
- [2] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, F. Senel, Topology management techniques for tolerating node failures in wireless sensor networks: A survey, *Computer Networks*, Vol. 58, pp. 254-283, January, 2014.
- [3] S. Shrivastav, D. Ghose, Round-table negotiation for fast restoration of connectivity in partitioned wireless sensor networks, *Ad Hoc Networks*, Vol. 77, pp. 11-27, August, 2018.
- [4] G. Q. Xie, H. R. Xu, Y. Li, X. B. Hu, C. D. Wang, Fast distributed consensus seeking in large-scale and high-density multi-agent systems with connectivity maintenance, *Information Sciences*, Vol. 608, pp. 1010-1028, August, 2022.
- [5] G. Rajeswari, K. Murugan, Healing of large-scale failures in WSN by the effectual placement of relay nodes, *IET Communications*, Vol. 14, No. 17, pp. 3030-3038, October, 2020.
- [6] W. Lalouani, M. Younis, N. Badache, Interconnecting isolated network segments through intermittent links, *Journal of Network and Computer Applications*, Vol. 108, pp. 53-63, April, 2018.
- [7] X. Liu, A. Liu, T. Qiu, B. Dai, T. Wang, L. Yang, Restoring connectivity of damaged sensor networks for long-term survival in hostile environments, *IEEE Internet of Things Journal*, Vol. 7, No. 2, pp. 1205-1215, February, 2020.
- [8] W. Tian, Z. Jiao, M. Liu, M. Zhang, D. Li, Cooperative communication based connectivity recovery for UAV networks, *The ACM Turing Celebration Conference-China*, Chengdu, China, 2019, pp. 1-5.
- [9] K. Nitesh, P. K. Jana, Relay Node Placement with Assured Coverage and Connectivity: A Jarvis March Approach, *Wireless Personal Communications*, Vol. 98, No. 1, pp. 1361-1381, January, 2018.
- [10] N. T. Hanh, H. T. T. Binh, N. V. Son, P. N. Lan, Minimal Node Placement for Ensuring Target Coverage With Network Connectivity and Fault Tolerance Constraints in Wireless Sensor Networks, *IEEE Congress on Evolutionary Computation*, Wellington, New Zealand, 2019, pp. 2923-2930.
- [11] O. Dagdeviren, V. K. Akram, B. Tavli, Design and evaluation of algorithms for energy efficient and complete determination of critical nodes for wireless sensor network reliability, *IEEE Transactions on Reliability*, Vol. 68, No. 1, pp. 280-290, March, 2019.
- [12] K. K. Nagalapur, E. G. Strom, F. Brannstrom, J. Carlsson, K. Karlsson, Robust connectivity with multiple directional antennas for vehicular communications, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 12, pp. 5305-5315, December, 2020.
- [13] A. Simonetto, T. Keviczky, R. Babuska, Constrained distributed algebraic connectivity maximization in robotic networks, *Automatica*, Vol. 49, No. 5, pp. 1348-1357, May, 2013.
- [14] M. M. Zavlanos, G. J. Pappas, Potential fields for maintaining connectivity of mobile networks, *IEEE Transactions on robotics*, Vol. 23, No. 4, pp. 812-816, August, 2007.
- [15] V. K. Akram, O. Dagdeviren, B. Tavli, A Coverage-Aware Distributed k-Connectivity Maintenance Algorithm for Arbitrarily Large k in Mobile Sensor Networks, *IEEE-ACM Transactions on Networking*, Vol. 30, No. 1, pp. 62-75, February, 2022.
- [16] F. Ahmed, H. Mahmood, Y. Niaz, Mobility modelling for urban traffic surveillance by a team of unmanned aerial vehicles, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 36, No. 2, pp. 89-100, February, 2021.
- [17] M. Chen, H. Wang, C.-Y. Chang, X. Wei, SIDR: A Swarm Intelligence-Based Damage-Resilient Mechanism for UAV Swarm Networks, *IEEE Access*, Vol. 8, pp. 77089-77105, May, 2020.
- [18] D. Mosk-Aoyama, Maximum algebraic connectivity augmentation is NP-hard, *Operations Research Letters*, Vol. 36, No. 6, pp. 677-679, November, 2008.
- [19] R. P. Brent, *Algorithms for Minimization Without Derivatives*, Courier Corporation, 2013.

Biographies



Huibin Wang received his Ph. D. degree in computer science and technology from Army Engineering University of PLA, Nanjing, China. He works as a lecturer in the College of Computer and Information Engineering, Chuzhou University, Chuzhou, China. His research interests include UAV networks and distributed computing.



Ming Chen received the Ph. D. degree from Institute of Communication Engineering, Nanjing, China, in 1991. He is currently a professor in College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research interests include UAV

networks, network measurement, future networks.



Xianglin Wei received the Ph.D. degree from the PLA University of Science and Technology, Nanjing, China, in 2012. He is currently working as a Researcher with The 63rd Research Institute, National University of Defense Technology, Nanjing. His research interests include mobile edge computing, wireless network optimization, and Internet of Things.