# Resilient UAV Swarm with Fast Connectivity Recovery and Extensive Coverage

**Yabin Peng[1,2], Chenyu Zhou[1,2*], Hainan Cui[3*], Tong Duan[4], Haoyang Chen[2], Fan Zhang[2,5], Shaoxun Liu[2]**

[1]School of Cyber Science and Engineering, Southeast University, Nanjing, China
[2]Purple Mountain Laboratories, Nanjing, China
[3]Institute of Automation, Chinese Academy of Sciences, Beijing, China
[4]PLA Information Engineering University, Zhengzhou, China
[5]Institute of Big Data, Fudan University, Shanghai, China
{yabinpeng, zhouchenyu}@seu.edu.cn, hncui@nlpr.ia.ac.cn, tduan@ualberta.ca,
chenhaoyang@pmlabs.com.cn, ffzhang@fudan.edu.cn, liushaoxun@pmlabs.com.cn

## Abstract

To address partial node failures in unmanned aerial vehicle swarms, self-healing communication techniques are commonly employed to restore backbone connectivity while preserving area coverage. However, existing heuristic methods struggle to scale under large-scale failures and dynamic conditions, while learning-based approaches often suffer from spatial collapse, resulting in significant coverage loss. To overcome these limitations, we propose a resilient self-healing framework that enables rapid connectivity recovery and wide-area coverage through a divide-and-conquer strategy. First, we introduce a buffered dynamic virtual force expansion mechanism that categorizes pairwise distances into repulsive, neutral, and attractive zones, allowing nodes to disperse appropriately while preserving communication links and maintaining safety buffers. Subsequently, we design a multipartite graph convolution module to reason over subnetwork-level interactions and facilitate cross-subnetwork reconnection with global structural awareness. Finally, we develop an adaptive fusion strategy that combines both outputs with time-aware weighting to generate the final motion decisions. Experimental results in both random and uniform deployment scenarios demonstrate that our approach outperforms many state-of-the-art methods in terms of connectivity restoration speed and communication coverage.

## Introduction

Unmanned aerial vehicle (UAV) swarms have become an effective solution for rapid and flexible deployment of wireless communication (Coletta et al. 2023; Mutzari, Aumann, and Kraus 2025). They can provide high-quality communication coverage for users in complex environments, such as post-disaster rescue operations, dynamic battlefields and congested areas (Shi et al. 2025; Wang et al. 2024b, 2022). However, in complex and open environments, UAVs usually suffer from partial node failures due to internal faults or external attacks (Wang et al. 2024a; Kurunathan et al. 2023). Once the network is split into multiple subnetworks, the communication service quality and coverage performance will degrade significantly (Li et al. 2023; Xu et al. 2022; Peng et al. 2022). To address this, the self-healing communication connectivity (SCC) technique is proposed to dynamically repositions remaining UAVs to rapidly restore network connectivity while simultaneously optimizing both communication coverage and connection quality.

The heuristic SCC methods mainly contain two classes: the first involves identifying failed key nodes and compensating for them, which is computationally expensive and suitable only for small-scale failures (Tosun et al. 2024; Mi and Yang 2015); the second involves gathering the remaining nodes or subnets to predefined rendezvous points, which heavily relies on expert knowledge and iterative solving processes, lacking adaptability to dynamic and uncertain environments (Chen et al. 2020; Shriwastav 2018).

Recently, learning-based SCC methods (Li et al. 2025; Mou et al. 2022; Wang et al. 2025b) using graph neural networks (GNNs) showed promising results by modeling the recovery process as an optimization problem and learning efficient solving strategies from large amounts of historical data. These methods exhibit faster recovery and stronger adaptability to different environments. However, GNN-based methods suffer from excessive node aggregation and distribution imbalance, leading to a trade-off between connectivity restoration and communication coverage. To alleviate this issue, approaches based on diffusion models and multi-hop differential subgraphs (Lin et al. 2024; Lin and Ding 2025) are proposed to guide the remaining nodes to fill damaged areas and improve spatial distribution, but these methods still struggle to achieve a stable and balanced node layout. As communication coverage is a key metric for UAV swarms performing auxiliary communication tasks, balancing connectivity recovery speed with communication coverage remains a critical challenge.

In this paper, we propose R2C, a **R**esilient self-healing algorithm that achieves rapid **C**onnectivity recovery and extensive **C**overage. In the event of node failures that result in topological fragmentation, the UAV swarm is easily split into multiple subnetworks. Hence our method adopts a divide-and-conquer paradigm. First, inspired by a buffered elastic mechanical model, an interaction mechanism based on virtual springs is introduced between nodes within each subnetwork. A dynamic virtual force field, segmented into

*Corresponding authors.

repulsive, neutral, and attractive zones, guides the nodes to maintain an ideal physical distance, generating node-level velocity vectors. This strategy effectively achieves stable expansion and uniform distribution within the subnetwork. Second, to accelerate the reconnection between fractured subnetworks, the topological relationship between the subnetworks is modeled as a multipartite graph structure, and a multipartite graph convolution module is designed to infer inter-subnetwork interactions, generating subnetwork-level velocity vectors, thus enabling coordinated translation with global structural awareness. Finally, considering that the importance of local expansion and global translation dynamically changes under different failure scales and recovery stages, this paper introduces an adaptive weight fusion mechanism. It dynamically adjusts the fusion weights of the two velocity vectors according to the recovery time steps, resulting in the final decision velocity. Extensive experiments on both random and uniform distribution scenarios show our method outperforms many of the state-of-the-art methods.

Our framework lies in the fact that connectivity restoration and spatial coverage are inherently complementary. When UAVs maintain an appropriate physical distance within communication range, it not only helps maximize coverage without compromising connectivity but also increases the potential for establishing connections between boundary nodes of different subnetworks, thereby accelerating the overall connectivity recovery speed of the network. The main contributions are summarized as follows:

- We propose the first resilient self-healing framework that fuses local virtual force fields with global multipartite graph convolution, enabling UAV swarms to achieve rapid connectivity restoration and coverage expansion.
- A buffered dynamic virtual force mechanism is introduced for intra-subnetwork adaptive expansion and stable distribution, and a "neutral zone" is proposed to enhance the trade-off between collision avoidance and coverage balance.
- A subnetwork translation strategy is shown to fuse multipartite structural awareness into global modeling of damaged swarms, allowing subnetworks to jointly reason motion patterns and improve restoration efficiency.

## Related Work

**Non-learning methods.** These methods rely on graph principles and heuristics to formulate node relocation strategies. One line of work focuses on identifying critical nodes and restoring connectivity through cascaded movements when such nodes fail (Akkaya et al. 2009; Mi and Yang 2015; Zhang et al. 2024). However, they often incur high computational complexity and are limited to small-scale failure scenarios. To address this, aggregation-based strategies are proposed, which guide remaining nodes or subnetworks toward predefined rendezvous points for reconnection. Examples include Coulomb-force-inspired motion planning (Joshi 2012), negotiation-based coordination (Shriwastav 2018), and artificial potential field-based aggregation (Chen et al. 2020; Wang et al. 2024c). Despite their conceptual simplicity, such non-learning methods heavily depend on static assumptions and handcrafted coordination rules, making them poorly suited for dynamic and uncertain scenes.

**Learning-based methods.** The SCC can be formulated as a graph optimization problem, motivating the use of GNNs for learning recovery strategies (Peng et al. 2025; Karalias and Loukas 2020). These methods model the damaged topology, extract node features, and predict relocation trajectories for residual UAVs. Representative techniques include virtual communication-based relaxed GCNs (Mou et al. 2022), spatiotemporal attention-based graph models (Guo et al. 2024), and bipartite convolution on multi-hop differential subgraphs (Lin and Ding 2025). While these models show promising performance in recovery speed and adaptability, they often lead to UAV over-concentration around high-degree nodes due to GNN aggregation bias, causing spatial imbalance and degraded coverage. To mitigate this, recent works incorporate diffusion-based propagation (Lin et al. 2024) or differential graph modeling (Lin and Ding 2025) to encourage spatially distributed recovery. However, they still struggle to maintain global stability under dynamic conditions and often overlook real-time collision avoidance, limiting practical deployment in UAV systems.

## Preliminaries and Problem Formulation

### Network Model

**Definition 1 (UAV Swarm Network).** At each discrete time step $t$, the UAV swarm network is modeled as a temporal graph $G_t = (V_t, E_t, P_t)$, where $V_t = \{v_{1,t}, v_{2,t}, \ldots, v_{|V_t|,t}\}$ is the set of UAV nodes, $E_t = \{e_{ij,t} \mid i \neq j, 1 \leq i, j \leq |V_t|\}$ is the set of communication links, and $P_t = \{\mathbf{p}_{1,t}, \mathbf{p}_{2,t}, \ldots, \mathbf{p}_{|V_t|,t}\}$ denotes UAV positions. Assuming all UAVs operate on a common horizontal plane, the position of UAV $v_i$ at time $t$ is given by $\mathbf{p}_{i,t} = [x_{i,t}, y_{i,t}]^\top$. The distance between $v_i$ and $v_j$ is defined as $d_{ij,t} = \|\mathbf{p}_{i,t} - \mathbf{p}_{j,t}\|_2$.

**Definition 2 (Network Connectivity).** Partial failures may split the network into multiple disconnected subnetworks. UAVs within the same subnetwork can communicate directly, while those in different subnetworks cannot. Let $\mathbf{A}_t$ be the adjacency matrix, and $\mathbf{D}_t = \mathrm{diag}(d_{1,t}, \ldots, d_{|V_t|,t})$ the degree matrix. The Laplacian matrix is $\mathbf{L}_t = \mathbf{D}_t - \mathbf{A}_t$. According to spectral graph theory (Zeng et al. 2023), the number of connected components $C_t$ equals the algebraic multiplicity of the zero eigenvalue of $\mathbf{L}_t$:

$$C_t = |\{\, k \mid \lambda_{k,t} = 0,\ 1 \leq k \leq |V_t|\}|\,, \tag{1}$$

where $\lambda_{k,t}$ denotes the $k$-th eigenvalue of $\mathbf{L}_t$. The network is connected if and only if $C_t = 1$.

**Definition 3 (Maximum Communication Range).** The communication channel between UAVs is modeled as an air-to-air link (Ma et al. 2019; Wang et al. 2025a). At time step $t$, the received power at UAV $v_i$ from UAV $v_j$ is:

$$P_{\mathrm{R}}(\mathbf{p}_{i,t}, \mathbf{p}_{j,t}) = P_{\mathrm{T}} + G_{\mathrm{T}} + G_{\mathrm{R}} - L_{\mathrm{LS}}(\mathbf{p}_{i,t}, \mathbf{p}_{j,t}), \tag{2}$$

where $P_{\mathrm{T}}$ is the transmission power, $G_{\mathrm{T}}$ and $G_{\mathrm{R}}$ are the transmitter and receiver antenna gains, and $L_{\mathrm{LS}}$ is the large-scale path loss. Under unobstructed line-of-sight, the path loss is dominated by free-space attenuation (Ni et al. 2024):

$$L_{\mathrm{LS}}(\mathbf{p}_{i,t}, \mathbf{p}_{j,t}) = 10\alpha \log_{10}\left(\frac{4\pi d_{ij,t} f_c}{v_c}\right), \tag{3}$$

where $\alpha > 0$ is the path loss exponent, $f_c$ is the carrier frequency, and $v_c$ is the speed of light. A link is established if and only if the received power exceeds a threshold $P_0$. The maximum communication distance is:

$$R_c = \frac{v_c}{4\pi f_c} \exp\left(\frac{\ln(10)}{10\alpha}(P_{\mathrm{T}} + G_{\mathrm{T}} + G_{\mathrm{R}} - P_0)\right). \quad (4)$$

Thus, a direct link exists between $v_i$ and $v_j$ if $d_{ij,t} \le R_c$.

**Definition 4 (Recovery Time Ratio).** Let $\{t_1, t_2, \ldots, t_T\}$ be a time window of $T$ steps, where node failure occurs at $t_1$, and $t_r$ is the earliest time when $C_{t_r} = 1$. The recovery time ratio is defined as:

$$T_{\mathrm{rec}} = (t_r - t_1)/T. \quad (5)$$

**Definition 5 (Communication Coverage).** Let $\mathcal{A} \subset \mathbb{R}^2$ denote the mission area, and assume each UAV provides circular coverage of radius $R_{\mathrm{cov}}$. The coverage area of UAV $v_i$ at time $t$ is defined as:

$$\mathcal{C}_i(t) = \left\{\mathbf{p} \in \mathbb{R}^2 \,\middle|\, \|\mathbf{p} - \mathbf{p}_{i,t}\| \le R_{\mathrm{cov}}, \mathbf{p}_{i,t} \in \mathcal{A}\right\}. \quad (6)$$

The swarm's overall communication coverage at time $t$ is:

$$C_{\mathrm{cov}}(t) = \frac{|\bigcup_i \mathcal{C}_i(t)|}{|\mathcal{A}|}. \quad (7)$$

## Problem Definition

Given initial UAV states and failure dynamics, the objective is to design a velocity control policy $\hat{V}_t = \{\hat{\mathbf{v}}_{1,t}, \ldots, \hat{\mathbf{v}}_{|V_t|,t}\}$ that minimizes:

$$\begin{aligned}
\mathcal{P}1: \quad & \min_{\hat{V}_t} \quad J_c = T_{\mathrm{rec}} + (1 - C_{\mathrm{cov}}(t_r)), \\
\text{s.t.} \quad & C_{t_r} = 1, && \text{(C1)} \\
& \|\hat{\mathbf{v}}_{i,t}\| \le \hat{v}_{\max}, \quad \mathbf{p}_{i,t} \in \mathcal{A}, && \text{(C2)} \\
& \mathbf{p}_{i,t} = \mathbf{p}_{i,t_1} + \sum_{t_1 \le k \le T} \Delta t \cdot \hat{\mathbf{v}}_{i,k}, && \text{(C3)} \\
& d_{ij,t} \le R_c, && \text{(C4)}
\end{aligned} \quad (8)$$

where $i, j \in \{1, \ldots, |V_t|\}$ and $t \in \{1, \ldots, T\}$. Constraints (C1)–(C4) respectively ensure: network connectivity restoration, velocity and position limits, position update dynamics, and pairwise communication feasibility.

# Method

The method framework is shown in Fig. 1, where the decision process is decided on two levels: intra-subnetwork expansion and inter-subnetwork translation. Given a damaged swarm topology, the intra-subnetwork expansion module categorizes pairwise distances into attractive, repulsive, and neutral zones, which make UAVs maximize coverage while maintaining communication connectivity and exploring potential external connections. The inter-subnetwork translation module is based on the multipartite graph convolution. According to global structural awareness, this module guides the subnetworks to move cooperatively and quickly rebuild cross-subnet connections. Finally, the swarm topology is restored by adaptive weighted fusion.

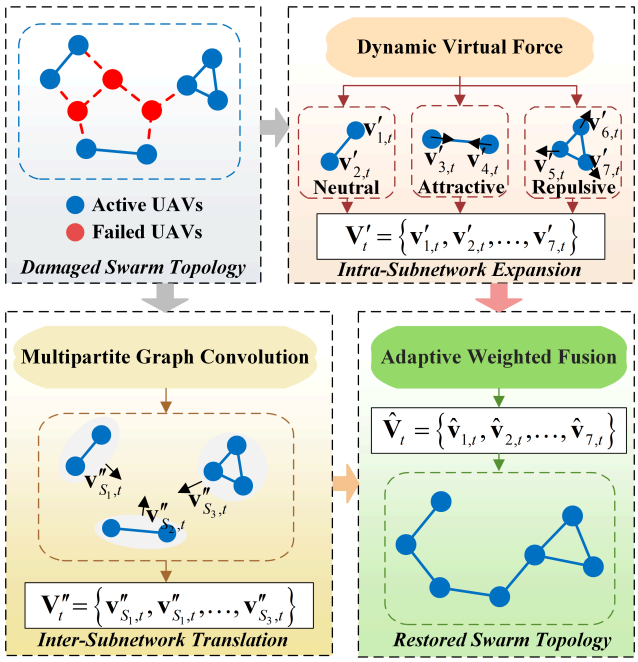


Figure 1: Framework of our method.

## Intra-Subnetwork Expansion

Inspired by Hooke's law (Rychlewski 1984), we model inter-UAV interactions as virtual springs: repulsive forces arise when UAVs are too close, attractive forces when too far, and no force within an ideal distance range. Unlike traditional springs with a single equilibrium point, we introduce a buffered spring model where the neutral zone spans a distance interval to better preserve inter-UAV spacing. This design retains elastic behavior with spacing tolerance, promoting both collision avoidance and coverage efficiency.

Assuming the optimal inter-UAV distance lies within a neutral zone $[d_l, d_u]$, we define a piecewise virtual force model inspired by buffered spring mechanics:

- **Repulsive Zone** ($d_{ij} < d_l$): UAVs are close, triggering a repulsive force to reduce overlap and potential collisions;
- **Neutral Zone** ($d_{ij} \in [d_l, d_u]$): No force is applied as the spacing is optimal;
- **Attractive Zone** ($d_{ij} > d_u$): UAVs are far, triggering an attractive force to avoid communication loss beyond range.

The corresponding virtual force applied to UAV $v_i$ from $v_j$ is defined as:

$$\mathbf{F}'_{ij,t} = \begin{cases} -\exp\left(-\alpha \cdot \frac{d_{ij,t}}{d_l}\right) \cdot \mathbf{u}_{ij,t}, & 0 \le d_{ij,t} < d_l \\ 0, & d_l \le d_{ij,t} \le d_u \\ \exp\left(-\beta \cdot \frac{R_c - d_{ij,t}}{R_c - d_u}\right) \cdot \mathbf{u}_{ij,t}, & d_u < d_{ij,t} \le R_c \end{cases} \quad (9)$$

where $\mathbf{u}_{ij,t} = (\mathbf{p}_{j,t} - \mathbf{p}_{i,t})/d_{ij,t}$ is the unit vector from $v_i$ to $v_j$, and $\alpha, \beta > 0$ control the decay rates of the repulsive and attractive forces, respectively.

To prevent UAVs from exiting the deployment area, we introduce edge repulsive forces near boundaries. Let $d_e$ denote the buffer width. If $v_i$ is within $d_e$ of any boundary $\mathcal{B} = \{\text{left}, \text{right}, \text{bottom}, \text{top}\}$, it endures an inward force:

$$\mathbf{F}''_{i,t} = \sum_{k \in \mathcal{B}} \exp(-\delta \cdot d_{i,k}) \cdot \mathbf{n}_k, \tag{10}$$

where $d_{i,k}$ is the distance from $v_i$ to boundary $k$, $\mathbf{n}_k$ is the inward normal vector, and $\delta > 0$ controls the decay rate.

The total virtual force on UAV $v_i$ at time $t$ is:

$$\mathbf{F}_{i,t} = \sum_{j \in \mathcal{N}_i} \mathbf{F}'_{ij,t} + \mathbf{F}''_{i,t}, \tag{11}$$

where $\mathcal{N}_i$ is the set of neighbors within communication range. Based on the virtual force $\mathbf{F}_{i,t}$, the motion velocity of $v_i$ at time step $t$ is:

$$\mathbf{v}'_{i,t} = \hat{v}_{\max} \cdot \frac{\mathbf{F}_{i,t}}{\|\mathbf{F}_{i,t}\| + \epsilon}, \tag{12}$$

where $\epsilon > 0$ prevents division by zero. The velocity direction follows the total force, with magnitude capped by $\hat{v}_{\max}$.

## Inter-Subnetwork Translation

After a failure, the swarm topology fragments into multiple disconnected subnetworks (or subgraphs). For connectivity recovery, a key challenge is determining the overall translation direction of subnetworks based on the relative topology among their boundary nodes. Noting that failure naturally partitions the swarm into distinct node sets, we construct a multipartite graph by introducing virtual edges between representative boundary nodes of different subnetworks. This structure captures inter-subnetwork relations and enables coordinated global translation. Based on it, we perform multipartite graph convolution to propagate structural context and predict subnetwork-level translation vectors.

**Multipartite Graph Modeling:** At time $t$, the damaged swarm is partitioned into disconnected subgraphs $S_t = \{G_t^{(1)}, G_t^{(2)}, \ldots, G_t^{(|S_t|)}\}$. To capture each subgraph's geometry, we apply the convex hull algorithm (Barber, Dobkin, and Huhdanpaa 1996) to extract representative boundary nodes $\mathcal{B}_t^{(k)}$ from each $G_t^{(k)}$. We then construct a virtual multipartite graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$, where the node set $\mathcal{V}_t$ includes all boundary nodes and their $M$ nearest neighbors from other subgraphs within an extended communication range $\hat{R}_{\text{com}}$. Since communication range is positively correlated with transmission power (defined in Eq. (4)), we temporarily boost the power of boundary nodes to support these virtual cross-subgraph links. Notably, neighboring nodes are not necessarily on the convex hull, which helps capture concave boundaries more accurately and mitigates the limitations of convex hull approximation.

Virtual edges are established from each $u \in \mathcal{B}_t^{(i)}$ to its top-$M$ nearest neighbors $v \in \bigcup_{j \neq i} V^{(j)}$ within $\hat{R}_{\text{com}}$:

$$\mathcal{E}_t = \{(u, v) \mid u \in \mathcal{B}_t,\ v \in \text{Top-}M(u)\}. \tag{13}$$

This construction enhances cross-subgraph structural awareness and effectively supports information propagation for coordinated translation.

**Multipartite Graph Convolution**: To achieve coordinated translation between subgraphs, we design a spectral approximation-based multipartite graph convolution module and introduce a subnetwork-level action generation strategy. On the constructed virtual multipartite graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$, we define its weighted adjacency matrix as:

$$\tilde{\mathbf{A}}_t(i,j) = \begin{cases} 1 - \dfrac{\|\mathbf{p}_i - \mathbf{p}_j\|}{\hat{R}_{\text{com}}}, & (i,j) \in \mathcal{E}_t \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

and construct the normalized Laplacian matrix $\tilde{\mathbf{L}}_t$ accordingly. To reduce the computational complexity of spectral graph convolution, we adopt the first-order Chebyshev approximation (Liu and Wang 2025), and the kernel is:

$$\mathbf{H}_t^{(L)} = \sigma\left((\mathbf{I} - \epsilon\tilde{\mathbf{L}}_t)\mathbf{H}_t^{(L-1)}\mathbf{W}^{(L)}\right), \tag{15}$$

where $\epsilon > 0$ is the spectral scaling factor, $\mathbf{H}_t^{(0)} \in \mathbb{R}^{|\mathcal{V}_t| \times d}$ encodes initial node features (positions and subnetwork IDs), $\mathbf{W}^{(L)} \in \mathbb{R}^{d \times d'}$ is a learnable weight matrix, and $\sigma(\cdot)$ is the ReLU activation.

To obtain subnetwork-level motion commands, we apply average pooling to each subnetwork $\mathcal{V}_t^{(k)}$ to extract its embedding:

$$\mathbf{z}_t^{(k)} = \frac{1}{|\mathcal{V}_t^{(k)}|} \sum_{v \in \mathcal{V}_t^{(k)}} \mathbf{H}_t^{(L)}[v]. \tag{16}$$

The embedding $\mathbf{z}_t^{(k)} \in \mathbb{R}^{d'}$ is passed through a two-layer MLP $\varphi(\cdot)$ to predict a normalized 2D translation vector:

$$\hat{\boldsymbol{\delta}}_t^{(k)} = \tanh(\varphi(\mathbf{z}_t^{(k)})) + \mathbf{1} \in (0,2)^2. \tag{17}$$

We then map $\hat{\boldsymbol{\delta}}_t^{(k)}$ to the physical space using a scaling factor $\mathbf{s} = [0.5L_x, 0.5L_y]$, which gives the target position for the subnetwork: $\tilde{\mathbf{p}}_t^{(k)} = \mathbf{s} \odot \hat{\boldsymbol{\delta}}_t^{(k)}$, where $\odot$ denotes element-wise multiplication. Comparing it with the current center of the subnetwork $\mathbf{c}_t^{(k)}$ yields the subnetwork translation velocity:

$$\mathbf{v}''^{(k)}_t = \hat{v}_{\max} \cdot \frac{\tilde{\mathbf{p}}_t^{(k)} - \mathbf{c}_t^{(k)}}{\left\|\tilde{\mathbf{p}}_t^{(k)} - \mathbf{c}_t^{(k)}\right\|}. \tag{18}$$

**Joint Loss Function Design:** To guide the multipartite graph convolution network in learning effective subnetwork translation strategies, we design a joint loss function: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{conn}} + \mathcal{L}_{\text{time}} + \mathcal{L}_{\text{move}}$, where the three components are defined as follows:

- **Connectivity Loss $\mathcal{L}_{\text{conn}}$.** This term measures the number of connected components in the resulting topology:

$$\mathcal{L}_{\text{conn}} = \lambda(C_t - 1), \tag{19}$$

  where $\lambda$ is a positive weighting coefficient and $C_t$ denotes the number of connected components at time $t$.

- **Recovery Time Loss $\mathcal{L}_{\text{time}}$.** This term evaluates the maximum time required for subnetworks to translate from their current to target positions:

$$\mathcal{L}_{\text{time}} = \max_k \left( \frac{\left\|\tilde{\mathbf{p}}_t^{(k)} - \mathbf{c}_t^{(k)}\right\|}{\hat{v}_{\max}} \right). \tag{20}$$

- **Average Displacement Loss $\mathcal{L}_{\text{move}}$.** This term constrains the overall movement cost of all UAVs:

$$\mathcal{L}_{\text{move}} = \frac{1}{|V_t|} \sum_{v_i \in V_t} \|\hat{\mathbf{p}}_i - \mathbf{p}_i\| . \tag{21}$$

## Adaptive Weighted Fusion

Both intra-subnetwork expansion (ISE) and inter-subnetwork translation (IST) are essential for connectivity recovery, but their importance varies across failure scales and recovery stages. Under small-scale failures, ISE alone suffices as subnetworks remain physically close. In contrast, large-scale failures require IST to coordinate global subnetwork migration for effective reconnection.

As the failure scale is typically unknown yet positively correlated with the recovery step, we design a time-step-based adaptive weighting function (AWF) to smoothly adjust the fusion ratio between ISE and IST strategies. At recovery step $t$, the fusion weight is computed using the following Sigmoid function (Elfwing, Uchibe, and Doya 2018):

$$w_{\text{T}}(t) = \frac{1}{1 + \exp(-k(t/T - s_0))}, \quad w_{\text{E}}(t) = 1 - w_{\text{T}}(t), \tag{22}$$

where $T$ is the maximum recovery step, $k > 0$ controls the transition slope, and $s_0 \in (0, 1)$ is the fusion center. This function enables a smooth transition from ISE-dominated to IST-dominated strategies.

The final velocity $\hat{\mathbf{v}}_{i,t}$ for UAV $v_i$ is a normalized weighted combination of its node-level action $\mathbf{v}'_{i,t}$ and the projected subnetwork-level action $\mathbf{v}''_{i,t}$:

$$\hat{\mathbf{v}}_{i,t} = \hat{v}_{\max} \cdot \frac{w_{\text{E}}(t)\mathbf{v}'_{i,t} + w_{\text{T}}(t)\mathbf{v}''_{i,t}}{\left\| w_{\text{E}}(t)\mathbf{v}'_{i,t} + w_{\text{T}}(t)\mathbf{v}''_{i,t} \right\| + \epsilon}, \tag{23}$$

where $\epsilon$ is a small constant to avoid division by zero. This fusion ensures consistent responsiveness across varying failure scales and recovery phases.

## Complexity Analysis

Let $N$ be the number of remaining UAVs, $E$ the number of communication links, $L$ the number of graph convolution layers, $d$ the feature dimension, $n_B$ the number of convex-hull boundary nodes, and $E_M$ the number of virtual edges in the multipartite graph. The total computational cost per time step is composed of: 1) **ISE**: Virtual-force computation relies on neighbor aggregation with time complexity $\mathcal{O}(|E|)$; 2) **IST**: Constructing the virtual multipartite graph requires nearest-neighbor search with $\mathcal{O}(n_B \log N)$ time using a $k$-d tree, followed by $L$ layers of first-order spectral graph convolution with complexity $\mathcal{O}(L\,|E_M|\,d)$; 3) **AWF**: Weight computation and action fusion are performed per subnetwork and incur $\mathcal{O}(N)$ complexity. Thus, the total per-step complexity is $\mathcal{O}\big(|E| + n_B \log N + L|E_M|d + N\big)$. In typical scenarios, $n_B \ll N$ and $E_M \ll E$, so the overall complexity grows approximately linearly with the node scale, satisfying the real-time recovery requirements for large-scale UAV swarms.

# Experiments

## Experimental Setup

**Deployment Scenarios:** Our method is evaluated in a $1000 \times 1000\text{m}^2$ area under two representative UAV swarm scenarios: 1) Random Deployment: 200 UAVs are randomly positioned following the approach in (Mou et al. 2022; Lin and Ding 2025); 2) Uniform Deployment: 196 UAVs are uniformly arranged on a $14 \times 14$ grid. All experiments adopt unpredictable random failure patterns, with failure ratios ranging from $20\%$ to $80\%$. For each case, 50 independent simulations are conducted to ensure statistical robustness.

**Parameter Settings:** Each UAV has both a communication and coverage radius of $120\,\text{m}$, the maximum flight speed is constrained to $\hat{v}_{\max} = 1\,\text{m/s}$, and the overall recovery horizon is set to $T = 400$ time steps. The proposed model employs an 8-layer graph convolutional network with an input feature dimension of $d = 3$ and a hidden dimension of $d' = 512$, and is optimized using Adam with a learning rate of $1 \times 10^{-4}$. In the R2C-specific algorithm, the optimal distance range is set to $[d_l, d_u] = [0.8R_c, 0.9R_c]$; the repulsion, attraction, and boundary decay coefficients are all fixed at $0.01$; the inter-subnetwork translation is triggered every $T_I = 50$ steps; and the fusion mechanism is parameterized by a transition center $s_0 = 0.125$ and a slope $k = 30$.

**Baseline Methods and Hardware Environment:** As shown in Table 1, our method is compared against eight methods, where four are non-learning approaches based on heuristic strategies (CEN, HERO, SIDR, MDSG-APF) and the other four are learning-based methods utilizing graph neural networks (GCN, CR-MGC, DEMD, MDSG-GC). All simulations are conducted on a server equipped with an NVIDIA A100 GPU and an Intel Core i9-9900X CPU.

## Quantitative Comparison

We evaluate our method (R2C) under varying node failure rates (20% to 80%) in both random and uniform deployment scenarios, focusing on two key metrics: recovery time (REC, Eq. (5)) and communication coverage (COV, Eq. (7)). Since connectivity restoration is a fundamental requirement for SCC, if a method fails to recover connectivity in all 50 trials, it is considered invalid and marked as "–" in the table. (1) Random Deployment: As shown in Table 1, R2C achieves the shortest average recovery time and the highest average coverage rate. Even under extreme failure conditions (80% failure), R2C maintains strong robustness, improving COV by over 60% compared to the best baseline MDSG-APF. (2) Uniform Deployment: As shown in Table 2, R2C again achieves the lowest REC and highest COV. In summary, R2C consistently achieves the fastest connectivity restoration and the highest communication coverage.

To validate the practical applicability, we evaluate algorithm efficiency on the Jetson Xavier NX platform, a widely adopted onboard computer in UAV systems (Ye et al. 2022). As shown in Table 1 and 2, although the heuristic methods such as HERO and SIDR exhibit shorter runtimes, their performances in connectivity restoration and coverage are significantly inferior to R2C. Compared with MDSG-APF, R2C adds 0.66 seconds of computation but reduces recovery

| Method | 20% | | 40% | | 60% | | 80% | | Avg. | | Avg. Runtime | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | MEAN(s) | STD(s) |
| CEN (Abbasi and Younis 2008) | 0.184 | 0.909 | 0.379 | 0.731 | 0.737 | 0.410 | 0.881 | 0.246 | 0.545 | 0.574 | 0.076 | 0.016 |
| HERO (Mi, Yang, and Liu 2011) | 0.146 | 0.990 | 0.714 | 0.968 | – | – | – | – | – | – | 0.096 | 0.035 |
| SIDR (Chen et al. 2020) | 0.145 | 0.994 | 0.462 | 0.987 | 0.863 | 0.940 | – | – | – | – | 0.101 | 0.031 |
| MDSG-APF (Lin and Ding 2025) | 0.074 | 0.988 | 0.203 | 0.919 | 0.427 | 0.731 | 0.733 | 0.416 | 0.359 | 0.764 | 0.681 | 0.104 |
| GCN (Kipf and Welling 2016) | 0.196 | 0.945 | 0.495 | 0.777 | 0.855 | 0.504 | 0.985 | 0.382 | 0.633 | 0.652 | 2.786 | 0.305 |
| CR-MGC (Mou et al. 2022) | 0.145 | 0.958 | 0.388 | 0.762 | 0.645 | 0.527 | 0.824 | 0.310 | 0.501 | 0.639 | 3.592 | 0.534 |
| DEMD (Lin et al. 2024) | 0.178 | 0.941 | 0.379 | 0.799 | 0.628 | 0.550 | 0.835 | 0.322 | 0.505 | 0.653 | 9.581 | 1.857 |
| MDSG-GC (Lin and Ding 2025) | 0.052 | 0.993 | 0.139 | 0.964 | 0.387 | 0.772 | 0.779 | 0.393 | 0.339 | 0.780 | 5.234 | 0.904 |
| Ours | **0.023** | **0.996** | **0.070** | **0.989** | **0.204** | **0.951** | **0.625** | **0.681** | **0.231** | **0.904** | 1.542 | 0.219 |

Table 1: Comparison of REC and COV under varying failure levels under random deployment scenario. Bold and underlined represent the best and second-best performance, respectively. "–" denotes invalid cases.

| Method | 20% | | 40% | | 60% | | 80% | | Avg. | | Avg. Runtime | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | MEAN(s) | STD(s) |
| CEN (Abbasi and Younis 2008) | 0.378 | 0.813 | 0.350 | 0.817 | 0.590 | 0.565 | 0.927 | 0.249 | 0.561 | 0.611 | 0.069 | 0.019 |
| HERO (Mi, Yang, and Liu 2011) | 0.045 | **1.000** | 0.058 | **0.999** | 0.979 | 0.975 | – | – | – | – | 0.084 | 0.040 |
| SIDR (Chen et al. 2020) | 0.370 | **1.000** | 0.332 | 0.998 | 0.664 | 0.969 | – | – | – | – | 0.095 | 0.036 |
| MDSG-APF (Lin and Ding 2025) | 0.177 | 0.998 | 0.190 | 0.987 | 0.305 | 0.888 | 0.700 | 0.475 | 0.343 | 0.837 | 0.614 | 0.078 |
| GCN (Kipf and Welling 2016) | 0.205 | 0.972 | 0.400 | 0.877 | 0.794 | 0.582 | 0.985 | 0.411 | 0.596 | 0.710 | 2.296 | 0.289 |
| CR-MGC (Mou et al. 2022) | 0.314 | 0.881 | 0.385 | 0.814 | 0.635 | 0.565 | 0.899 | 0.302 | 0.558 | 0.640 | 2.940 | 0.445 |
| DEMD (Lin et al. 2024) | 0.235 | 0.926 | 0.327 | 0.860 | 0.549 | 0.637 | 0.796 | 0.351 | 0.477 | 0.694 | 7.009 | 1.395 |
| MDSG-GC (Lin and Ding 2025) | 0.068 | **1.000** | 0.110 | 0.992 | 0.323 | 0.863 | 0.867 | 0.374 | 0.342 | 0.807 | 3.295 | 0.626 |
| Ours | **0.037** | **1.000** | **0.051** | **0.999** | **0.181** | **0.978** | **0.617** | **0.668** | **0.222** | **0.911** | 1.078 | 0.173 |

Table 2: Comparison of REC and COV under varying failure levels under uniform deployment scenario.

| Ablations | 20% | | 40% | | 60% | | 80% | |
|---|---|---|---|---|---|---|---|---|
| | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ |
| R2C | **0.023** | **0.996** | **0.070** | **0.989** | **0.204** | **0.951** | **0.625** | 0.681 |
| w/o IST | 0.024 | **0.996** | 0.107 | 0.987 | 0.477 | 0.950 | – | **0.722** |
| w/o ISE | 0.046 | 0.993 | 0.122 | 0.974 | 0.311 | 0.896 | 0.746 | 0.611 |
| w/o AWF | **0.023** | **0.996** | 0.088 | 0.980 | 0.257 | 0.950 | 0.702 | 0.696 |

Table 3: Ablation study of R2C under random deployment.

| Ablations | 20% | | 40% | | 60% | | 80% | |
|---|---|---|---|---|---|---|---|---|
| | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ | REC ↓ | COV ↑ |
| R2C | **0.037** | **1.000** | **0.051** | **0.999** | **0.181** | 0.978 | **0.617** | 0.668 |
| w/o IST | 0.038 | **1.000** | 0.058 | **0.999** | 0.403 | **0.980** | – | **0.704** |
| w/o ISE | 0.054 | **1.000** | 0.103 | 0.996 | 0.248 | 0.943 | 0.813 | 0.606 |
| w/o AWF | 0.041 | **1.000** | 0.057 | **0.999** | 0.192 | 0.978 | 0.698 | 0.689 |

Table 4: Ablation study of R2C under uniform deployment.

time by 35% and improves coverage by 13%. For learning-based methods, R2C outperforms GCN, CR-MGC, DEMD, and MDSG-GC in computational efficiency. Overall, R2C achieves a balance between performance and runtime, and its inference speed on resource-limited onboard platforms meets the real-time requirements of UAV swarm operations.

## Ablation Study

We conduct ablation experiments under both deployment scenarios by individually disabling each module: 1) w/o IST: inter-subnetwork translation is disabled; 2) w/o ISE: intra-subnetwork expansion is disabled; 3) w/o AWF: the adaptive fusion strategy is removed by fixing the weights $w_{\mathrm{T}}$ and $w_{\mathrm{E}}$ to 0.5. As illustrated in Table 3 and Table 4, the model without IST struggles to restore connectivity efficiently under large-scale failures; removing ISE results in a significant drop in communication coverage and disabling AWF leads to slower connectivity restoration. Our method achieves the fastest connectivity recovery under both deployment schemes, while maintaining comparable communication coverage to w/o IST and w/o AWF. These results demonstrate that the synergy between intra-subnetwork expansion and inter-subnetwork translation is essential for achieving both rapid connectivity restoration and effective coverage optimization. Furthermore, incorporating AWF enhances the coordination between these two mechanisms.

## Visualization Analysis

To intuitively show the recovery process, we visualize the restored topologies under a $50\%$ node failure rate for both random and uniform deployment scenarios. As shown in Fig. 2, we present results of CR-MGC, MDSG-APF, and MDSG-GC. The shaded areas represent the reconstructed communication coverage areas. CR-MGC, as the first method to incorporate GCN into connectivity restoration, suffers from severe node over-concentration, resulting in limited coverage. MDSG-GC alleviates this issue by guiding the remaining UAVs to fill uncovered areas, thereby improving spatial distribution. In contrast, our method mitigates both node clustering and insufficient coverage. It demonstrates the shortest recovery trajectories and the most uniform final distribution, indicating superior performance in both recovery speed and communication coverage.

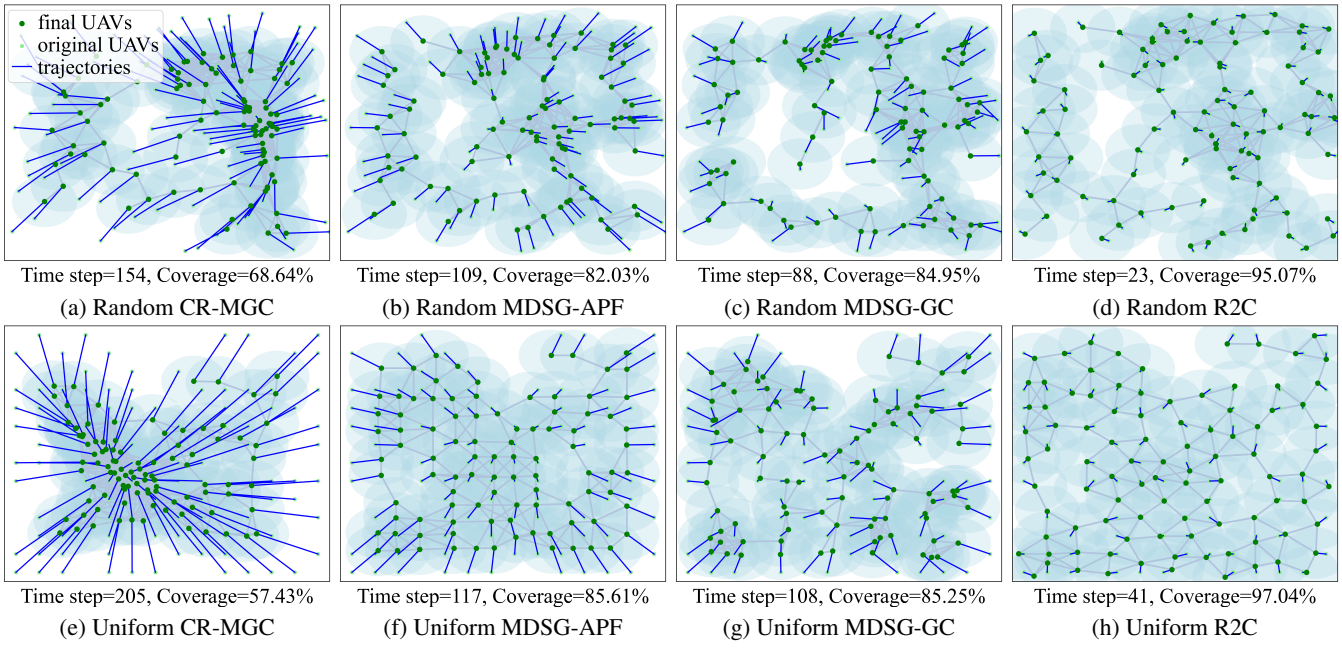


Figure 2: Visualization of restored topologies in random and uniform deployment. The legend in (a) is shared by (a)–(h).

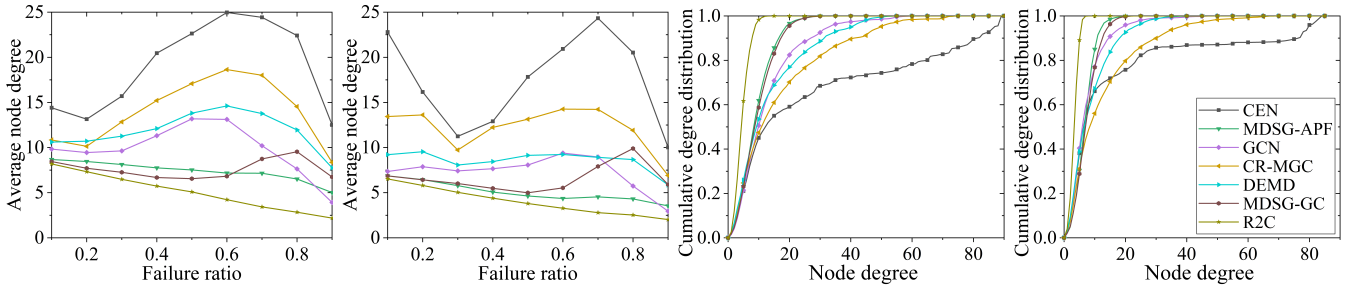


Figure 3: Topological analysis of restored networks, including average node degree (AND) under different failure ratios and cumulative degree distribution (CDD) at $50\%$ failure. The legend in (d) is shared by (a)–(d).

## Analysis of Restored Topologies

To evaluate the quality of the recovered communication topologies across different algorithms, we analyze key structural metrics as shown in Figs. 3(a)–(b). Specifically, we report the average node degree $\langle k \rangle$ under varying failure ratios, which characterizes the trade-off between link-maintenance overhead and channel reuse. A smaller $\langle k \rangle$ implies lower per-node energy consumption and interference, enabling more efficient bandwidth allocation. As shown, R2C consistently achieves the lowest and smoothest $\langle k \rangle$ in both random and uniform deployments, thereby reducing communication overhead and mitigating channel interference.

Furthermore, Figs. 3(c)–(d) show the cumulative degree distribution $P(k)$ under a $50\%$ failure ratio. R2C concentrates node degrees in the low-degree range ($k \leq 10$), indicating a more uniform recovered topology. This mitigates communication congestion and spatial overlap caused by excessive clustering. Moreover, the absence of high-degree nodes enhances the network's robustness and reduces secondary failures due to energy depletion.

Overall, R2C achieves the unity of low energy consumption, uniform topology and high robustness while guaranteeing connectivity and coverage, which lays a reliable foundation for subsequent data forwarding and cooperative control.

## Conclusion

We propose a resilient self-healing framework designed to restore connectivity in UAV swarms under partial node failures. By combining a buffered virtual force model with multipartite graph convolution, the recovery process contains intra-subnetwork expansion and inter-subnetwork translation. An adaptive fusion strategy adjusts their contributions over time to ensure efficient coordination. Extensive simulations demonstrate the effectiveness of our method and real-world evaluations on an onboard platform confirm our practical feasibility for real-time UAV swarm applications.

## Acknowledgments

This work was supported in part by the Jiangsu Provincial Department of Science and Technology (Grant No. 26042501), the National Natural Science Foundation of China (Grant No. 62073320), and the SEU Innovation Capability Enhancement Plan for Doctoral Students (Grant No. CXJH_SEU_25245).

## References

Abbasi, A. A.; and Younis, M. 2008. Movement-Assisted Connectivity Restoration in Wireless Sensor and Actor Networks. *IEEE Transactions on Parallel and Distributed Systems*, 20(9): 1366–1379.

Akkaya, K.; Senel, F.; Thimmapuram, A.; and Uludag, S. 2009. Distributed Recovery from Network Partitioning in Movable Sensor/Actor Networks via Controlled Mobility. *IEEE Transactions on Computers*, 59(2): 258–271.

Barber, C. B.; Dobkin, D. P.; and Huhdanpaa, H. 1996. The Quickhull Algorithm for Convex Hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4): 469–483.

Chen, M.; Wang, H.; Chang, C.-Y.; and Wei, X. 2020. SIDR: A Swarm Intelligence-Based Damage-Resilient Mechanism for UAV Swarm Networks. *IEEE Access*, 8: 77089–77105.

Coletta, A.; Giorgi, F.; Maselli, G.; Prata, M.; Silvestri, D.; Ashdown, J.; and Restuccia, F. 2023. A 2-UAV: Application-Aware Content and Network Optimization of Edge-Assisted UAV Systems. In *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*, 1–10. IEEE.

Elfwing, S.; Uchibe, E.; and Doya, K. 2018. Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning. *Neural Networks*, 107: 3–11.

Guo, Z.; Sun, Y.; Wang, Y.; Fu, C.; and Zhong, J. 2024. An Intelligent Framework for Resilience Recovery of FANETs with Spatio-Temporal Aggregation and Multi-Head Attention Mechanism. *Computers, Materials & Continua*, 79(2): 2375–2398.

Joshi, Y. K. 2012. Autonomous Recovery from Multi-Node Failure in Wireless Sensor Networks. In *2012 IEEE Global Communications Conference (GLOBECOM)*, 652–657. IEEE.

Karalias, N.; and Loukas, A. 2020. Erdos Goes Neural: An Unsupervised Learning Framework for Combinatorial Optimization on Graphs. *Advances in Neural Information Processing Systems*, 33: 6659–6672.

Kipf, T. N.; and Welling, M. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*.

Kurunathan, H.; Huang, H.; Li, K.; Ni, W.; and Hossain, E. 2023. Machine Learning-Aided Operations and Communications of Unmanned Aerial Vehicles: A Contemporary Survey. *IEEE Communications Surveys & Tutorials*, 26(1): 496–533.

Li, J.; Yi, P.; Duan, T.; Zhang, Z.; Li, J.; Wang, Y.; and Yu, J. 2025. Fast Connectivity Restoration of UAV Communication Networks Based on Distributed Hybrid MADDPG and APF Algorithm. *Ad Hoc Networks*, 171: 103785.

Li, J.; Zhou, C.; Liu, J.; Sheng, M.; Zhao, N.; and Su, Y. 2023. Reinforcement Learning-Based Resource Allocation for Coverage Continuity in High Dynamic UAV Communication Networks. *IEEE Transactions on Wireless Communications*, 23(2): 848–860.

Lin, H.; and Ding, L. 2025. Multi-Hop Differential Topology Based Algorithms for Resilient Network of UAV Swarm. *arXiv preprint arXiv:2411.11342v2*.

Lin, H.; Ding, L.; Chen, S.; Yang, F.; and Qian, L. 2024. Multi-Hop Diffused Graph Convolution for Resilient UAV Swarm Networks. In *2024 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 1–6. IEEE.

Liu, F.; and Wang, Q. 2025. Asymmetric Learning for Spectral Graph Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 18798–18806.

Ma, Z.; Ai, B.; He, R.; Wang, G.; Niu, Y.; and Zhong, Z. 2019. A Wideband Non-Stationary Air-to-Air Channel Model for UAV Communications. *IEEE Transactions on Vehicular Technology*, 69(2): 1214–1226.

Mi, Z.; and Yang, J. Y. 2015. Obstacle-Avoidance Connectivity Restoration for Mobile Sensor Systems with Local Information. In *2015 IEEE International Conference on Communications (ICC)*, 6395–6399.

Mi, Z.; Yang, Y.; and Liu, G. 2011. HERO: A Hybrid Connectivity Restoration Framework for Mobile Multi-Agent Networks. In *2011 IEEE International Conference on Robotics and Automation*, 1702–1707. IEEE.

Mou, Z.; Gao, F.; Liu, J.; and Wu, Q. 2022. Resilient UAV Swarm Communications with Graph Convolutional Neural Network. *IEEE Journal on Selected Areas in Communications*, 40(1): 393–411.

Mutzari, D.; Aumann, Y.; and Kraus, S. 2025. Heterogeneous Multi-Robot Graph Coverage with Proximity and Movement Constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 14646–14654.

Ni, H.; Zhu, Q.; Hua, B.; Mao, K.; Pan, Y.; Ali, F.; Zhong, W.; and Chen, X. 2024. Path Loss and Shadowing for UAV-to-Ground UWB Channels Incorporating the Effects of Built-Up Areas and Airframe. *IEEE Transactions on Intelligent Transportation Systems*, 25(11): 17066–17077.

Peng, Y.; Liu, C.; Liu, S.; Liu, Y.; and Wu, Y. 2022. SmartTRO: Optimizing Topology Robustness for Internet of Things via Deep Reinforcement Learning with Graph Convolutional Networks. *Computer Networks*, 218: 109385.

Peng, Y.; Wu, J.; Duan, T.; Liu, Y.; Zhang, Z.; and Zhang, J. 2025. Prioritized Recovery Strategy for Robust UAV Swarm Communication via Graph Reinforcement Learning. *IEEE Internet of Things Journal*, 12(13): 23891–23904.

Rychlewski, J. 1984. On Hooke's Law. *Journal of Applied Mathematics and Mechanics*, 48(3): 303–314.

Shi, R.; Yu, X.; Wang, Y.; Tian, Y.; Liu, Z.; Wu, W.; Zhang, X.-P.; and Veloso, M. M. 2025. Symmetry-Informed MARL: A Decentralized and Cooperative UAV Swarm Control Approach for Communication Coverage. *IEEE Transactions on Mobile Computing*, 24(9): 8039–8056.

Shriwastav, S. 2018. Round-Table Negotiation for Fast Restoration of Connectivity in Partitioned Wireless Sensor Networks. *Ad Hoc Networks*, 77: 11–27.

Tosun, M.; Cabuk, U. C.; Haytaoglu, E.; Dagdeviren, O.; and Ozturk, Y. 2024. DPkCR: Distributed Proactive K-Connectivity Recovery Algorithm for UAV-Based MANETs. *IEEE Transactions on Reliability*, 73(4): 1918–1932.

Wang, J.; Li, S.; Chen, D.; Fu, Q.; Qi, Q.; Sun, H.; He, B.; and Liao, J. 2025a. Flight Trajectory Control with Network-Oriented Hierarchical Reinforcement Learning for UAV-Assisted Time-Sensitive IoT. *IEEE Transactions on Intelligent Transportation Systems*, 26(5): 6332–6345.

Wang, X.; Zhao, Z.; Yi, L.; Ning, Z.; Guo, L.; Yu, F. R.; and Guo, S. 2024a. A Survey on Security of UAV Swarm Networks: Attacks and Countermeasures. *ACM Computing Surveys*, 57(3): 1–37.

Wang, Y.; Chen, W.; Luan, T. H.; Su, Z.; Xu, Q.; Li, R.; and Chen, N. 2022. Task Offloading for Post-Disaster Rescue in Unmanned Aerial Vehicle Networks. *IEEE/ACM Transactions on Networking*, 30(4): 1525–1539.

Wang, Y.; Feng, Z.; Zhang, H.; Gao, Y.; Lei, J.; Sun, L.; and Song, M. 2024b. Angle Robustness Unmanned Aerial Vehicle Navigation in GNSS-Denied Scenarios. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 10386–10394.

Wang, Y.; Wang, X.; Wei, C.; Ren, Q.; and Tang, Y. 2025b. UAV Swarm Network Topology Self-Healing via Graph-Based Deep Reinforcement Learning. In *2025 IEEE Wireless Communications and Networking Conference (WCNC)*, 1–6.

Wang, Z.; Zhao, X.; Zhang, J.; Yang, N.; Wang, P.; Tang, J.; Zhang, J.; and Shi, L. 2024c. APF-CPP: An Artificial Potential Field Based Multi-Robot Online Coverage Path Planning Approach. *IEEE Robotics and Automation Letters*, 9(11): 9199–9206.

Xu, C.; Ding, W.; Lyu, W.; Liu, Z.; Wang, S.; He, Y.; Hu, H.; Zhao, D.; and Li, B. 2022. SafeBench: A Benchmarking Platform for Safety Evaluation of Autonomous Vehicles. *Advances in Neural Information Processing Systems*, 35: 25667–25682.

Ye, Z.; Wang, K.; Chen, Y.; Jiang, X.; and Song, G. 2022. Multi-UAV Navigation for Partially Observable Communication Coverage by Graph Reinforcement Learning. *IEEE Transactions on Mobile Computing*, 22(7): 4056–4069.

Zeng, D.; Liu, W.; Chen, W.; Zhou, L.; Zhang, M.; and Qu, H. 2023. Substructure Aware Graph Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 11129–11137.

Zhang, L.; Du, Y.; Xu, J.; and Wang, X. 2024. UAV-Enabled IoT: Cascading Failure Model and Topology-Control-Based Recovery Scheme. *IEEE Internet of Things Journal*, 11(12): 22562–22577.