IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Real-Time Path-Reconfigurable Coverage Planning for Multi-UAV Missions Over Disjoint Areas

Cai Luo, *Senior Member, IEEE*, Jintao Guo, Zixuan Liu, Lei Liu, *Member, IEEE*, and Chunbo Luo

***Abstract*—Multi-uncrewed aerial vehicle (UAV) cooperative coverage missions across geographically separated regions face significant challenges due to potential UAV failures during mission execution. To address the challenges of efficient multi-region coverage and dynamic failure handling, this paper presents a novel real-time path-reconfigurable coverage-planning algorithm for multi-UAV coverage path planning across separated geographical areas with real-time path reconfiguration capabilities. The proposed approach, GRIT-M (Greedy Repair Initializes Tabu search for Multiple regions), extends the GRIT algorithm to efficiently handle both initial planning and online path repair when UAVs fail during mission execution. Unlike existing methods that either focus on single-region coverage or lack failure handling mechanisms, GRIT-M incorporates region-specific knowledge through three key innovations: (1) a composite transition cost function that optimizes inter-region movements, (2) a region-priority greedy repair mechanism that minimizes transition costs during replanning, and (3) a region-preserving Tabu search that maintains area coverage coherence while balancing workload. Extensive simulations demonstrate GRIT-M's scalability and robustness against operational constraints. The algorithm reduces inter-region flights by up to 27% compared to baselines and maintains excellent workload balance (CV $<$ 7%). Crucially, it shows effective scalability with larger teams (up to 12 UAVs), maintaining a $>$ 91% replanning success rate with average repair times under 2.1s. Its resilience to variations in battery capacity, sweep patterns, and initial UAV distributions confirms its suitability for time-critical aerial surveys over disjoint areas.**

***Index Terms*—Aerial systems: applications, multi-robot systems, path planning for multiple mobile robots or agents.**

I. Introduction

MULTI-UAV systems offer significant advantages in surveying large areas, with applications in search and rescue [1], forest fire monitoring [2], aerial mapping [3], precision agriculture [4], and wildlife monitoring [5]. These systems perform such tasks more rapidly and efficiently than single-agent approaches. However, a key challenge lies in developing Coverage Path Planning (CPP) algorithms that are both efficient and resilient for real-world deployment.

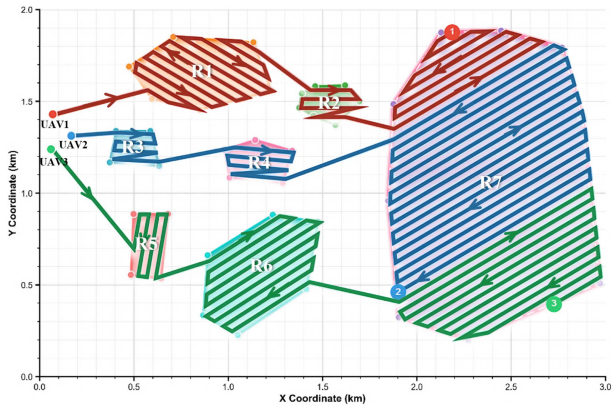


Fig. 1. Coordinated coverage of seven regions (R1–R7) by the red, blue, and green UAVs. The task allocation prioritizes assigning each region to a single UAV to ensure spatial separation and reduce overlap. Due to operational constraints such as limited battery capacity, region R7 is jointly covered by all three UAVs.

Theoretical studies in multi-uncrewed aerial vehicle (UAV) CPP typically address area segmentation and path optimization to minimize operational costs [6]. Common approaches such as lawnmower-style patterns are often used for initial path generation under ideal conditions [7].

In practice, unexpected events—such as UAV failures due to weather, battery drain, or mechanical issues—complicate preplanned routes. The challenge is magnified in missions covering multiple, geographically separated regions, which introduce additional coordination complexities and energy costs due to inter-region travel [8]. Fig. 1 illustrates such a scenario, where several UAVs must cooperatively cover disjoint regions. Recent studies have begun addressing these issues. For instance, Clark et al. [9] introduced a greedy path repair algorithm to redistribute a failed UAV's waypoints. Luna et al. [10] proposed a system for replanning coverage of multiple separated areas mid-flight. However, few approaches integrate the dual challenges of (i) efficient multi-region planning and (ii) dynamic adaptation to in-flight failures, highlighting the need for a unified framework.

This paper introduces GRIT-M (Greedy Repair Initializes Tabu search for Multiple regions), a novel approach for multi-UAV CPP that handles both initial multi-region mission planning and dynamic path repair after UAV failures. As an extension

Received 24 June 2025; accepted 17 November 2025. Date of publication 5 December 2025; date of current version 12 December 2025. This article was recommended for publication by Associate Editor S. Bhattacharya and Editor M. A. Hsieh upon evaluation of the reviewers' comments. This work was supported in part by the National Key R&D Program of China under Grant 2021YFE0111600, in part by EU Horizon 2020 under Grant 101008297, in part by UKRI under Grant EP/Y036786/1, in part by Horizon EU under Grant 101129910, in part by the Fundamental Research Funds for the Central Universities under Grant 24CX02030A, and in part by Taishan Scholars Program. *(Corresponding author: Cai Luo.)*

Cai Luo, Jintao Guo, and Zixuan Liu are with the College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao 266580, China (e-mail: tsai.lo.95@gmail.com; bigpot2001@163.com; liuzixuan684@gmail.com).

Lei Liu is with the School of Software, Shandong University, Jinan 250101, China, and also with the Shandong Research Institute of Industrial Technology, Jinan 250000, China (e-mail: l.liu@sdu.edu.cn).

Chunbo Luo is with the Department of Computer Science, University of Exeter, EX4 4QF Exeter, U.K. (e-mail: c.luo@exeter.ac.uk).

Digital Object Identifier 10.1109/LRA.2025.3641039

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

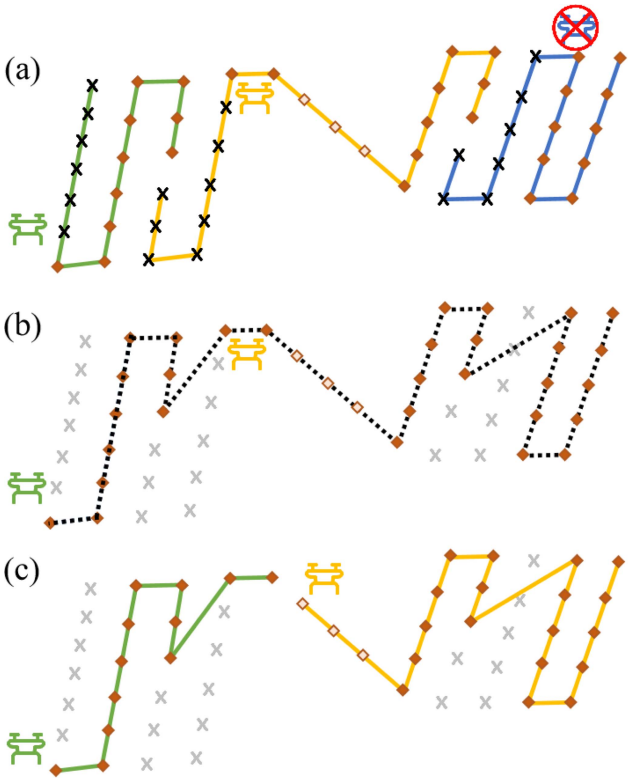


Fig. 2. Replanning Scheme: (a) *Multi-UAV Planning:* GRIT-M assigns one route to each UAV (green, yellow, and blue). Visited waypoints are marked with "x". Unvisited waypoints within regions are shown as solid brown dots, while transitional waypoints between regions are represented by hollow brown dots. The blue UAV is detected as lost. (b) *Concatenation of Unvisited Waypoints:* Remaining unvisited waypoints are connected to form new route segments for task replanning. (c) *Reassignment of Routes:* Updated green and yellow routes are generated, and new missions are allocated to the remaining UAVs.

of the GRIT algorithm [9], GRIT-M redistributes uncovered waypoints from failed UAVs among remaining agents while maintaining region coverage coherence, as shown in Fig. 2. The key contributions of this paper are:

1) A composite transition cost function that optimizes inter-region movements for both energy efficiency and coverage continuity.
2) A region-priority greedy repair mechanism that minimizes transition costs during replanning.
3) A region-preserving Tabu search that ensures area coverage coherence while balancing workload.

Simulation results demonstrate that GRIT-M achieves a 92% replanning success rate within 10 seconds, reduces region transitions by 27% compared to baselines, and maintains workload balance with a coefficient of variation below 7%. The algorithm's computational efficiency (average repair time of 1.4 seconds) confirms its suitability for real-time applications where rapid failure recovery is crucial.

II. Related Work

A. Multi-Region Coverage Path Planning

Coverage path planning (CPP) methods are generally categorized into decomposition-based and pattern-based approaches [6]. Decomposition techniques divide the coverage area into simpler sub-regions using methods like trapezoidal [11] or grid-based methods [12]. Pattern-based approaches apply geometric patterns directly to the area, such as the back-and-forth (B&F) pattern [13].

While most CPP research focuses on single contiguous regions, many applications require covering multiple separated areas. This introduces the challenge of determining the optimal visiting sequence for these regions, combining the traveling salesman problem with CPP [14].

To address this, Xie and Chen [15] formulated a combined problem, generating back-and-forth patterns for each region while optimizing the visiting order. Yu et al. [16] used ant colony optimization for multi-UAV systems to optimize both visit order and task allocation. Khanam et al. [17] proposed methods for inter-region and intra-region path planning that consider precedence constraints. A primary challenge in multi-area CPP is computational complexity. Luna et al. [10] presented a heuristic that achieves near-optimal coverage while prioritizing planning speed, making it practical for real-time applications.

B. Fault-Tolerant Coverage and Online Replanning

UAV operations are susceptible to failures, necessitating online replanning capabilities for mission success [18]. These methods require both failure detection and efficient algorithms to redistribute uncompleted tasks.

Fault-tolerant approaches in UAV systems have been explored in several works [19], focusing mainly on formation control [20] and task reassignment [21]. However, these methods typically do not address the specific challenges of CPP, where spatial continuity and complete coverage are critical.

Clark et al. [9] introduced the GRIT (Greedy Repair Initializes Tabu search) algorithm for online path repair in multi-UAV missions. It uses a greedy heuristic to redistribute a failed UAV's waypoints and refines the new routes with Tabu search, achieving replanning times 10-50 times faster than traditional planners. In a different context, Hong et al. [22] proposed a Receding Horizon Task Assignment algorithm for CPP that handles dynamic changes, such as the addition of UAVs or regions, rather than failures.

The key challenge in fault-tolerant replanning is balancing computational speed with solution quality. Approaches like GRIT [9], which leverage the structure of initial plans to guide rapid replanning, offer practical solutions for real-world deployments. Our work builds on these foundations by integrating efficient multi-area planning with rapid online replanning to create a unified framework for resilient, multi-region missions.

III. Preliminaries and Problem Formulation

A. Notation and Coverage Model

We consider a team of N homogeneous rotary–wing UAVs $U = \{u_1, \ldots, u_N\}$ with planar start positions $\mathbf{p}_i \in \mathbb{R}^2$, where homogeneity refers to identical kinematics and sensor footprints, while battery capacities are tracked individually. The mission comprises M disjoint, convex polygonal areas $R = \{R_1, \ldots, R_M\}$, with $R_j = \mathrm{poly}(\mathbf{r}_{j1}, \ldots, \mathbf{r}_{jn_j})$.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Rasterisation: We adopt the *sub-polygon sweep selection* module of [23] to precompute the raster for each region R_j (each region is one sub-polygon). The procedure is: (i) *Longest-edges enumeration:* take the orientations of several longest boundary edges of R_j as candidate sweep azimuths; (ii) *Four start–end variants:* for each candidate azimuth, instantiate four B&F candidates by starting from either endpoint of the chosen longest edge and traversing forward/back, which yields a unique boundary entry and exit; (iii) *Feasibility check:* build the lane lattice with footprint spacing s, clip lanes to R_j, and keep only candidates whose sweep lines remain inside the polygon; (iv) *Local selection (lightweight):* from the feasible set, pick the variant with the smallest intra-region coverage length (lanes plus connectors); if tied, pick the one whose entry waypoint is closest to the team centroid $\bar{\mathbf{p}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i$. The module designates $\mathbf{w}_{j1}$ and $\mathbf{w}_{jk_j}$ as entry/exit and records the chosen sweep azimuth α_j. The resulting ordered waypoint list is

$$W_j = \{\mathbf{w}_{j1}, \ldots, \mathbf{w}_{jk_j}\} \subset \mathbb{R}^2, \qquad k_j = |W_j|. \tag{1}$$

Lane axis and lane direction: For every waypoint $\mathbf{w}_{ji} = (x_{ji}, y_{ji})$, we record its orientation relative to the next waypoint in the sweep sequence as $\theta_{ji} = \text{atan2}(y_{ji+1} - y_{ji}, x_{ji+1} - x_{ji})$. All waypoints within the same region R_j share a common *lane-axis orientation*, which is the angle of the normal to the sweep lanes, defined as

$$\varphi_j = \text{wrap}_\pi\big(\theta_{j1} - \pi/2\big). \tag{2}$$

where $\text{wrap}_\pi(\cdot)$ wraps an angle to $(-\pi, \pi]$. The corresponding lane-axis unit vector is $\mathbf{a}_j = (\cos\varphi_j, \sin\varphi_j)$.

From this, we define the *principal travel direction* vector $\mathbf{d}_j = (-\sin\varphi_j, \cos\varphi_j)$, which is orthogonal to $\mathbf{a}_j$. The sign $s_j \in \{-1, +1\}$ indicates whether local motion is along $+\mathbf{d}_j$ or $-\mathbf{d}_j$. The entry/exit lane-direction vectors of a region are defined as

$$\mathbf{l}_j^{\text{en}} = s_j^{\text{en}}\mathbf{d}_j, \qquad \mathbf{l}_j^{\text{ex}} = s_j^{\text{ex}}\mathbf{d}_j, \tag{3}$$

which are later required to measure *coverage continuity* between regions.

Global waypoint graph: The union $V = \bigcup_{j=1}^{M} W_j$ is the vertex set of an undirected graph $G(V, E)$. An edge $e = \{\mathbf{v}_a, \mathbf{v}_b\} \in E$ exists iff the straight-line segment between $\mathbf{v}_a$ and $\mathbf{v}_b$ can be flown directly (i.e., it is obstacle-free and feasible under UAV kinematics), with weight $w_e = \|\mathbf{v}_a - \mathbf{v}_b\|_2$. The waypoints in V are generated via non-overlapping back-and-forth sweeps within a set of disjoint mission regions. This structure ensures that the planned coverage is free of redundancy and, combined with the convex region assumption, guarantees static obstacle avoidance for all intra-region segments by design.

B. Initial Planning Objective

In our operational model, all UAVs execute their assigned paths *concurrently* to minimize the overall mission time. The mission requires partitioning the global set of waypoints V into N paths, $P = \{P_1, \ldots, P_N\}$, one for each UAV. The path length for UAV u_i, which traverses the sequence of waypoints in its assigned path P_i, is defined as:

$$\|P_i\| = \sum_{\ell=1}^{|P_i|-1} \|\mathbf{v}_{i\ell} - \mathbf{v}_{i(\ell+1)}\|_2, \tag{4}$$

and must satisfy $\|P_i\| \le B_i^{\max}$. The planning objective is the makespan minimisation $\min\ \max_i \|P_i\|$.

C. Online Re-Planning After Failures

Assume UAV u_k fails at time t. At this moment, the path of each UAV u_i (including the failed one) is split into a completed part P_i^c and an unvisited part P_i^u. The set of all waypoints that still need to be covered is thus $V^u = \bigcup_{i=1}^{N} P_i^u$. Let B_i^{rem} be the remaining battery capacity of each surviving UAV u_i ($i \neq k$).

The online repair problem is to re-partition the set of all unvisited waypoints V^u among the surviving UAVs. This results in a new set of paths $\{P_i^p\}_{i\neq k}$, where P_i^p denotes the new, complete post-repair path assigned to the surviving UAV u_i. The objective is to minimize the new makespan, which is the maximum length of any new path, subject to coverage and energy constraints. The problem is formulated as:

$$\min_{\{P_i^p\}_{i\neq k}} \max_{i\neq k} \|P_i^p\| \tag{5}$$

$$\text{s.t.} \quad \bigcup_{i\neq k} P_i^p \cup \bigcup_{i=1}^{N} P_i^c = V \qquad \text{(Coverage)} \tag{6}$$

$$\|P_i^p\| \le B_i^{\text{rem}}, \ \ \forall i \neq k \qquad \text{(Energy)} \tag{7}$$

D. Assumptions

- All UAVs have identical kinematics and sensor footprints, while battery capacities are tracked individually.
- The B&F lattice fully covers each polygonal region, which may be split among multiple UAVs if it exceeds a single UAV's energy budget.
- Failures are detected instantaneously and broadcast.
- Wind effects are negligible at the planning level.

IV. The GRIT-M Algorithm

We present GRIT-M, which integrates *region awareness* into both initial mission planning and dynamic path repair. The key components are as follows:

1) We define a *composite transition cost* function that captures distance, heading changes, and coverage continuity between regions (Section IV-A).
2) We incorporate this cost into initial *region-aware path planning* that sequences regions and assigns tasks to UAVs while preserving transitions (Section IV-B).
3) We design a two-step *multi-region path repair* approach (Section IV-C) combining a greedy region-priority heuristic with a region-preserving Tabu search.

A. Composite Transition Cost

Given distinct regions R_i and R_j, the three normalised sub-metrics are:

1) Distance penalty: For regions R_i and R_j, let their ordered waypoint lists be $W_i = \{\mathbf{w}_{i1}, \ldots, \mathbf{w}_{ik_i}\}$ and $W_j = \{\mathbf{w}_{j1}, \ldots, \mathbf{w}_{jk_j}\}$. The *transition distance* is defined between the *exit* waypoint of R_i ($\mathbf{w}_{ik_i}$) and the *entry* waypoint of R_j

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

$(\mathbf{w}_{j1})$:

$$d_{\text{trans}}(R_i, R_j) = \|\mathbf{w}_{ik_i} - \mathbf{w}_{j1}\|_2. \tag{8}$$

The normalized distance penalty is then

$$\tilde{T}_{\text{dist}}(R_i, R_j) = \frac{d_{\text{trans}}(R_i, R_j)}{\max\limits_{p \neq q} d_{\text{trans}}(R_p, R_q)}. \tag{9}$$

2) Orientation-mismatch penalty: Let θ_i^{ex} be the exit course from R_i and θ_j^{en} the entry course into R_j. The wrapped heading difference $\Delta\theta_{ij} = |\text{wrap}_\pi(\theta_i^{\text{ex}} - \theta_j^{\text{en}})|$ yields

$$\tilde{T}_{\text{orient}}(R_i, R_j) = \frac{1 - \cos\Delta\theta_{ij}}{2}. \tag{10}$$

(3) Coverage-continuity reward: Coverage coherence is captured by the *exit* lane-direction vector $\mathbf{l}_i^{\text{ex}}$ and the *entry* vector $\mathbf{l}_j^{\text{en}}$ (Section III). Their normalised dot product

$$\tilde{C}_{\text{cont}}(R_i, R_j) = \frac{1 + \langle \mathbf{l}_i^{\text{ex}}, \mathbf{l}_j^{\text{en}} \rangle}{2} \in [0, 1] \tag{11}$$

reflects both the alignment of sweep-lane axes and the forward travel directions across regions. This differs from the heading-mismatch cost, which compares only the UAV's instantaneous exit and entry headings. In our framework as shown in Fig. 3, UAVs always move forward along their designated sweep direction, ensuring that continuity captures lane-level alignment rather than duplicating the heading-based metric. It equals 1 when the sweep lanes extend seamlessly, 0 when they meet head-on, and 0.5 for a 90° cross.

Composite cost:

$$T_{\text{comp}}(R_i, R_j) = \alpha\,\tilde{T}_{\text{dist}} + \beta\,\tilde{T}_{\text{orient}} - \gamma\,\tilde{C}_{\text{cont}}, \tag{12}$$

with $\alpha + \beta + \gamma = 1$ and $\alpha, \beta, \gamma \geq 0$. Since each sub-metric is normalized to the range [0,1] before weighting, the composite cost T_{comp} is consequently bounded within $[-\gamma, 1-\gamma]$. This normalization also means the coefficients do not require retuning for different map sizes or region areas. We use $(\alpha, \beta, \gamma) = (0.6, 0.25, 0.15)$ throughout.

B. Region-Aware Path Planning

Initial path planning partitions the global waypoint set V into N paths to minimize the makespan. The process is detailed in Algorithm 1 and unfolds in two phases: region sequencing and task allocation, guided by the composite transition cost T_{comp}. Key inputs include UAV starting positions $\{\mathbf{p}_i\}$, battery limits $\{B_i^{\max}\}$, per-UAV transition budgets $\{T_i^{\text{budget}}\}$, and the cost weights (α, β, γ). We assume each region's sweep pattern (lane orientation and direction) is a fixed input from a prior rasterization step, which defines the entry/exit waypoints.

1) Region Sequencing: A greedy heuristic determines the region visit order, R_{seq}. To minimize initial travel, the sequence starts at the region R_{start} whose entry point $\mathbf{w}_{j1}$ is closest to the team centroid $\bar{\mathbf{p}}$.

$$R_{\text{start}} = \arg\min_{R_j \in R} \|\bar{\mathbf{p}} - \mathbf{w}_{j1}\|_2 \tag{13}$$

Subsequent regions are appended by iteratively selecting the unvisited region with the minimum T_{comp} from the last region in the sequence.

2) Task Allocation: The algorithm iterates through R_{seq}, assigning each region R_j to a UAV. An assignment to UAV u_i is feasible if its updated path length and cumulative transition cost remain within its operational limits:

$$\|P_i\| + d_i(R_j) + C_{\text{cov}}(R_j) \leq B_i^{\max} \tag{14}$$

$$T_i^{\text{cumul}} + T_{\text{comp}}(R_{\text{prev}}(u_i), R_j) \leq T_i^{\text{budget}} \tag{15}$$

Here, P_i is the UAV's current path, $C_{\text{cov}}(R_j)$ is the internal path length of region R_j, and $R_{\text{prev}}(u_i)$ is the last region assigned to u_i. The transition distance $d_i(R_j)$ is the travel from u_i's last waypoint (or start position $\mathbf{p}_i$ if its path is empty) to R_j's entry waypoint $\mathbf{w}_{j1}$.The term T_i^{budget} is a *user-defined* per-UAV transition budget that caps the cumulative composite transition cost T_i^{cumul} along P_i. Since $T_{\text{comp}} \in [-\gamma, 1-\gamma]$ by (12), a conservative range for this budget is $T_i^{\text{budget}} \in \big[0,\ (1-\gamma)\cdot(M-1)\big]$.

The same best-fit rule is applied iteratively when a region must be split. From the set of feasible candidates, the best-fit UAV is chosen as the one with the currently shortest path length $(\arg\min_{u_i} \|P_i\|)$.

If no single UAV can feasibly cover all of R_j, the region is split. Its ordered waypoint list, W_j, is partitioned into contiguous sub-sequences. These sub-sequences are then greedily assigned to the best available UAVs (i.e., for each sub-sequence, selecting the feasible UAV with the smallest current path length) until the entire region is allocated, ensuring even large areas are covered efficiently.

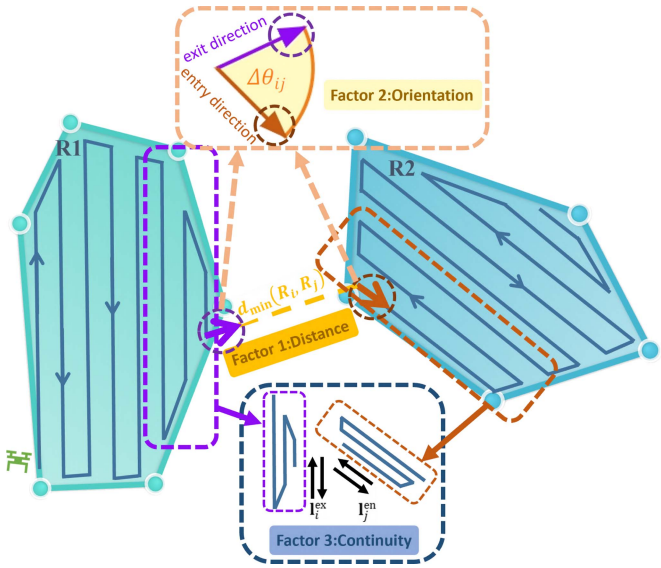


Fig. 3. Illustration of the composite transition cost T_{comp} between two coverage regions. When a UAV moves from region R_1 to R_2, three normalised factors are considered: Factor 1 – Distance ($\tilde{T}_{\text{dist}}$): the Euclidean distance between the exit waypoint of R_1 and the entry waypoint of R_2; Factor 2 – Orientation ($\tilde{T}_{\text{orient}}$): the heading mismatch between the UAV's instantaneous departure direction from R_1 and its arrival direction into R_2; Factor 3 – Continuity ($\tilde{C}_{\text{cont}}$): the alignment of the backandforth sweep patterns of the two regions, expressed as the cosine similarity of their lanedirection vectors. Weighting these three terms yields T_{comp}, which steers both the initial planner and the online repair module toward energetically efficient and coveragecoherent region transitions.

C. Multi-Region Path Repair

Upon UAV failure, the repair module re-plans paths for the surviving UAVs U' to cover all unvisited waypoints V^u (as

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Algorithm 1: Region-Aware Initial Path Planning.

Require: Regions $R = \{R_1, \ldots, R_M\}$, UAVs $U = \{u_1, \ldots, u_N\}$, Initial positions $\{\mathbf{p}_i\}$, Battery limits $\{B_i^{\max}\}$, Parameters (α, β, γ), Transition budgets $\{T_i^{\text{budget}}\}$.
Ensure: Initial UAV paths $\{P_1, \ldots, P_N\}$.

Initialize: $P_i \leftarrow \emptyset$, $\|P_i\| \leftarrow 0$, $T_i^{\text{cumul}} \leftarrow 0$ for $i = 1, \ldots, N$.
Pre-compute entry/exit waypoints and orientations for each region from pre-specified sweep patterns.
Compute cost matrix $T_{\text{comp}}(R_i, R_j)$ for all pairs using (12).
{1. Determine optimal region sequence}
$R_{\text{start}} \leftarrow \arg\min_{R_j \in R} \|\bar{\mathbf{p}} - \mathbf{w}_{j1}\|_2$
$R_{\text{seq}} \leftarrow \text{GreedySequence}(R, R_{\text{start}}, T_{\text{comp}})$
{2. Assign regions to UAVs}
for each region R_j in R_{seq} **do**
  $C_{\text{cov}}(R_j) \leftarrow \sum_{\ell=1}^{|W_j|-1} \|\mathbf{w}_{j\ell} - \mathbf{w}_{j(\ell+1)}\|_2$
  Candidates $\leftarrow \emptyset$
  for each UAV $u_i \in U$ **do**
    Check feasibility using Eq. (14) and (15).
    if feasible **then**
      Candidates $\leftarrow$ Candidates $\cup \{u_i\}$
    end if
  end for
  if Candidates $\neq \emptyset$ **then**
    {Assign whole region to best-fit UAV}
    $u^* \leftarrow \arg\min_{u_i \in \text{Candidates}} \|P_i\|$
    $P_{u^*} \leftarrow \text{Append}(P_{u^*}, W_j)$
    Update $\|P_{u^*}\|$ and $T_{u^*}^{\text{cumul}}$.
  else
    {Split region among multiple UAVs}
    $\text{SplitAndAssign}(R_j, U, \{P_i\}, \{B_i^{\max}\})$
  end if
end for
return $\{P_1, \ldots, P_N\}$

Algorithm 2: Multi-Region Path Repair.

Require: Current paths $\{P_i\}_{i \neq k}$, Unvisited waypoints V^u, UAVs U', Remaining energy $\{B_i^{\text{rem}}\}$, Threshold τ, Max iterations $I_{\max}$
Ensure: Updated paths $\{P_i^*\}_{i \neq k}$

$\mathcal{W} \leftarrow \text{GroupWaypointsByRegion}(V^u)$
for each region-specific waypoint set $W_j^u \in \mathcal{W}$ **do**
  {Partition UAVs based on proximity to region R_j}
  $U_{\text{in}} \leftarrow \{u_i \in U' \mid u_i \text{ already serves region } R_j\}$
  $U_{\text{near}} \leftarrow \emptyset$
  for each $u_i \in U' \setminus U_{\text{in}}$ **do**
    if $\exists\, R_\ell \in \text{seq}(P_i)$ s.t. $T_{\text{comp}}(R_\ell, R_j) < \tau$ **then**
      $U_{\text{near}} \leftarrow U_{\text{near}} \cup \{u_i\}$
    end if
  end for
  $U_{\text{other}} \leftarrow U' \setminus (U_{\text{in}} \cup U_{\text{near}})$
  $P \leftarrow \text{GreedyInsert}(P, W_j^u, U_{\text{in}} \cup U_{\text{near}} \cup U_{\text{other}})$
end for
$P^* \leftarrow P$
Initialize Tabu List $\mathcal{T} \leftarrow \emptyset$
for iter $= 1$ to $I_{\max}$ **do**
  $N(P) \leftarrow \text{GenerateNeighbors}(P)$
  Find best move $m^* \in N(P)$ minimizing $\max_i \|P_i'\|$
  if m^* is not Tabu or meets aspiration criterion **then**
    $P \leftarrow \text{ApplyMove}(P, m^*)$; Add m^* to $\mathcal{T}$
    if $\max_i \|P_i\| < \max_i \|P_i^*\|$ **then**
      $P^* \leftarrow P$
    end if
  end if
end for
return P^*

defined in Section III), subject to remaining energy constraints B_i^{rem}. The process consists of a greedy repair followed by a Tabu Search refinement, detailed in Algorithm 2.

Phase 1. Region-Priority Greedy Repair: Uncovered waypoints are grouped by region and assigned using a *cheapest feasible insertion* heuristic. For each region R_j's waypoints, candidate UAVs are considered in a prioritized order: first, UAVs already in that region (U_{in}); second, those in "nearby" regions (U_{near}); and finally, all others (U_{other}). Nearness is determined by the static pairwise cost $T_{\text{comp}}(R_\ell, R_j) < \tau$, which serves as a proximity heuristic, not a dynamic insertion cost. The path length increase ΔC for inserting a sequence S between waypoints $\mathbf{w}_a$ and $\mathbf{w}_b$ is:

$$\Delta C(P_i, S, a) = \|\mathbf{w}_a - \mathbf{s}_1\| + \sum_{\ell=1}^{m-1} \|\mathbf{s}_\ell - \mathbf{s}_{\ell+1}\| + \|\mathbf{s}_m - \mathbf{w}_b\| - \|\mathbf{w}_a - \mathbf{w}_b\|. \quad (16)$$

An insertion is feasible if $\|P_i\| + \Delta C \leq B_i^{\text{rem}}$. The feasible insertion minimizing ΔC is chosen. Unassignable waypoints indicate partial mission failure.

Phase 2. Region-Preserving Tabu Search: The greedy solution is refined by Tabu Search (TS), a metaheuristic that uses a memory structure (a *tabu list*) to forbid recent moves and escape local optima, with the goal of minimizing the makespan $\max_{i \in U'} \|P_i\|$. The neighborhood is generated by all feasible *relocate moves*, where a single waypoint is moved from one UAV's path to another's. The search is guided by a *region-preserving* constraint, which ensures the total transition cost of the solution does not increase.

To formalize this, let the inter-region sequence for path P_i be $\text{seq}(P_i) = \langle R_{i,1}, \ldots, R_{i,m_i} \rangle$. The total transition cost is defined as:

$$C_{\text{trans}}^{\text{total}}(P) = \sum_{i \in U'} \sum_{\ell=1}^{m_i - 1} T_{\text{comp}}(R_{i,\ell}, R_{i,\ell+1}). \quad (17)$$

A move transforming P to P' is only considered if $C_{\text{trans}}^{\text{total}}(P') \leq C_{\text{trans}}^{\text{total}}(P)$. In each iteration, the best non-tabu move is applied. An aspiration criterion can override tabu status. The search terminates after $I_{\max}$ iterations or upon convergence.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

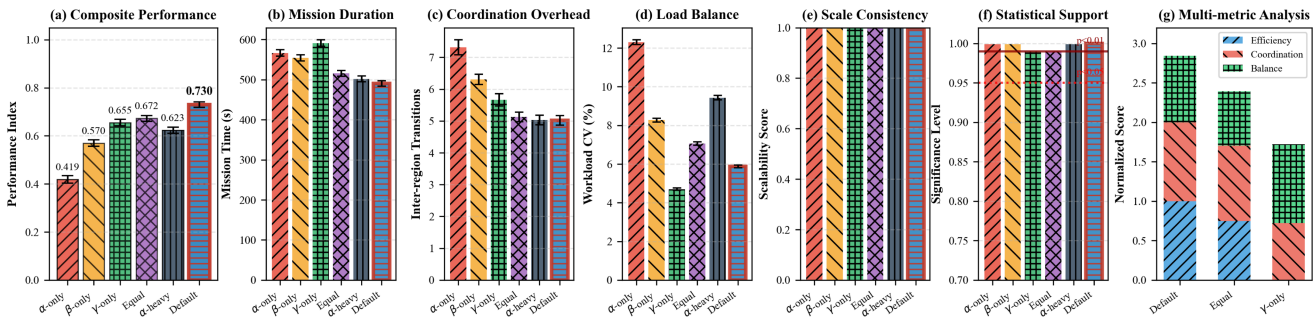


Fig. 4. Parameter sensitivity analysis. Performance across six weighting configurations is evaluated using: (a) composite index, (b) mission time, (c) transitions, (d) workload CV, (e) cross-scenario consistency, (f) significance, and (g) breakdown. The default configuration ($\alpha = 0.6, \beta = 0.25, \gamma = 0.15$) attains the highest composite score (0.730) and shows statistical superiority ($p < 0.01$) while balancing time, transitions, and workload.

D. Computational Complexity and Scalability

Initial planning: Computing the $M \times M$ matrix T_{comp} is $O(M^2)$ because each entry relies on the fixed (exit, entry) waypoints, i.e., $d_{\text{trans}}(R_i, R_j) = \|\mathbf{w}_{ik_i} - \mathbf{w}_{j1}\|_2$. The intra-region path lengths $\{C_{\text{cov}}(R_j)\}$ sum to $O(|V|)$. For each region R_j, feasibility checks across up to N UAVs involve $O(1)$ per UAV (distance from the last waypoint and cumulative transition budget are stored), so assignment is $O(MN)$, plus $O(|V|)$ for computing $\{C_{\text{cov}}\}$. When a region is split, the added work remains linear in its waypoint count.

Repair: Let h be the number of uncovered waypoints. The greedy insertion step examines, in the worst case, $O(h\,|V|)$ insertion positions across receiving routes (substantially reduced in practice by the T_{comp}-based $U_{\text{in}} \cup U_{\text{near}}$ filtering). Tabu Search uses relocate moves; per iteration, up to $O(h\,|V|)$ energy-feasible and region-preserving candidates are evaluated in $O(1)$ using the incremental cost ΔC and local transition checks, yielding $O(I_{\max}\, h\,|V|)$ worst-case time.

V. Simulation Results

In this section, we present a detailed evaluation of the proposed GRIT-M algorithm. We begin by analyzing the impact of the composite transition cost parameters (α, β, γ) on overall mission metrics. We then assess the algorithm's performance in initial planning and online repair scenarios, comparing GRIT-M with three baseline methods. Finally, we discuss scalability results.

A. Simulation Setup and Metrics

Simulations were run in a 3×2 km area with M disjoint regions (0.05–1.32 km^2). We used N homogeneous UAVs (10 m/s speed, 80 m sensor footprint, $\sim$ 25 min flight time). Performance was evaluated on: mission time, region transitions, workload balance (CV, %), computation time, and post-failure time increase (%). Baselines included the IP-based POPCORN [5], the region-agnostic GRIT [9], and a standard Tabu Search (TS).

B. Parameter Sensitivity Analysis

We tested six configurations: α-only $(1.0, 0.0, 0.0)$, β-only $(0.0, 1.0, 0.0)$, γ-only $(0.0, 0.0, 1.0)$, equal weights $(0.33, 0.33, 0.34)$, α-heavy $(0.8, 0.1, 0.1)$, and default $(0.6, 0.25, 0.15)$. Over 2,700 trials across $N \in \{3, 5, 7\}$, $M \in \{4, 6, 8\}$, and spatial distributions, we evaluated mission time, transitions, and CV. A composite index is $\text{PI} = 0.4(1 - \tilde{T}_{\text{time}}) + 0.3(1 - \tilde{T}_{\text{trans}}) + 0.3(1 - \tilde{CV})$, with tildes denoting min-max normalization.

Fig. 4 shows the default configuration achieves the highest PI (0.730), with mission time only 6.5% above α -only (570s) while reducing transitions by 29% and keeping CV below 5.8%. Compared with the extremes, α-only yields fastest time but high imbalance (CV $\sim$12–13%), while γ-only achieves best balance (CV $\sim$4–5%) at a 31% time penalty. The default strikes the best trade-off, statistically superior ($p < 0.01$, two-sided t-test), balancing distance ($\alpha = 0.6$), orientation ($\beta = 0.25$), and continuity ($\gamma = 0.15$).

C. Initial Planning Performance

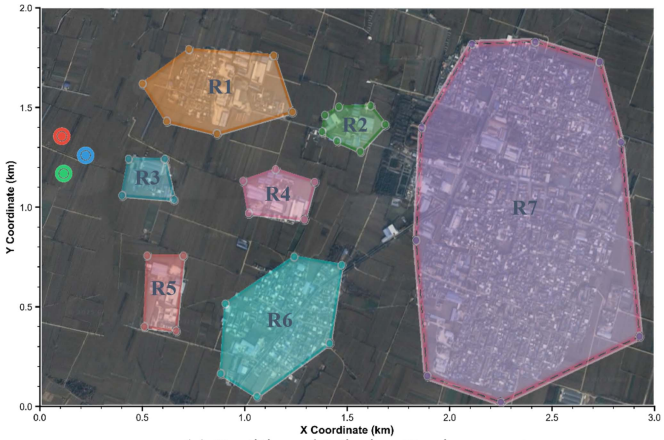
1) Coverage Path Examples: Fig. 5 shows a nominal scenario with $N = 3$ UAVs covering $M = 7$ regions from a satellite image. GRIT-M assigns entire regions to single UAVs when possible, splitting only the largest due to battery limits, to minimize transitions.

2) Comparison With Baselines: Fig. 6 compares metrics for $N = 3$, $M = 7$ over 30 runs. GRIT-M's mission time is within 5% of POPCORN's optimum but with much faster planning (< 2s). It reduces transitions by 23% vs. Original GRIT, with better balance (CV $\sim$5–7%) than TS.

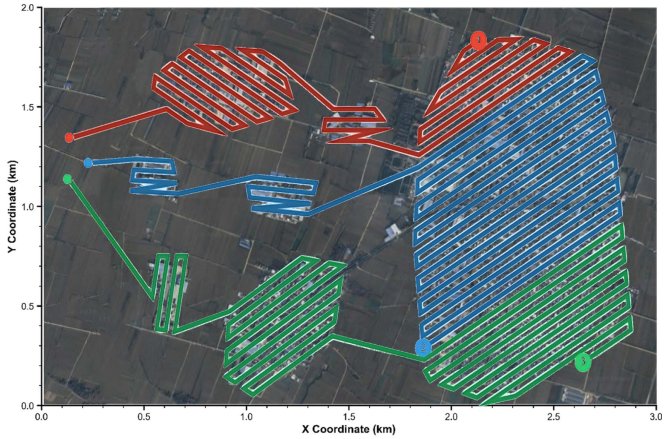
D. Online Repair Under UAV Failures

1) Single Failure Scenario: Fig. 7 evaluates repair vs. uncovered waypoints. GRIT-M achieves $> 90\%$ success for up to 250 waypoints, competitive CV (5.8–7.6%), 21–35% transition reduction, and repair times of 1.1–11.2s. It outperforms baselines in transition minimization and efficiency, though POPCORN excels in balance for large scales.

2) Multiple Failures and Scalability: Fig. 8 shows near-linear scaling, consistent with its $O(I_{\max} h|V|)$ worst-case complexity, with $<$ 10s repair for up to 560 variables. In multi-failure cases, success rates are 92% (single) to 58% (sequential), 15–30% above baselines, with times 1.4–6.3 s. The radar chart highlights GRIT-M's balanced performance across



(a) Partitioned Mission Environment



(b) Initial Multi-UAV Path Plan

Fig. 5. (a) Geometric division of the coverage area in the satellite image of the task environment, with the initial positions of UAV1, UAV2, and UAV3 marked in red, blue, and green, respectively. Seven non-overlapping regions are assigned for cooperative coverage. (b) Planned coverage paths of the three UAVs within their designated regions, with Region R7 being collaboratively covered by all three UAVs.

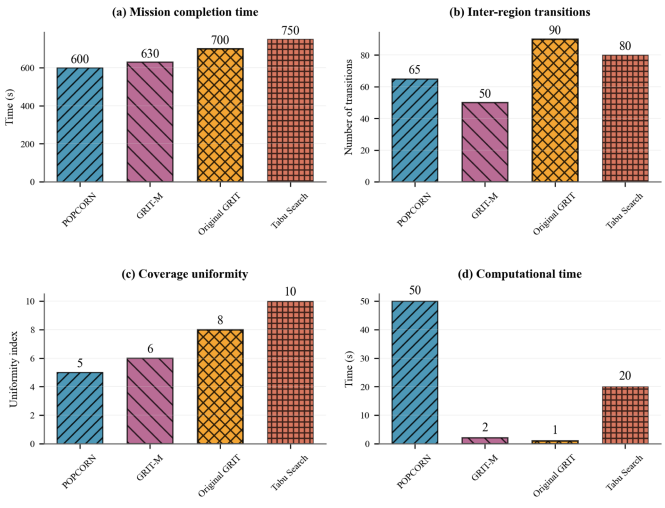


Fig. 6. Performance comparison of four algorithms in initial planning (3 UAVs, 7 regions, avg. 30 Monte Carlo runs): (a) Total task duration; (b) Cross-region flights; (c) CV of path length (load balancing); (d) Offline planning time. GRIT-M reduces region transitions by 23% vs. baseline, with only 5% longer duration, planning under 2 s, balancing efficiency and effectiveness.

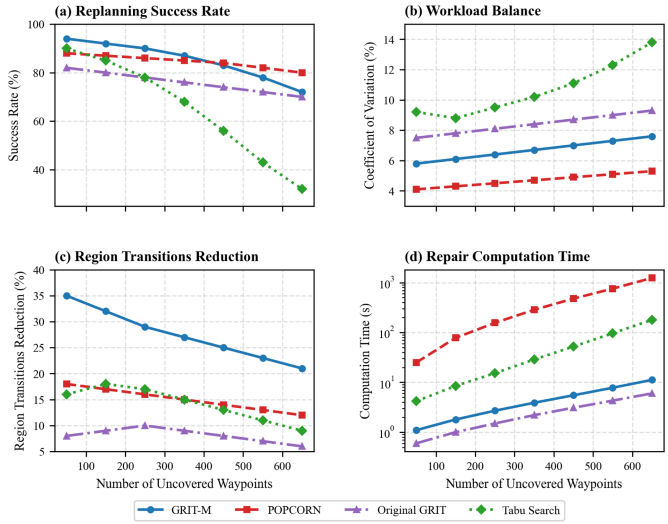


Fig. 7. Repair performance vs. number of uncovered waypoints: (a) replanning success rate, (b) workload balance (CV), (c) reduction in inter-region transitions, and (d) computation time. GRIT-M markedly reduces inter-region transitions while sustaining high replanning success, good balance, and low computation overhead.

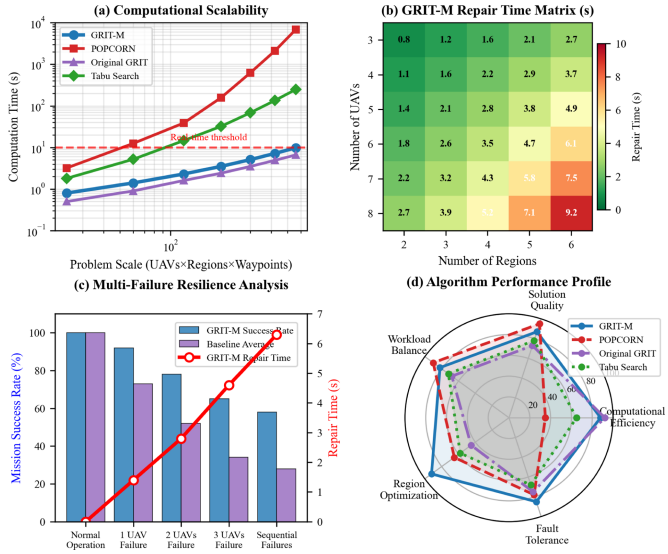


Fig. 8. GRIT-M scalability and resilience: (a) near-linear growth in complexity vs. exponential baselines; (b) repair times remain under 10 s in realistic setups; (c) 15–30% higher success rates under multiple failures; (d) radar chart highlights balanced performance across five dimensions.

efficiency, quality, balance, optimization, and tolerance (scores 82–92).

E. Extended Empirical Analysis

We further assessed GRIT-M's scalability and robustness.

Scalability was tested with 9–12 UAVs in a 15-region scenario (Table I). Adding agents effectively reduced mission time (e.g., from 1125 s to 834 s) while repair time scaled efficiently (1.6 s to 2.1 s) and success rates held above 91%.

Robustness was confirmed against operational constraints (Fig. 9): (a) *Battery variations* ($\pm 25\%$ capacity) altered mission

TABLE I
PERFORMANCE SCALABILITY WITH LARGER UAV TEAMS

Number of UAVs	Mission Time (s)	Repair Time (s)	Success Rate (%)	Workload CV (%)
9	1125 ± 42	1.6 ± 0.4	92.5	6.8
10	1018 ± 38	1.7 ± 0.5	92.1	6.5
11	923 ± 35	1.9 ± 0.5	91.8	6.3
12	834 ± 31	2.1 ± 0.6	91.3	6.1

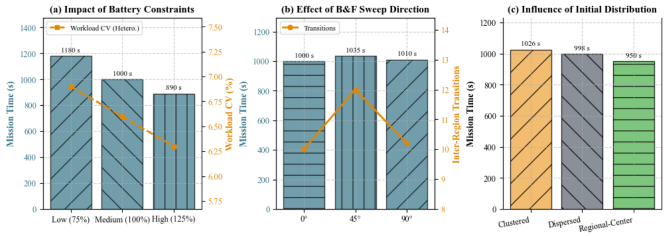


Fig. 9. Robustness analysis of GRIT-M. (a) Impact of absolute battery capacity (Low: 75%, Medium: 100%, High: 125%) and heterogeneity (variance of capacity) on mission time. (b) Effect of pre-defined B&F sweep direction on mission time and inter-region transitions. (c) Influence of initial spatial distribution (Clustered, Dispersed, Regional-Center) on overall performance.

time by $+18\%/-11\%$, and the system handled heterogeneity well; (b) *Sweep patterns* (varied azimuths) caused $< 3.5\%$ time variance due to our cost function's adaptability; and (c) *Initial UAV distributions*, where our heuristic compensated for a clustered start (8% penalty vs. ideal) and a dispersed start (about 5% penalty), confirming deployment flexibility.

VI. CONCLUSION

This letter introduced GRIT-M, a novel algorithm designed for multi-UAV coverage path planning across separated regions, incorporating real-time path reconfiguration in response to UAV failures. The key contributions include a composite transition cost function for efficient inter-region movements, a region-priority repair mechanism for dynamic task reassignment, and a region-preserving Tabu search to maintain coverage continuity and workload balance. The extensive simulation campaign validated the effectiveness, scalability, and robustness of GRIT-M. The algorithm consistently reduced inter-region flights by up to 27% and maintained workload balance (CV $< 7\%$) against leading baselines. Notably, it demonstrated excellent scalability to larger UAV teams and robustness against diverse operational constraints, including varying battery capacities, pre-defined sweep patterns, and different deployment configurations. With average repair times remaining in the low single-digit seconds even in complex scenarios, GRIT-M is proven to be a computationally efficient and resilient solution for real-time applications where swift adaptation to failures is critical. Future research will focus on enhancing GRIT-M's adaptability to dynamic obstacles, integrating heterogeneous UAVs, and exploring distributed implementations for improved system resilience.

REFERENCES

[1] Y. Liu, Q. Wang, H. Hu, and Y. He, "A novel real-time moving target tracking and path planning system for a quadrotor UAV in unknown unstructured outdoor scenes," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 11, pp. 2362–2372, Nov. 2019.
[2] L. Merino, F. Caballero, J. R. Martıńez-De-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *J. Intell. Robot. Syst.*, vol. 65, no. 1, pp. 533–548, 2012.
[3] F. Nex and F. Remondino, "UAV for 3D mapping applications: A review," *Appl. Geomatics*, vol. 6, no. 1, pp. 1–15, 2014.
[4] J. Primicerio et al., "A flexible unmanned aerial vehicle for precision agriculture," *Precis. Agriculture*, vol. 13, no. 4, pp. 517–523, 2012.
[5] K. Shah, G. Ballard, A. Schmidt, and M. Schwager, "Multidrone aerial surveys of penguin colonies in antarctica," *Sci. Robot.*, vol. 5, no. 47, 2020, Art. no. eabc3000.
[6] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robot. Auton. Syst.*, vol. 61, no. 12, pp. 1258–1276, 2013.
[7] G. S. C. Avellar et al., "Multi-UAV routing for area coverage and remote sensing with minimum time," *Sensors*, vol. 15, no. 11, pp. 27783–27803, 2015.
[8] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.
[9] J. Clark, K. Shah, and M. Schwager, "Online path repair: Adapting to robot failures in multi-robot aerial surveys," *IEEE Robot. Automat. Lett.*, vol. 9, no. 3, pp. 2319–2326, Mar. 2024.
[10] M. A. Luna et al., "A multi-UAV system for coverage path planning applications with in-flight re-planning capabilities," *J. Field Robot.*, vol. 41, no. 5, pp. 1480–1497, 2024.
[11] C. Becker, H. Gonzalez-Banos, J. C. Latombe, and C. Tomasi, "An intelligent observer," in *Proc. 4th Int. Symp. Exp. Robot.*, 1997, pp. 151–160.
[12] T. Oksanen and A. Visala, "Coverage path planning algorithms for agricultural field machines," *J. Field Robot.*, vol. 26, no. 8, pp. 651–668, 2009.
[13] J. J. Acevedo, B. C. Arrue, I. Maza, and A. Ollero, "A decentralized algorithm for area surveillance missions using a team of aerial robots with different sensing capabilities," in *Proc. IEEE Int. Conf. Robot. Automat.*, Hong Kong, SAR, China, 2014, pp. 4735–4740.
[14] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE Trans. Robot.*, vol. 25, no. 4, pp. 912–926, Aug. 2009.
[15] J. Xie and J. Chen, "Multiregional coverage path planning for multiple energy constrained UAVs," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 17366–17381, Oct. 2022.
[16] X. Yu et al., "Balanced multi-region coverage path planning for unmanned aerial vehicles," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2020, pp. 3499–3506.
[17] Z. Khanam, S. Saha, S. Ehsan, R. Stolkin, and K. McDonald-Maier, "Coverage path planning techniques for inspection of disjoint regions with precedence provision," *IEEE Access*, vol. 9, pp. 5412–5427, 2021.
[18] S. Mayer, L. Lischke, and P. Woźniak, "Drones for search and rescue," in *Proc. 21st Int. Conf. Hum.- Comput. Interact. Mobile Devices Serv.*, 2019, pp. 1–11.
[19] G. K. Fourlas and G. C. Karras, "A survey on fault diagnosis and fault-tolerant control methods for unmanned aerial vehicles," *Machines*, vol. 9, no. 9, 2021, Art. no. 197.
[20] A. Tahir, J. Böling, M.-H. Haghbayan, and J. Plosila, "Development of a fault-tolerant control system for a swarm of drones," in *Proc. Int. Symp. ELMAR*, 2020, pp. 79–82.
[21] H. Wang, M. Chen, and P. Fu, "A distributed fault-tolerant mechanism for mission-oriented unmanned aerial vehicle swarms," *Int. J. Commun. Syst.*, vol. 34, no. 8, 2021, Art. no. e4789.
[22] Y. Hong, S. Jung, S. Kim, and J. Cha, "Extensions of receding horizon task assignment for area coverage in dynamic environments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 3, pp. 3114–3128, Jun. 2023.
[23] D. Datsko, F. Nekovar, R. Penicka, and M. Saska, "Energy-aware multi-UAV coverage mission planning with optimal speed of flight," *IEEE Robot. Automat. Lett.*, vol. 9, no. 3, pp. 2893–2900, Mar. 2024.