This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2026.3689386

Terrain-Coupled Hierarchical Optimization for Multi-Robot Deployment with a Global Reachability-Cost Atlas

Tianwei Niu[1], Shengshan Ma[1], Runjiao Bao[1], Lin Zhang[1], Haoyu Yuan[1], Liang Wang[2], and Shoukun Wang[1]*

***Abstract*—Autonomous multi-robot deployment (MRD) aims to determine feasible and task-effective deployment poses for robots to accomplish cooperative missions. When applied to unstructured outdoor environments, MRD must contend with irregular terrain geometry, uneven contact surfaces, and terrain-dependent traversability, challenging purely geometric MRD methods. This work proposes a hierarchical MRD framework that unifies terrain-coupled feasibility modeling and execution-aware optimization into a coherent deployment process. An upper-stage nonlinear programming, constrained by a signed-distance field derived from a multi-layer terrain assessment map (MTAM), enforces strict geometric deployment feasibility and yields high-quality candidate fleet configurations. A lower-stage multi-objective evolutionary optimizer then refines fleet poses using a global reachability-cost atlas (GRCA) that precomputes reachability and minimal traversal cost of each robot to every map cell, enabling constant-time candidate evaluation. Terrain coupling is achieved by embedding terrain-risk layers of the MTAM into the optimization objectives, guiding pose refinement toward stable and traversable regions. Extensive simulations and field trials on vibroseis fleet demonstrate system-level integration within a seismic source excitation pipeline, achieving task-level geometric accuracy, stable terrain contact at deployment poses, and real-time onboard optimization with an average latency below 10 s.**

***Index Terms*—Multi-robot deployment, hierarchical optimization, unstructured outdoor environment, reachability analysis, cooperative multi-robot systems, unstructured field robotics.**

I. Introduction

In large-scale field operations such as seismic exploration, precision agriculture, and disaster response, multi-robot often need to autonomously deploy to prescribed locations before engaging in coordinated operations [1]–[4]. Fig. 1(a) illustrates a representative example from our target application, where a fleet of vibroseis trucks autonomously navigates to designated deployment sites and performs seismic source excitation to collect seismic wave signals for oil exploration. We refer to this deployment problem as multi-robot deployment (MRD):

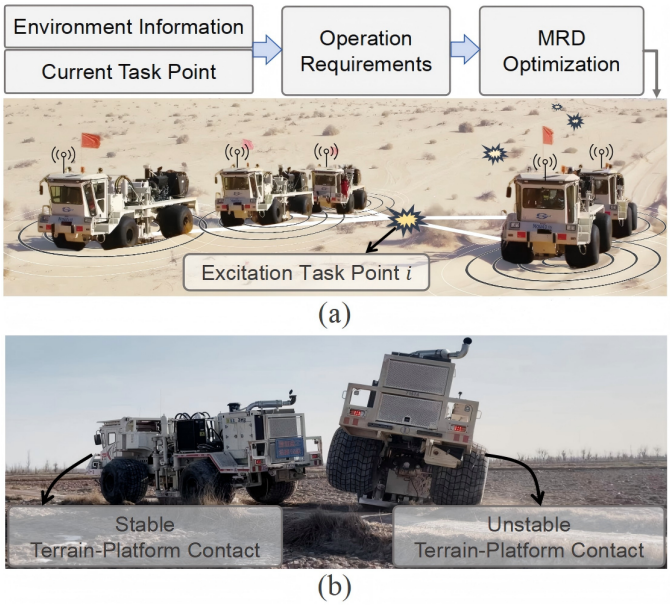


Fig. 1. (a) Fleet of vibroseis trucks executing synchronized source excitation tasks in an unstructured environment. (b) Deployment positions and headings strongly influence the fleet's stability and operational accuracy.

given real-time sensor information and the current task point, to determine feasible, coordinated poses for all robots so that subsequent operation can be carried out safely, accurately, and efficiently.

MRD in unstructured outdoor environments (UOE-MRD) must satisfy geometric configuration requirements, such as inter-robot spacing and fleet-centroid accuracy, while confronting operational challenges posed by complex environments. Terrain irregularities and uneven contact surfaces make the feasible configuration space highly nonconvex and discontinuous. Moreover, the UOE-MRD must ensure that each robot can reach its assigned pose from its current position, a condition referred to as reachability, while jointly minimizing the fleet's aggregate traversal cost. Furthermore, terrain–platform contact quality on rough terrain couples multiple robots through competition for scarce flat support, making positions and headings jointly optimized at the fleet level. The influence of different deployment positions and headings on the quality of terrain–platform contact is illustrated in Fig. 1(b). Even at the same location, different headings can lead to significant variations in contact quality. When coordinating the fleet under geometric constraints, this sensitivity further increases the difficulty of achieving feasible deployments.

This work was supported in part by the National Natural Science Foundation of China under Grants 62473044 and 62503300.

* Corresponding author: Shoukun Wang

[1]Tianwei Niu, Shengshan Ma, Runjiao Bao, Lin Zhang, Haoyu Yuan, and Shoukun Wang are with the School of Automation, Beijing Institute of Technology, Beijing 100081, China (e-mail: ntwbit@bit.edu.cn, lemontree0820@bit.edu.cn, 3120230765@bit.edu.cn, bit.zhanglin@bit.edu.cn, 3120230769@bit.edu.cn, bitwsk@bit.edu.cn).

[2]Liang Wang is with the School of Automation and Software Engineering, Shanxi University, Taiyuan 030031, China (e-mail: yanqianyiliang@sxu.edu.cn).

Prior MRD studies [5]–[7] have largely focused on purely geometric deployment formulations, where the fleet configuration is evaluated mainly by whether spatial constraints are satisfied. More importantly, most existing methods [8], [9] defer the evaluation of reachability and traversal cost to the subsequent task allocation stage, rather than embedding these factors directly into the MRD optimization itself, leading to suboptimal and sometimes infeasible fleet-level solutions.

To address this gap, the main novelty of this work is to embed both terrain–platform contact quality and execution awareness (reachability and traversal cost) directly into the UOE-MRD optimization. Table I summarizes these capability differences against the closest pipelines. To the best of our knowledge, few studies have provided a unified UOE-MRD framework with this terrain-coupled capability, even though these factors are ubiquitous in practice.

TABLE I
FEATURE-LEVEL COMPARISON TO CLARIFY THE INCREMENTAL CONTRIBUTIONS FOR UOE-MRD.

Key aspect	Purely geometric MRD [5]–[7]	Reachability and traversal cost deferred [8], [9]	**Ours**
Fleet geometric configuration	✓	✗	✓
Collision-free placement	✓	✗	✓
Terrain constraints	✗	✗	✓
Terrain–platform contact	✗	✗	✓
Deployment reachability	✗	✓	✓
Traversal cost	✗	✓	✓

Therefore, this work introduces a terrain-coupled, hierarchical optimization framework that unifies geometric feasibility and execution awareness for UOE-MRD. In the upper stage, an interior-point optimization constrained by the signed-distance field (SDF) over fleet positions enforces strict geometric feasibility and produces high-quality candidate configurations. Building on these configurations, the lower stage applies a multi-objective evolutionary optimization (MOEO) that extends the search to full fleet poses and jointly improves terrain–platform contact quality, fleet-level reachability, and aggregate traversal cost.

The contributions of this work are as follows:

1) **Hierarchical optimization for UOE-MRD**. The upper-stage interior-point optimization, constrained by SDF, ensures strict geometric feasibility, providing high-quality seeds for a lower-stage MOEO that refines the solution toward a Pareto front over terrain-coupled performance objectives.
2) **Terrain-Coupled Formulation for Multi-Robot Pose Optimization.** UOE-MRD is formalized as a terrain-coupled pose optimization problem that jointly accounts for terrain–platform contact, reachability, and traversal cost, bridging classical MRD objectives and field-relevant deployment requirements.
3) **Global reachability-cost atlas (GRCA)**. A GRCA is proposed to encode, for each robot, reachability and minimal traversal cost to all map cells, enabling an $O(1)$ evaluation within MOEO and integrating execution-aware reasoning without repeated graph searches.
4) **Field deployment on heavy platforms**. Beyond extensive simulation, a fully integrated pipeline is demonstrated on heavy vibroseis trucks in unstructured environments, showing feasibility, stability, and efficiency under real operational constraints.

The remainder of this paper is organized as follows. Section II reviews related work. Section III formulates the UOE-MRD problem and constructs the multi-layer terrain assessment map. Section IV presents the proposed hierarchical UOE-MRD optimization framework, including the SDF-constrained nonlinear programming and the terrain-coupled MOEO. Section V provides simulation results, ablation analyses, and scalability evaluations. Section VI provides results of real-world deployments on vibroseis fleet in unstructured terrain. Section VII concludes the paper and discusses future extensions.

II. Related Work

A. Geometric Deployment and Formation-Based Coordination

Existing MRD methods draw heavily on geometric coverage and formation control. Centroidal Voronoi tessellation-based deployments [7], [10], [11] and Lloyd-type iterations [12], [13] optimize robot positions for sensing and coverage over planar domains. For instance, Kim et al. [11] incorporated heterogeneous robot speeds into Voronoi-based coverage to achieve time-efficient area partitioning and deployment. The resulting control law minimizes a time-based coverage cost, yielding temporally optimal rather than spatially uniform configurations. Formation control [14]–[17], in turn, regulates inter-agent distances and angles to achieve prescribed shapes, with obstacle avoidance handled heuristically [16] or through reactive layers [17]. For example, Zhang et al. [5] addressed pattern formation in obstacle environments by jointly optimizing goal assignments through convex quadratic programming and iteratively planning collision-free paths for each robot.

These approaches provide scalable geometry and assignment primitives but typically assume simplified terrain and kinematics, evaluate feasibility over discrete grids, and optimize positions without explicitly modeling terrain-pose coupling or execution costs. Recent formation-planning pipelines [6], [18]–[20] that iterate between optimal assignment and collision-free motion improve practicality, yet still frame deployment largely as a geometry-first problem and defer reachability analysis and traversal cost evaluation to a later stage.

B. Multi-Objective Evolutionary Optimization for Deployment and Siting

MOEOs such as the non-dominated sorting genetic algorithm II (NSGA-II) [21]–[24] and the multi-objective evolutionary algorithm based on decomposition (MOEA/D) [25], [26] are widely used for multi-robot deployment [22], [26], sensor deployment [21], [25], and facility-location variants [23], [24] because they maintain diverse Pareto sets under conflicting objectives. Standard constraint-handling schemes, such as Deb's constraint-domination principle [27], are widely adopted for their simplicity and effectiveness. For example, Wang et al. [24] formulated a multi-objective competitive

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2026.3689386

location problem incorporating cooperative coverage, where facilities aim to maximize collective attractiveness while minimizing fixed and distance costs. Yakici et al. [28] addressed a two-level heterogeneous sensor-deployment problem to enhance multi-type coverage performance. Using NSGA-II, they generated diverse Pareto-optimal layouts efficiently, highlighting the potential of evolutionary multi-objective optimization for large-scale deployment tasks with spatial and heterogeneity constraints.

Existing multi-robot exploration, trajectory planning, and graph-based path-planning methods [4], [29], [30] provide important tools for reachability reasoning and collision-free motion generation. However, they are not designed to optimize deployment poses under terrain–platform contact quality, fleet-level geometric constraints, and execution cost simultaneously. Consequently, directly embedding such path-planning or exploration routines into an evolutionary deployment optimizer would require repeated reachability or path-cost evaluations, which can become computationally expensive at scale.

III. Preliminaries

A. Problem Formulation

We address the general problem of pose optimization for UOE-MRD, which exhibits common objectives and constraints across diverse field applications. Building on this formulation, we implement the framework on a fleet of vibroseis trucks performing seismic excitation for oil exploration in unstructured environments.

1) State Variables: The fleet comprising N robots is indexed by

$$\mathcal{R} \triangleq \{1, 2, \ldots, N\}. \tag{1}$$

The stacked state variables are

$$\Xi \triangleq \{\boldsymbol{\xi}_i \,|\, i \in \mathcal{R}\}, \qquad \boldsymbol{\xi}_i^\top \triangleq [x_i \; y_i \; \theta_i] \in \mathbb{R}^3, \tag{2}$$

where $\mathbf{p}_i \triangleq (x_i, y_i)$ denotes the planar position of robot i and θ_i denotes its heading.

2) Objectives: The optimization objectives determine whether the final solution meets the operational intent of UOE-MRD, and Fig. 2 illustrates these objectives schematically.

***Task-Centered Compactness* O1**: Minimize the average squared distance between the robot deployment positions and the task point $\mathbf{s}_\mathrm{t}$. A smaller value encourages a compact fleet configuration around the task point, thereby improving the spatial accuracy of vibroseis operations. **O1** is defined as:

$$\min_{\{\mathbf{p}_i\}_{i=1}^N} \; \mathcal{J}_\mathrm{tcc}(\Xi_{\setminus\theta}) = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{p}_i - \mathbf{s}_\mathrm{t}\|_2^2, \tag{3}$$

where $\mathbf{s}_\mathrm{t}$ denotes a task point. $\Xi_{\setminus\theta} \triangleq \{\mathbf{p}_i \mid i \in \mathcal{R}\}$ is the set obtained by removing the heading components θ from Ξ, where $\mathbf{p}_i = [x_i, y_i]^\top$.

***Terrain–Platform Contact* O2**: Maximize terrain–platform contact quality at the deployed poses, thereby improving energy transmission during excitation and enhancing seismic data quality. More broadly, strong contact conditions promote fleet stability in cooperative tasks over heterogeneous off-road terrain. **O2** is formulated as:

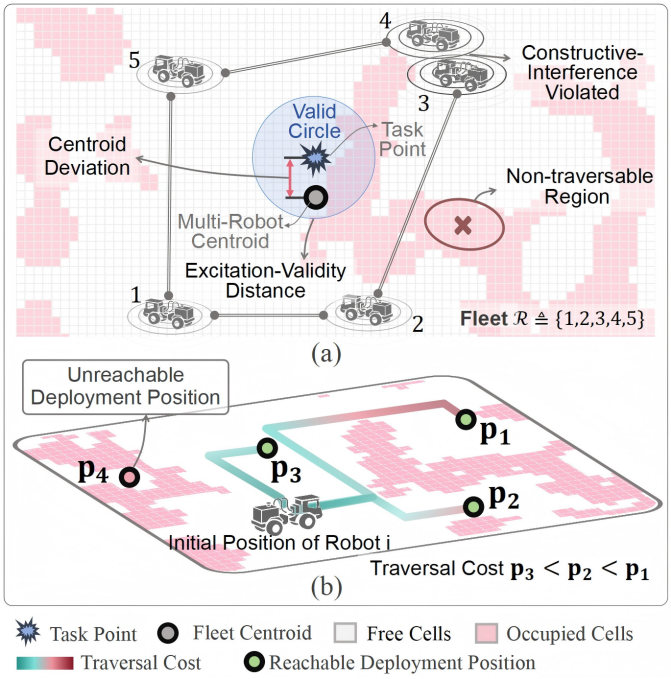


Fig. 2. Illustration of the problem formulation. (a) Geometric objectives and constraints. (b) Reachability and traversal cost evaluation for a robot over four candidate deployment positions ($\mathbf{p}_1$–$\mathbf{p}_4$). Although $\mathbf{p}_4$ lies in a collision-free region, it is dynamically unreachable from the robot's current position. In UOE-MRD, reachability analysis is essential for distinguishing geometrically feasible but dynamically infeasible configurations.

$$\mathcal{J}_\mathrm{tpc} \triangleq \frac{1}{N}\sum_{i=1}^{N} \phi_c(\boldsymbol{\xi}_i), \tag{4}$$

with

$$\phi_c(\boldsymbol{\xi}_i) \triangleq \frac{1}{|\Lambda(\boldsymbol{\xi}_i)|}\sum_{\mathbf{p}\in\Lambda(\boldsymbol{\xi}_i)} \exp\Big(-w_\sigma\,\phi_\sigma(\mathbf{p}) - w_\mathrm{r}\,\phi_\mathrm{r}(\mathbf{p}) - w_\lambda\,\phi_\lambda(\mathbf{p})\Big), \tag{5}$$

where $\Lambda(\boldsymbol{\xi}_i) \subseteq \Omega$ is a query function returning the set of grid cells covered by robot i, and $|\Lambda(\boldsymbol{\xi}_i)|$ denotes its cardinality. $\phi_\sigma(\mathbf{p})$, $\phi_\mathrm{r}(\mathbf{p})$, and $\phi_\lambda(\mathbf{p})$ denote the slope, roughness, and sparsity maps, respectively (computed as described in Sec. III-B), and w_σ, w_r, and w_λ are positive weights. Since the footprint query $\Lambda(\boldsymbol{\xi}_i)$ is defined over the full pose ξ_i, the heading θ_i is explicitly coupled to **O2**: allowing the optimizer to favor deployment orientations with better terrain–platform contact.

***Traversal Cost* O3**: Minimize the traversal cost required for the fleet to reach the deployment configuration from the robots' initial positions. For large-scale or long-duration missions, redundant travel increases fuel or electricity consumption and increases maintenance requirements. Explicitly incorporating objective **O3** into the optimization allows the deployment stage to consider both task assignment and access-

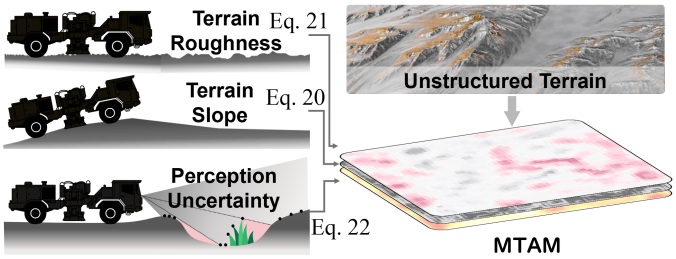


Fig. 3. Construction of the MTAM from slope, roughness, and perception uncertainty derived from unstructured terrain.

path cost, enabling execution-aware optimization. **O3** is defined as:

$$\mathcal{J}_{\text{tc}} \triangleq \frac{1}{N}\sum_{i=1}^{N}\alpha_i\, C_i(\mathbf{o}_i,\mathbf{p}_i) \quad \text{s.t. } \mathbf{p}_i \in \Xi_{\setminus\theta},\ \mathbf{o}_i \in \mathcal{I}, \tag{6}$$

where $C_i(\mathbf{o}_i,\mathbf{p}_i) \in [0,+\infty]$ denotes the minimal path cost from robot i's initial position $\mathbf{o}_i$ to $\mathbf{p}_i$ and α_i is a user-defined weighting factor reflecting each robot's task priority. $\mathcal{I}$ represents the set of initial positions of all robots.

3) Constraints: The constraint set specifies the conditions that must be satisfied to ensure deployable solutions, as illustrated in Fig. 2.

Feasible-Location Constraint **C1**: Leveraging the MTAM (see Section III-B), non-traversable regions are explicitly masked so that candidate poses are restricted to the traversable region $\mathcal{C}_{\text{free}}$. **C1** is defined as:

$$\mathbf{p}_i \notin \{\, \mathbf{p}\in\Omega \mid \mathcal{M}_{\text{occ}}(\mathbf{p})=1 \,\}, \qquad i=1,\dots,N, \tag{7}$$

where Ω denotes the workspace and the non-traversable map $\mathcal{M}_{\text{occ}}(\mathbf{p})=1$ indicates an occupied cell.

Excitation-Validity Constraint **C2**: To ensure valid excitation, the fleet centroid must remain within a prescribed radius of the task point $\mathbf{s}_{\text{t}}$. **C2** is defined as:

$$\|\bar{\mathbf{p}}-\mathbf{s}_{\text{t}}\|_2 \le d_{\text{exc}}, \tag{8}$$

where $\bar{\mathbf{p}}=\frac{1}{N}\sum_{i=1}^{N}\mathbf{p}_i$ denotes the fleet centroid and d_{exc} is the allowable deviation distance of $\bar{\mathbf{p}}$ from $\mathbf{s}_{\text{t}}$. The distance $\|\bar{\mathbf{p}}-\mathbf{s}_{\text{t}}\|_2$ is referred to as the centroid deviation.

Constructive-Interference Constraint **C3**: A minimum pairwise distance is enforced to suppress mutual interference among excitation signals generated by the fleet. **C3** is defined as:

$$\|\mathbf{p}_i-\mathbf{p}_j\|_2 \ge d_{\text{con}}, \qquad \forall\, 1\le i<j\le N, \tag{9}$$

where d_{con} is the minimum pairwise separation required to prevent mutual interference between excitation signals.

Reachability Constraint **C4**: Each robot's deployment pose must be reachable from its initial position. This constraint accounts for the traversability limitations arising from cluttered environments and challenging terrain. **C4** is defined by

$$\chi_i(\mathbf{o}_i,\mathbf{p}_i)=1, \qquad \forall i\in\mathcal{R}, \tag{10}$$

where $\chi_i(\mathbf{o}_i,\mathbf{p}_i)\in\{0,1\}$ is a reachability indicator that equals 1 if there exists a path from $\mathbf{o}_i$ to $\mathbf{p}_i$ entirely contained in free space $\mathcal{C}_{\text{free}}$.

4) Unified Optimization Problem: Based on the above definitions, UOE-MRD is summarized as a unified constrained multi-objective optimization problem, in which **O1** and **O3** are minimized, **O2** is maximized, and **C1–C4** must be satisfied. The corresponding objective vector is

$$\mathbf{F}(\Xi)=\frac{1}{N}\Big[\sum_{i=1}^{N}\|\mathbf{p}_i-\mathbf{s}_{\text{t}}\|_2^2, -\sum_{i=1}^{N}\phi_c(\boldsymbol{\xi}_i), \sum_{i=1}^{N}\alpha_i\, C_i(\mathbf{o}_i,\mathbf{p}_i)\Big]^{\top}. \tag{11}$$

The constraints g_k are given by:

$$g_k(\Xi)\le 0,\ \ g_k(\Xi)\in\mathcal{G}(\Xi),\ \ k=1,\dots,4, \tag{12}$$

where $\mathcal{G}(\Xi)$ collects all constraint functions, corresponding to the following four categories:

$$\mathcal{G}(\Xi)=\begin{cases} g_1(\Xi)=\delta-\tilde{\psi}(\mathbf{p}_i), & \forall i\in\mathcal{R},\\ g_2(\Xi)=\|\bar{\mathbf{p}}-\mathbf{s}_{\text{t}}\|_2-d_{\text{exc}}, & \\ g_3(\Xi)=d_{\text{con}}-\|\mathbf{p}_i-\mathbf{p}_j\|_2, & \forall i,j\in\mathcal{R},\ i<j,\\ g_4(\Xi)=1-\chi_i(\mathbf{o}_i,\mathbf{p}_i), & \forall i\in\mathcal{R}, \end{cases} \tag{13}$$

where $\tilde{\psi}(\mathbf{p}_i)$ is the bilinearly interpolated SDF value derived from $\mathcal{M}_{\text{occ}}(\mathbf{p})$ to express the discrete mask constraint in a continuous, optimization-friendly form (see Sec. IV-B). $\delta\ge 0$ is the SDF safety margin.

We solve the above unified UOE-MRD problem via a two-stage hierarchical optimization that first ensures strict geometric feasibility and then performs execution-aware multi-objective refinement, as detailed in Sec. IV.

B. Dynamic Multi-Layer Terrain Assessment Map

Conventional map abstractions are insufficient for UOE-MRD. We therefore adopt a multi-layer terrain assessment map (MTAM), as illustrated in Fig. 3. The non-traversable map $\mathcal{M}_{\text{occ}}$ augments a 2D occupancy grid with high-risk terrain information and is defined as:

$$\mathcal{M}_{\text{occ}}(\mathbf{p}) \triangleq \mathbf{1}_{\mathcal{U}}(\mathbf{p}), \tag{14}$$

$$\mathcal{U} \triangleq \mathcal{U}_{\text{o}}\cup\mathcal{U}_{\sigma}\cup\mathcal{U}_{\text{r}}\cup\mathcal{U}_{\lambda}, \tag{15}$$

$$\mathcal{U}_{\sigma} \triangleq \{\,\mathbf{p}\in\Omega \mid \phi_{\sigma}(\mathbf{p})\ge\tau_{\sigma}\,\}, \tag{16}$$

$$\mathcal{U}_{\text{r}} \triangleq \{\,\mathbf{p}\in\Omega \mid \phi_{\text{r}}(\mathbf{p})\ge\tau_{\text{r}}\,\}, \tag{17}$$

$$\mathcal{U}_{\lambda} \triangleq \{\,\mathbf{p}\in\Omega \mid \phi_{\lambda}(\mathbf{p})\ge\tau_{\lambda}\,\}, \tag{18}$$

where $\mathbf{1}_{\mathcal{U}}(\mathbf{p}):\Omega\to\{0,1\}$ is the indicator of the non-traversable set $\mathcal{U}$. Ω is the robot workspace. $\mathcal{U}$ comprises polygonal obstacles $\mathcal{U}_{\text{o}}$ and high-risk terrain layers: steep-slope regions $\mathcal{U}_{\sigma}$, high-roughness regions $\mathcal{U}_{\text{r}}$, and perception-sparse regions $\mathcal{U}_{\lambda}$. Here $\phi_{\sigma}(\mathbf{p})$, $\phi_{\text{r}}(\mathbf{p})$, and $\phi_{\lambda}(\mathbf{p})$ denote the slope, roughness, and sparsity maps, respectively, and $\tau_{\sigma},\tau_{\text{r}},\tau_{\lambda}$ are prescribed thresholds. The free space is defined by

$$\mathcal{C}_{\text{free}} \triangleq \Omega\setminus\{\mathbf{p}\in\Omega \mid \mathcal{M}_{\text{occ}}(\mathbf{p})=1\}. \tag{19}$$

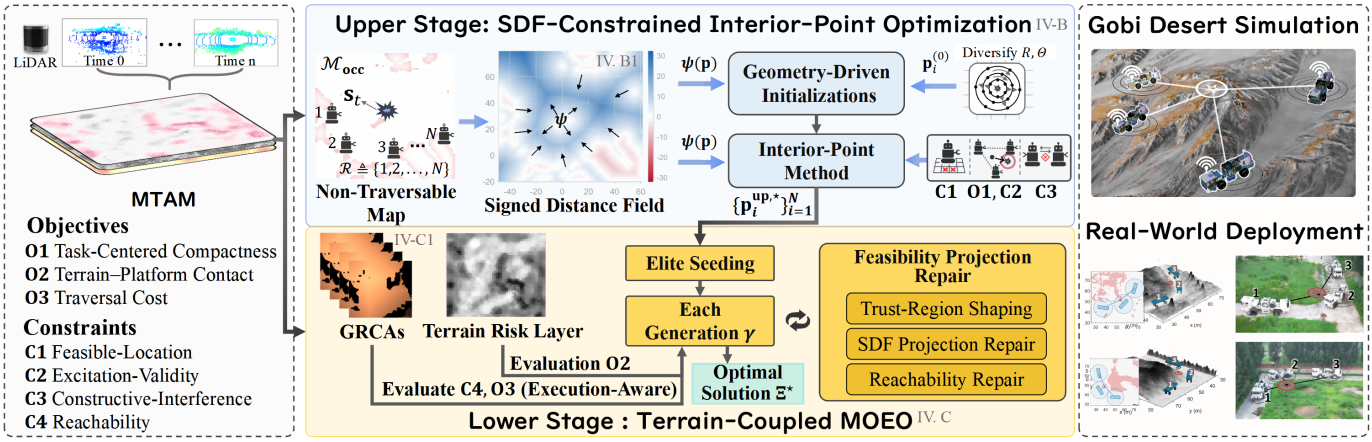


Fig. 4. Overview of the proposed terrain-coupled, hierarchical optimization framework for UOE-MRD.

Following [31], slope is estimated via principal component analysis (PCA). The slope is the angle between the local surface normal $\mathbf{n}(\mathbf{p})$ and the vertical axis $\mathbf{e}_z$:

$$\phi_\sigma(\mathbf{p}) \triangleq \arccos\left(\frac{\mathbf{n}(\mathbf{p})\cdot\mathbf{e}_z}{\|\mathbf{n}(\mathbf{p})\|}\right), \tag{20}$$

where $\mathbf{e}_z = [0,0,1]^\top$.

$\phi_{\mathrm{r}}(\mathbf{p})$ is computed using a frequency-domain terrain-profile metric [32]. We apply a Fourier transform to the local point cloud of $\mathbf{p}$ to obtain the spatial frequency response $\mathcal{A}(\mathbf{p})$ and define

$$\phi_{\mathrm{r}}(\mathbf{p}) = \frac{f_{\mathrm{samp}}}{N_{\mathrm{s}}} \sum_{k=k_{\mathrm{low}}}^{k_{\mathrm{high}}} A(k), \tag{21}$$

where N_{s} is the signal length, f_{samp} the sampling frequency, $A(k)$ is the normalized amplitude of $\mathcal{A}(\mathbf{p})$. k_{low} and k_{high} define the effective frequency band.

$\phi_\lambda(\mathbf{p})$ captures perception-related uncertainty, including effects from occlusions and limited sensor viewpoints. Let r_{b} denote the local blank ratio of the local point cloud centered at $\mathbf{p}$. Then $\phi_\lambda(\mathbf{p})$ is defined by

$$\phi_\lambda(\mathbf{p}) = \begin{cases} 1, & r_b > r_{\max}, \\ \dfrac{r_b - r_{\min}}{r_{\max} - r_{\min}}, & r_b \in [r_{\min}, r_{\max}], \\ 0, & \text{otherwise}, \end{cases} \tag{22}$$

where $r_{\min} < r_{\max}$ are user-specified bounds for r_b.

The resulting dynamic MTAM is

$$\mathbf{M}(\mathbf{p}) \triangleq \begin{bmatrix} \mathcal{M}_{\mathrm{occ}}(\mathbf{p}) \\ \phi_\sigma(\mathbf{p}) \\ \phi_{\mathrm{r}}(\mathbf{p}) \\ \phi_\lambda(\mathbf{p}) \end{bmatrix} \in \mathbb{R}^4, \tag{23}$$

which is updated in real time by the fleet.

In practical UOE-MRD, a prior global map is often unavailable due to changing terrain and limited pre-surveying. We therefore build the MTAM online using an exploration or lead vehicle within the fleet. The vehicle maintains a tightly-coupled LiDAR–inertial localization and mapping pipeline [33], which provides high-rate (typically on the order of tens of Hz) state estimates and continuously updated local point clouds. After standard preprocessing (outlier removal, cropping to a task-site neighborhood, and gridding at the chosen resolution), the MTAM layers in Eq. (23) are computed over this local workspace and updated at a typical LiDAR-driven refresh rate during the approach to the current task site. The resulting MTAM is then directly consumed by the coordinator as the map input to the deployment optimizer, enabling task-point online deployment planning.

IV. Methodology

This section presents the proposed terrain-coupled hierarchical optimization framework for UOE-MRD.

A. Motivation and Framework Overview

Directly solving the UOE-MRD problem (Eq. (11)) in a single shot is challenging in practice. The feasible set induced by constraints **C1–C3** is highly nonconvex and often fragmented in unstructured terrain, and naive global search therefore wastes a large fraction of evaluations on infeasible configurations. Meanwhile, constraint **C4** and objective **O3** require reachability and path-cost reasoning, and objective **O2** couples pose and terrain through footprint-level queries, making the objective and constraint evaluations non-analytic and expensive when repeatedly invoked within an optimizer.

To address these difficulties, we adopt a hierarchical solve-and-refine strategy that decomposes the unified problem into two tractable stages. The upper stage solves a geometry-first subproblem in the position subspace $\Xi_{\setminus\theta}$: it enforces strict feasibility with respect to constraints **C1–C3** using a continuous SDF representation and minimizes the task-centered compactness objective **O1** to generate a small set of high-quality, strictly feasible seed configurations. The lower stage then lifts these seeds to full poses Ξ and performs multi-objective refinement: it jointly optimizes all objectives under constraints **C1–C4**, while leveraging the **GRCA** to retrieve reachability and traversal cost in $O(1)$ time during evolutionary search. This decomposition preserves the original objectives and constraints, but substantially improves solvability by (i) preventing early-stage divergence caused by infeasible exploration, and (ii) enabling scalable, execution-aware

evaluation without repeated shortest-path searches, which is essential for real-time onboard MRD.

Building on this decomposition, the proposed overall framework is illustrated in Fig. 4. Specifically, an upper-stage SDF-constrained interior-point optimizer (SIPO) first produces strictly feasible, geometry-consistent position seeds, and a lower-stage GRCA-informed multi-objective evolutionary optimizer (MOEO) then refines the full fleet poses to achieve robust and scalable deployments in unstructured terrain.

B. Upper Stage: SDF-Constrained Interior-Point Optimization

We first solve a position-only nonlinear programming (NLP) problem in the subspace $\Xi_{\setminus\theta}$ to obtain strictly feasible seeds $\{\mathbf{p}_i^{\mathrm{up},\star}\}_{i=1}^{N}$ by optimizing $\mathbf{O1}$ under $\mathbf{C1}$–$\mathbf{C3}$.

A key difficulty is that the feasible-location constraint $\mathbf{C1}$ involves geometric modeling and grid-based representation of complex environments. The resulting feasible set is typically highly nonconvex due to discretization, which makes it difficult for numerical solvers to converge. To address this issue, we employ a continuous representation based on the SDF, which embeds the geometric structure of complex polygonal occupancies into the optimization process.

1) Polygonal Occupancies and Signed Distance Field: SDF encodes the Euclidean distance from any query point to the nearest boundary of an object. Its differentiability and unit-norm gradient properties make it highly compatible with gradient-based frameworks for learning, optimization, and control [34].

In $\mathcal{M}_{\mathrm{occ}}$, both polygonal obstacles and high-risk terrain regions are jointly represented by

$$\mathbb{O} \triangleq \{\mathcal{O}_k\}_{k=1}^{K}, \quad \mathcal{O}_k = \{(x_{k,m}, y_{k,m})\}_{m=1}^{M_k}, \tag{24}$$

where K denotes the number of occupied regions, M_k is the number of vertices in the k-th polygon, and $(x_{k,m}, y_{k,m})$ are the coordinates of its m-th vertex.

A conservative mask $\tilde{\mathcal{M}}_{\mathrm{occ}}$ is obtained by applying a slight morphological dilation to mitigate boundary-sampling artifacts during optimization. We compute the distance transforms for $\tilde{\mathcal{M}}_{\mathrm{occ}}$ as

$$\begin{cases} D_{\mathrm{occ}}(\mathbf{q}) \triangleq \mathrm{dist}\left(\mathbf{q}, \{\tilde{\mathcal{M}}_{\mathrm{occ}} = 1\}\right), \\ D_{\mathrm{free}}(\mathbf{q}) \triangleq \mathrm{dist}\left(\mathbf{q}, \{\tilde{\mathcal{M}}_{\mathrm{occ}} = 0\}\right), \end{cases} \tag{25}$$

where $\mathrm{dist}(\mathbf{q}, \mathcal{S})$ denotes the Euclidean distance from query point $\mathbf{q}$ to the nearest point in the set $\mathcal{S}$. The SDF is constructed from the distance transforms as

$$\psi(\mathbf{q}) = D_{\mathrm{occ}}(\mathbf{q}) - D_{\mathrm{free}}(\mathbf{q}). \tag{26}$$

Under the positive-outside and negative-inside convention, $\psi(\mathbf{q})$ satisfies

$$\psi(\mathbf{q}) \begin{cases} > 0, & \mathbf{q} \text{ lies outside the occupied region,} \\ = 0, & \mathbf{q} \text{ lies on the boundary of an occupied region,} \\ < 0, & \mathbf{q} \text{ lies inside the occupied region,} \end{cases} \tag{27}$$

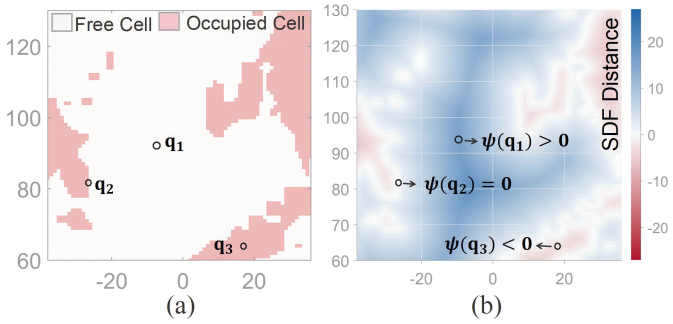


Fig. 5. SDF constructed from polygonal occupancy regions. (a) Example of a non-traversable map, where free and occupied cells are represented in white and pink, respectively. (b) Corresponding SDF derived from (a).

which is illustrated in Fig. 5. The SDF value at continuous $\mathbf{p}$ is obtained by bilinear interpolation:

$$\tilde{\psi}(\mathbf{p}) \triangleq \sum_{a=0}^{1}\sum_{b=0}^{1} w_{\mathrm{ab}}(\mathbf{p})\, \psi(i+a, j+b), \quad \mathbf{p} \in [i, i+1)\times[j, j+1), \tag{28}$$

where i and j are the integer grid indices of the cell containing $\mathbf{p}$, and $w_{\mathrm{ab}}(\cdot)$ are the standard bilinear weighting coefficients.

This construction alleviates the severe nonconvexity and discretization inherent to constraint $\mathbf{C1}$, thereby facilitating its integration into the standard NLP formulation.

2) Geometric Configuration Problem Solving: By combining Eqs. (3–9), we obtain a bounded NLP:

$$\begin{aligned} \min_{\{\mathbf{p}_i\}_{i=1}^{N}} \quad & \mathcal{J}_{\mathrm{tcc}}(\Xi_{\setminus\theta}) \;=\; \frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{p}_i - \mathbf{s}_{\mathrm{t}}\right\|_2^2 \\ \text{s.t.} \quad & \delta - \tilde{\psi}(\mathbf{p}_i) \;\le 0, && \forall\, i \in \mathcal{R}, \\ & \left\|\bar{\mathbf{p}} - \mathbf{s}_{\mathrm{t}}\right\|_2 - d_{\mathrm{exc}} \;\le 0, && \\ & d_{\mathrm{con}} - \left\|\mathbf{p}_i - \mathbf{p}_j\right\|_2 \;\le 0, && \forall\, i, j \in \mathcal{R},\ i < j, \\ & \mathbf{p}_i \in \Omega, && \forall\, i \in \mathcal{R}, \end{aligned} \tag{29}$$

where $\delta \ge 0$ is the SDF safety margin that prevents any robot from penetrating an occupied region.

To mitigate convergence to poor local minima, we construct geometry-driven initialization. Let the set of $N_R^{\max}$ candidate radii be $\mathbf{R}_{\mathrm{init}} = \{R_1, R_2, \ldots, R_{N_R^{\max}}\}$, and the set of $N_\vartheta^{\max}$ angular offsets be $\Theta_{\mathrm{init}} = \{\vartheta_1, \vartheta_2, \ldots, \vartheta_{N_\vartheta^{\max}}\}$. For each $R_k \in \mathbf{R}_{\mathrm{init}}$ and $\vartheta_m \in \Theta_{\mathrm{init}}$, the initial position of robot i is generated as

$$\mathbf{p}_i^{(0)}(R_k, \vartheta_m) = \mathbf{s}_{\mathrm{t}} + R_k \begin{bmatrix} \cos\left(\frac{2\pi(i-1)}{N} + \vartheta_m\right) \\ \sin\left(\frac{2\pi(i-1)}{N} + \vartheta_m\right) \end{bmatrix}, i = 1, \ldots, N. \tag{30}$$

where the offset ϑ_m diversifies the initial configurations.

We solve the above NLP using an interior-point method to obtain $\{\mathbf{p}_i^{\mathrm{up},\star}\}$, and impose box constraints $\mathbf{p}_i \in \Omega$. Gradient of the objective and constraints is obtained numerically by the solver using finite-difference approximations.

C. Lower Stage: Terrain-Coupled Multi-Objective Evolutionary Optimization

The terrain–platform contact objective $\mathbf{O2}$, the traversal cost objective $\mathbf{O3}$, and the reachability constraint $\mathbf{C4}$ are inherently

nonconvex, non-analytic, and strongly environment-dependent, with higher-order couplings between path and pose. Leveraging the SDF-constrained solution obtained in Sec. IV-B as a high-quality warm start, we employ an MOEO to perform a global approximation to the Pareto front. While preserving feasibility, the optimization explores an expanded search space and jointly improves the three competing performance criteria.

1) GRCA Construction and Terrain-Coupled Optimization: In addition to the objective **O1** and constraints **C1–C3** defined in Sec. IV-B, we introduce a joint representation mechanism that couples the objective **O3** with the constraint **C4**.

To efficiently evaluate the reachability and the traversal cost from robot i's initial position $\mathbf{o}_i$ to any candidate deployment pose, we precompute a **GRCA** for robot i:

$$\mathbf{GRCA}_i \triangleq \big\{\chi_i(\mathbf{o}_i,\mathbf{z}),\, C_i(\mathbf{o}_i,\mathbf{z}) \;\big|\; \mathbf{z}\in\Omega\big\},\quad \mathbf{o}_i\in\Omega, \tag{31}$$

where $\mathbf{z}\in\Omega$ denotes a generic query position in the workspace, $\chi_i(\mathbf{o}_i,\mathbf{z})\in\{0,1\}$ is a reachability indicator that equals 1 if there exists a path from $\mathbf{o}_i$ to $\mathbf{z}$ entirely contained in $\mathcal{C}_{\text{free}}$, and $C_i(\mathbf{o}_i,\mathbf{z})\in[0,+\infty]$ denotes the minimal path cost from $\mathbf{o}_i$ to $\mathbf{z}$.

We discretize Ω as a grid graph $G=(\mathcal{V},\mathcal{E})$, where $\mathcal{V}$ is the set of free cells and $\mathcal{E}$ encodes 4-connected adjacency. Prioritizing solving speed, we employ unweighted breadth-first search (BFS). Starting from $\mathbf{o}_i$, BFS propagates breadth-wise over the grid graph and records the reachability and path cost for each visited query cell $\mathbf{z}$, and $\chi_i(\mathbf{z})$ is defined as

$$\chi_i(\mathbf{z}) \;=\; \begin{cases} 1, & \exists\,\pi_i:\ \mathbf{o}_i \rightsquigarrow \mathbf{z},\ \ \pi_i\subseteq\mathcal{C}_{\text{free}},\\ 0, & \text{otherwise}, \end{cases} \tag{32}$$

where χ_i determines whether a feasible path π_i exists from $\mathbf{o}_i$ to the query $\mathbf{z}$, where $\rightsquigarrow$ denotes connectivity within the free space. By explicitly encoding reachability, χ_i effectively eliminates configurations that may appear admissible under static geometric optimization but are dynamically unreachable for ground mobile robots, as illustrated in Fig. 2 (candidate position $\mathbf{p}_4$).

The cost map C_i is defined as:

$$C_i(\mathbf{o}_i,\mathbf{z}) = \min_{\pi_i:\,\mathbf{o}_i\rightsquigarrow\mathbf{z}} \sum_{e\in\pi_i} c(\mathbf{e}), \tag{33}$$

where grid cells that are not reachable are assigned $\chi_i(\mathbf{o}_i,\mathbf{z})=0$ and $C_i(\mathbf{o}_i,\mathbf{z})=+\infty$, and $c(\mathbf{e})$ denotes the cost associated with edge $\mathbf{e}$. In this work, $C_i(\mathbf{o}_i,\mathbf{z})$ is implemented as a path-length-based proxy for traversal effort, with $c(\mathbf{e})=1$ under uniform grid spacing. It provides a lightweight execution-aware measure that distinguishes unreachable deployments and penalizes excessive detours under real-time onboard constraints.

The GRCA precomputes and encodes both dynamic reachability and path cost into a knowledge atlas for each robot, as illustrated in Fig. 6. During subsequent evaluations within the MOEO, objective **O3** and constraint **C4** can be retrieved in $O(1)$ time, substantially reducing the computational burden of the evolutionary process and avoiding the instability associated with repeated shortest-path computations across iterations.

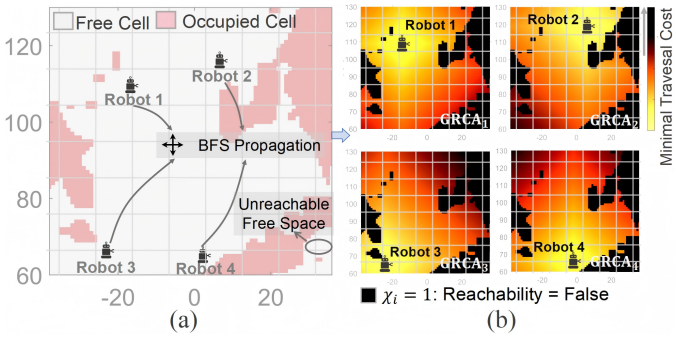


Fig. 6. Construction of the GRCAs for four robots with different initial positions. (a) Non-traversability map derived from the MTAM and the initial positions of four robots. (b) Per-robot GRCA visualizations, where traversal cost is shown in gradient color and unreachable regions are indicated in black.

Remark: The current GRCA uses 4-connected BFS with uniform edge costs rather than a high-fidelity physical traversal cost on rough terrain. We adopt this lightweight formulation to preserve fast online GRCA updates from onboard perception during multi-robot operations. More realistic reachability/cost models can be incorporated by introducing terrain-dependent edge weights and replacing BFS with Dijkstra or other planners. Therefore, the proposed framework is not restricted to uniform-cost BFS, and the specific planner can be chosen according to the desired fidelity–runtime trade-off for the target vehicle and operating conditions.

2) Multi-Objective Evolutionary Optimization Instantiation: We now turn to the solution of the unified UOE-MRD problem in Eq. (11). Building on the bounded-constraint seeds $\{\mathbf{p}_i^{\text{up},\star}\}$ obtained in Sec. IV-B, the lower stage performs terrain-coupled multi-objective refinement over the full pose variables Ξ. In this stage, the terrain–platform contact objective introduces heading-dependent optimization through the pose footprint in Eq. (4), while the GRCA provides efficient reachability and traversal-cost evaluation for execution-aware search.

The optimization is instantiated using NSGA-II [27], whose well-established nondominated sorting and crowding-distance mechanisms provide robust performance in maintaining solution diversity and search efficiency. Moreover, its constraint-domination principle enables a unified integration of the objectives and constraints modeled within the GRCA.

To fully exploit $\{\mathbf{p}_i^{\text{up},\star}\}$, we introduce a warm-start strategy combined with adaptive feasibility shaping to enhance optimization efficiency. The overall mechanism consists of three components:

(i) *Elite Seeding.* A subset of the initial population is directly initialized by perturbing $\{\mathbf{p}_i^{\text{up},\star}\}$, ensuring that high-quality individuals are embedded. This promotes stable and efficient convergence in the early search stages.

(ii) *Trust-Region Shaping.* For each $\mathbf{p}_i$, a soft constraint $\|\mathbf{p}_i-\mathbf{p}_i^{\text{up},\star}\|_2\le\rho$ is imposed, where ρ is gradually reduced across iterations to balance exploration and convergence. The resulting trust region effectively accelerates convergence in cluttered or narrow environments while maintaining population diversity.

(iii) *Feasibility Projection Repair.* After crossover and mutation, two projection-based repairs are applied sequentially. The

Algorithm 1 Terrain-Coupled MOEO

Require: Seed set $\{\mathbf{p}_i^{\mathrm{up},\star}\}_{i=1}^{N}$ from SIPO; per-robot GRCA $\{\mathbf{GRCA}_i\}_{i=1}^{N}$; bounds $\mathcal{B}$; population size P; generations Γ; initial trust region ρ_0
Ensure: Deployment solution $\Xi^\star$
1: **procedure** EVALUATE(Ξ)
2: $\mathbf{F}(\Xi) \leftarrow [\mathcal{J}_{\mathrm{tcc}}, -\mathcal{J}_{\mathrm{tpc}}, \mathcal{J}_{\mathrm{tc}}]$
3: $V(\Xi) \leftarrow$ violations from **C1**–**C4**
4: **return** $(\mathbf{F}(\Xi), V(\Xi))$
5: **end procedure**
6: **Initialization (Elite Seeding):**
7: Create $\lfloor 0.3P \rfloor$ elites by perturbing $\{\mathbf{p}_i^{\mathrm{up},\star}\}$
8: Fill remaining individuals uniformly in $\mathcal{B}$
9: Apply projection repairs (Section IV-C2)
10: Evaluate all individuals by EVALUATE($\cdot$), then perform fast nondominated sorting and crowding-distance assignment
11: **for** $\gamma = 1$ **to** Γ **do**
12: Update trust region $\rho \leftarrow \mathrm{ADAPT}(\rho_0, \gamma)$ ▷ slightly annealed across generations
13: **Selection**: Perform binary tournament selection under Deb's constraint-domination rule
14: **Variation**:
15: Apply SBX to selected pairs
16: Apply PM to offspring
17: Bound-saturate to $\mathcal{B}$
18: Apply trust-region shaping
19: Apply feasibility projection repair
20: Evaluate offspring by EVALUATE($\cdot$)
21: **Elitist merge**:
22: Merged population $\mathcal{Q} \leftarrow$ parents $\cup$ offspring
23: Perform fast nondominated sorting on $\mathcal{Q}$; refill the next population by Pareto fronts with crowding-distance tie-break
24: **end for**
25: Pareto set $\mathcal{P} \leftarrow$ nondominated solutions of the final population
26: **return** $\Xi^\star$

SDF projection repair moves $\mathbf{p}_i$ along the positive gradient direction $\nabla\tilde{\psi}$ whenever $\tilde{\psi}(\mathbf{p}_i) < \delta$, until $\tilde{\psi}(\mathbf{p}_i) = \delta$; the minimum-distance projection repair translates $\mathbf{p}_i$ and $\mathbf{p}_j$ along their connecting direction whenever $\|\mathbf{p}_i - \mathbf{p}_j\|_2 < d_{\mathrm{con}}$, until $\|\mathbf{p}_i - \mathbf{p}_j\|_2 \geq d_{\mathrm{con}}$.

In each generation, the elite NSGA-II procedure with parent-offspring merging is applied, as outlined in Lines 11–24 of Algorithm 1.

Subsequently, a reachability repair is performed using the GRCA. If $\chi_i = 0$, $\mathbf{p}_i$ is retracted along its BFS predecessor chain to the nearest reachable grid cell. Individuals that remain infeasible after repair are retained for constraint-domination comparison during selection.

The fitness evaluation follows Eq. (11). For infeasible solutions, Deb's constraint-domination principle [27] is adopted. The constraint violation is defined as:

$$V(\Xi) \triangleq \sum_{g \in \mathcal{G}(\Xi)} \max\{0, g(\Xi)\}. \tag{34}$$

When comparing two solutions Ξ_a and Ξ_b, the following rules are applied: (i) If $V(\Xi_a) = 0$ and $V(\Xi_b) > 0$, then Ξ_a dominates Ξ_b; (ii) If both are infeasible, the one with a smaller $V(\cdot)$ dominates; (iii) If both are feasible, standard nondominated sorting and crowding-distance ranking are used.

We employ a real-coded NSGA-II with simulated binary crossover (SBX) and polynomial mutation (PM), and handle boundaries via saturation clipping followed by projection repair to preserve feasibility and population diversity.

TABLE II
COMPLEXITY SUMMARY OF KEY EVALUATIONS.

Component	Complexity / Description
O1	$O(N)$: one pass over robots to compute $\mathcal{J}_{\mathrm{tcc}}$
O2	$O(N\lvert\Lambda\rvert)$: footprint query over MTAM, $\lvert\Lambda\rvert$ cells per footprint
O3	$O(N)$: GRCA lookup gives $C_i(\mathbf{o}_i, \mathbf{p}_i)$ in $O(1)$ per robot
C1	$O(N)$: $O(1)$ SDF query per robot (bilinear interpolation on grid)
C2	$O(N)$: one pass over robots to compute $\overline{\mathbf{p}}$
C3	$O(N^2)$: $N(N-1)/2$ pairwise distance checks (typically dominant)
C4	$O(N)$: GRCA lookup gives $\chi_i(\mathbf{o}_i, \mathbf{p}_i)$ in $O(1)$ per robot
SDF	$O(\lvert\Omega_g\rvert)$: distance transform on occupancy grid, $\lvert\Omega_g\rvert$ grid cells
GRCA	$O(N\lvert\mathcal{V}\rvert)$: BFS over grid graph for each robot, $\lvert\mathcal{V}\rvert$ free cells

Computational complexity analysis. Table II summarizes the per-evaluation complexity of the key objectives, constraints, and map updates in our framework. In the upper stage (SIPO), evaluating **O1** and the constraints **C1**–**C2** scales as $O(N)$, whereas enforcing the minimum-separation constraint **C3** requires $O(N^2)$ pairwise distance checks and typically dominates the per-iteration cost. Let I_{SIPO} denote the number of interior-point iterations until convergence. Since each iteration involves a constant number of such objective/constraint evaluations, the overall upper-stage complexity is dominated by $O(I_{\mathrm{SIPO}}\, N^2)$ under fixed map resolution, plus an $O(|\Omega_g|)$ cost to construct the SDF from the occupancy grid.

In the lower stage (MOEO), each candidate solution evaluation involves all objectives **O1**–**O3** and constraints **C1**–**C4** in Table II. The per-candidate cost is dominated by the pairwise separation checks $O(N^2)$ and the footprint-level MTAM queries $O(N|\Lambda|)$, while GRCA-based reachability and traversal-cost lookups contribute only linear overhead in N. Therefore, for a population size P and Γ generations, the overall lower-stage complexity is dominated by $O\big(P\Gamma\,(N^2 + N|\Lambda|)\big)$ under fixed map resolution, plus a GRCA construction cost $O(N|\mathcal{V}|)$.

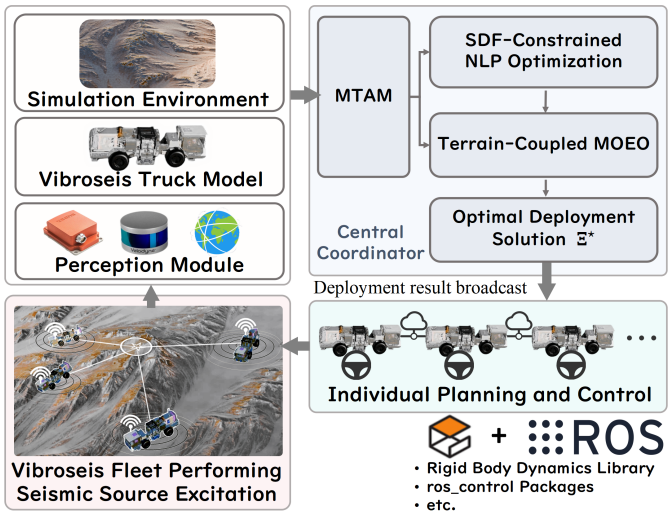


Fig. 7. Overview of the Gazebo/ROS-based workflow for evaluating the proposed UOE-MRD framework. The central coordinator executes the optimization framework and broadcasts the deployment result $\Xi^\star$ to all vehicles.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2026.3689386

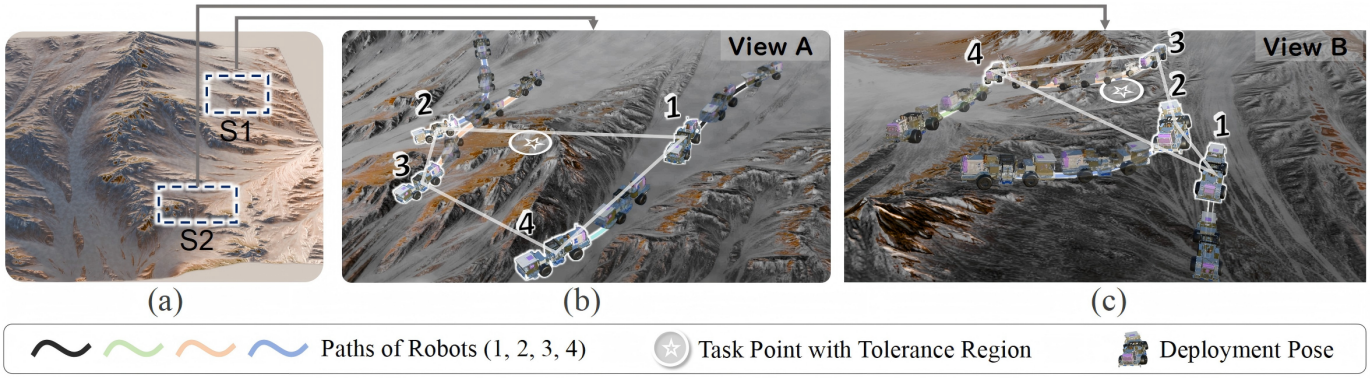


Fig. 8. (a) Global overhead view of the simulation environment with two representative sites marked. (b) and (c) show terrain details and representative four-vehicle deployment results, respectively. Each view illustrates the motion of the four vehicles from their initial positions to the final deployment poses.

V. Simulation Results

A. Simulation Setup

1) Simulation Environment and Platform: We built a Gazebo/ROS simulation environment that closely reflects the field scenario, as illustrated in Fig. 7. The environment captures representative off-road complexity—including undulating ground, height-varying structures, and realistic terrain textures—providing a reliable testbed for evaluating performance under real-world conditions, as illustrated in Fig. 8. The experimental platform models a fleet of 40-ton vibroseis trucks. Table III summarizes the key parameters of the simulation. All simulations were run on a Nuvo-7160GC industrial PC (Intel Core i7-8700 @ 4.3 GHz, 16 GB RAM) with an NVIDIA GTX 1660 Ti GPU.

TABLE III
Key Parameters of the Simulation.

Parameter	Value / Description
Environment size	430.5 m × 431.9 m (simulation area)
Grid resolution	0.5 m (LiDAR-based MTAM layers)
LiDAR configuration	32-beam LiDAR, 20 Hz scan rate
Localization update rate	30 Hz (multi-sensor fusion [33])
Communication mechanism	wireless mesh network
Coordination model	Centralized coordination
Computation platform	Nuvo-7160GC (i7-8700, GTX 1660 Ti)

TABLE IV
Hyperparameter Settings of Genetic Operators in NSGA-II.

Parameter	Setting
Population size P	50
Generations Γ	100
SDF safety margin δ	0.5 m
Initial trust-region radius ρ_0	8 m
SBX probability	$p_c \in [0.8, 0.95]$
SBX index	$\eta_c \in [10, 20]$
PM probability	$p_m = 1/(3N)$
PM index	$\eta_m \in [15, 25]$

2) Simulation Workflow: To approximate real vibroseis fleet operations, the simulation assumes no prior global map and sequential local decision making. Around each task point, the lead vehicle acquires LiDAR data and builds an MTAM in real time. The MTAM is then fed to our terrain-coupled, hierarchical UOE-MRD optimizer. All computation runs on the central coordinator (implemented on the lead vehicle), which has the latest MTAM. The coordinator solves for the deployment solution $\Xi^\star$ and broadcasts it to the fleet. In our implementation, the lead vehicle can update the MTAM online (100 m × 100 m at 0.5 m resolution) within 400 ms, and the subsequent MRD optimization is required to complete within 10 s for real-time onboard operations. This time budget is compatible with the task-point stop-and-go field operation cycle, where vehicles briefly slow down or pause for mapping and deployment planning.

The overall Gazebo/ROS-based simulation workflow is illustrated in Fig. 7. Hyperparameters are listed in Table IV. The listed parameters represent the baseline configuration and may be slightly adjusted to suit different problem settings.

3) Baseline Method and Evaluation Metrics: To the best of our knowledge, few prior studies address UOE-MRD optimization under complex, terrain-coupled conditions. Among existing works, the iterative formation-planning (IFP) pipeline [5] is widely recognized for coupling goal assignment with distributed collision avoidance. Although originally designed for pattern formation in obstacle-rich environments, its two-stage structure—integrating optimal assignment and reactive motion—makes it a strong geometric baseline for evaluating our hierarchical UOE-MRD optimization.

For fair comparison, we next outline the evaluation metrics used to assess optimization performance. Only key task-specific metrics are reported here, as common quantities such as centroid deviation and computation time are self-explanatory.

(i) *Success Rate*. An optimization trial is regarded as a failure if the solver does not converge within 10 s or any constraints (**C1**–**C4**) are violated.

(ii) *Average Roughness*. The average roughness metric evaluates terrain–platform contact quality by projecting each vehicle's deployed footprint onto the roughness layer of the MTAM and averaging the roughness values within it. A lower

value indicates flat contact surfaces and better deployment stability.

(iii) *Traversal Cost*. Traversal cost is assessed by the mean path length from each robot's initial position to its assigned deployment pose, computed using BFS on the free-space grid. Shorter paths correspond to lower nominal traversal cost and higher deployment efficiency.

B. Overall Performance

Fig. 8 shows a global overhead view of the environment and highlights two representative sites (S1 and S2) together with representative deployment results. Fig. 9(a) and (b) illustrate multiple results at S1 and S2, respectively, including runs with four robots under different initial conditions and runs with varying fleet sizes under identical initial conditions.

1) Deployment Analysis at S1: Across all results, the optimized configurations consistently form a stable arrangement around the task point (Fig. 9(a)). All robots remain in the traversable region with a safety margin to occupancies, satisfying constraint **C1**; the fleet centroid lies within the tolerance region of the task point, satisfying constraint **C2**; all pairwise separations remain compact yet greater than the minimum distance, satisfying constraint **C3**; and each robot can reach its assigned pose from its initial state, satisfying constraint **C4**.

Solutions tend to place robots on flatter terrain, reflecting a preference for objective **O2**; in terms of traversal cost, trajectories from initial positions to deployed poses avoid unnecessary detours, aligning with objective **O3**. These behaviors are visually corroborated by the photorealistic simulation snapshots shown in Fig. 8. As the fleet size increases from three to seven, the configuration evolves from a sparse ring to a locally asymmetric pattern that adapts to terrain undulations and obstacle layouts. Despite this geometric change, all constraints remain satisfied and a balanced trade-off among objectives is achieved.

2) Deployment Analysis at S2: On terrain with rich textures and steep slopes, the optimized solutions consistently satisfy constraints **C1**–**C4** (Fig. 9(b)). Under the joint influence of objectives **O1**–**O3**, the fleet preferentially occupies flat terrain and avoids slopes while respecting the constraints. However, as the fleet expands to seven robots, some vehicles are inevitably deployed on gentle slopes to preserve cooperative effectiveness. This outcome indicates that the algorithm adaptively identifies near-optimal configurations within the feasible set under multi-objective trade-offs, maximizing overall deployability and stability while satisfying all constraints.

Across both sites, the proposed method reliably produces constraint-satisfying deployments under complex terrain and randomized initial conditions, and it retains robust performance as the fleet size increases. The diversity of converged configurations under different initial conditions further demonstrates the method's ability to adaptively explore the feasible solution space.

C. Comparison and Ablation Results

Table V summarizes comparative and ablation results across four representative scenarios under the five-robot setting. Each result is averaged over 50 independent runs, and the reported values represent the mean ± standard deviation. The characteristics of all scenarios are listed in Table VI. The compared methods include the baseline IFP [5], an online-query MOEO baseline (OQ-MOEO) that evaluates reachability and traversal cost by online shortest-path queries, and the proposed full method (ours). To examine the role of the main formulation components, we also report three ablations: w/o SIPO, w/o GRCA, and w/o TPC, where TPC denotes the terrain–platform contact objective. Among them, GRCA and TPC correspond to the representative execution-aware and terrain-coupled elements in UOE-MRD, and are therefore the most informative factors to isolate.

Table VII complements this analysis by examining two algorithmic mechanisms used in MOEO, namely trust-region shaping (TRS) and feasibility projection repair (FPR), under the S2-R5 and S3-R5 settings. These components are not formulation terms of the UOE-MRD problem itself, but practical mechanisms introduced to improve convergence and feasibility preservation during evolutionary search.

1) Contribution of the SIPO: Relative to w/o SIPO, ours markedly reduces the occurrence of infeasible seeds and increases overall success rate (notably in S2 and S3), while reducing computation time in all four scenarios by approximately 12 %–23 %. The SIPO supplies feasible, geometrically consistent warm starts, thereby curbing infeasible evaluations and fruitless exploration during initialization.

2) Contribution of the GRCA: Compared with w/o GRCA, ours consistently achieves lower traversal cost and higher success rates, with notable improvements in obstacle-rich scenarios (S3 and S4). This indicates that the GRCA with dynamic reachability modeling eliminates deployment poses that are geometrically feasible but dynamically unreachable, and further mitigates unnecessary travel.

3) Contribution of the TPC: Compared with w/o TPC, ours consistently achieves lower average roughness across all four scenarios, showing that the TPC objective effectively guides the fleet toward flatter and more contact-favorable deployment poses. This gain is obtained with comparable success rates and similar runtime, indicating that TPC cannot be adequately captured by geometric feasibility together with reachability and traversal cost alone.

4) Effect of TRS and FPR: Table VII shows that removing either TRS or FPR reduces the success rate and increases the computation time in both S2-R5 and S3-R5. The average roughness also becomes slightly worse, indicating that these two mechanisms not only improve search efficiency but also help maintain feasible and stable deployment solutions during evolutionary optimization.

5) Overall Performance: Across all scenarios, the full method delivers balanced improvements across key performance metrics, with only a modest increase in runtime compared to the IFP. Moreover, the measured runtimes remain within the 10 s planning budget used for vibroseis fleet operation, supporting real-time UOE-MRD. By contrast, OQ-MOEO incurs substantially higher computation time because reachability and traversal cost are evaluated by online shortest-

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2026.3689386

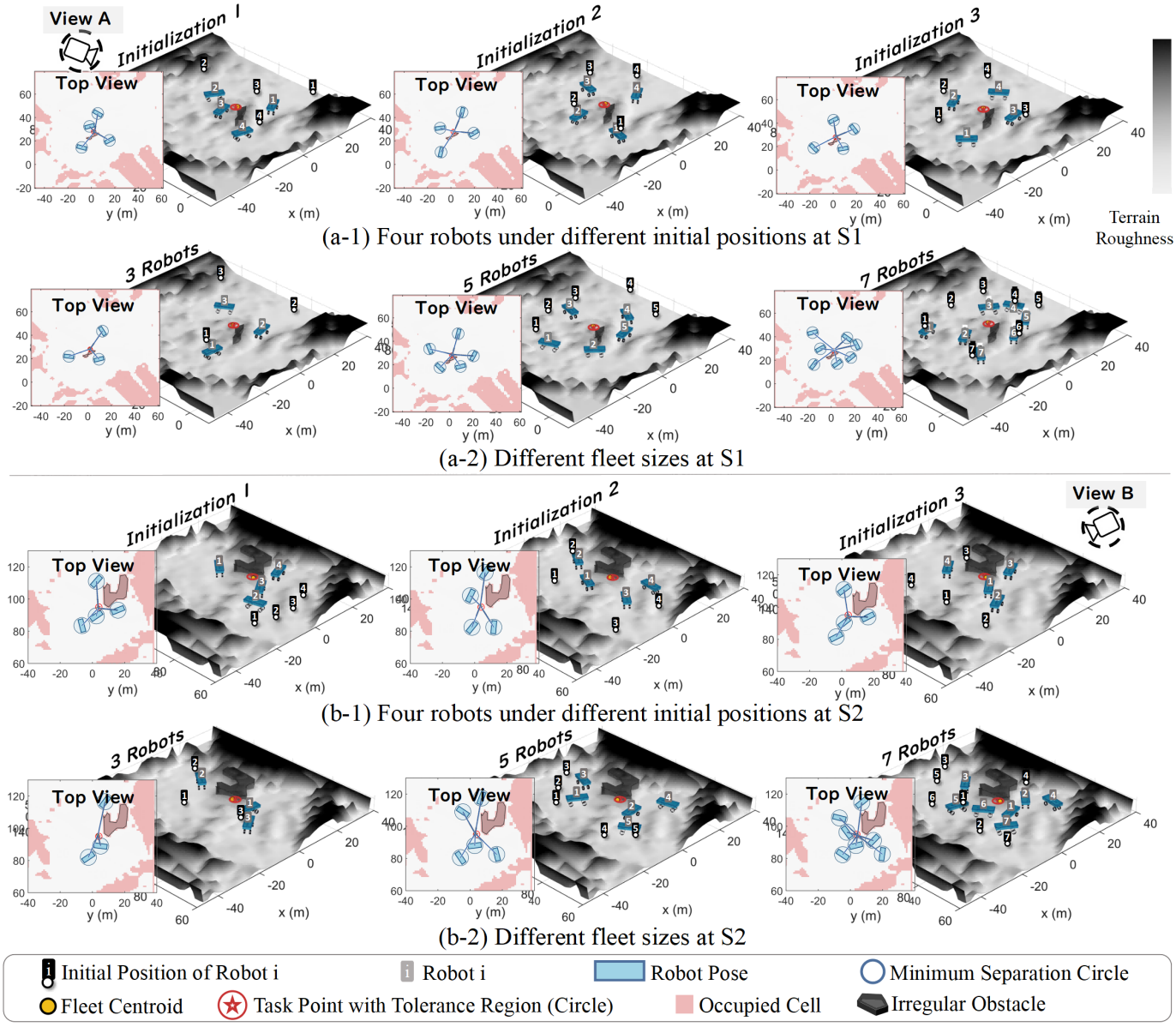


Fig. 9. Simulation results of UOE-MRD deployment at representative sites S1 and S2. Subplots (a-1) and (b-1) show deployment results under three distinct initial positions at S1 and S2, respectively. Subplots (a-2) and (b-2) present deployment results with varying fleet sizes (3, 5, and 7 robots) under identical initial positions at S1 and S2, respectively. The results labeled as View A and View B in this figure correspond to the real simulation snapshots shown in Fig. 8, which are provided for visual reference. For clarity, only the terrain roughness layer of the MTAM is visualized. Insets labeled "Top View" show the corresponding top-down projections of the fleet configurations over the occupancy map.

path queries for each candidate, and it fails to produce feasible solutions within the prescribed budget in all four scenarios.

Compared with the IFP, the proposed hierarchical UOE-MRD optimization yields substantially more robust and higher-quality solutions without incurring prohibitive overhead. These improvements stem from the terrain-coupled formulation and the GRCA-based reachability modeling, which are absent in the IFP and are seldom addressed in existing MRD studies. The slightly larger centroid deviation observed in some cases results from the multi-objective trade-off that favors stable and reachable deployment configurations.

In addition, we observe that the success rate and traversal cost vary noticeably across scenarios in Table V. This is mainly because the scenarios differ in both terrain structure (Table VI) and fleet initial layouts, which jointly affect free-space connectivity and the availability of feasible configurations under the constraints. In harsher scenarios, feasible configurations become rarer and the access paths are more likely to detour, leading to lower success rates and higher traversal costs. This cross-scenario analysis highlights the importance of evaluating UOE-MRD under diverse terrain conditions and motivates execution-aware deployment optimization in practical field operations.

TABLE V
COMPARISON OF OPTIMIZATION PERFORMANCE ACROSS FOUR SCENARIOS OVER 50 INDEPENDENT RUNS

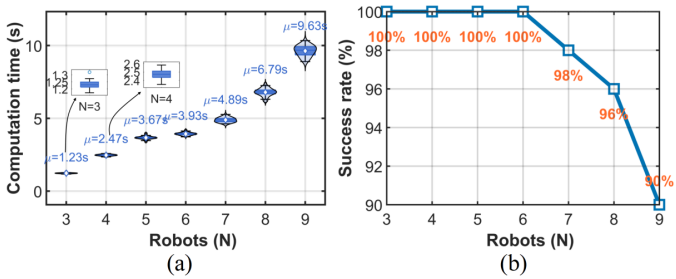
Scenario	Method	Success Rate [%]	Average Roughness	Centroid Deviation [m]	Computation Time [s]	Traversal Cost
S1–R5	IFP [5]	82.0	0.18 ± 0.06	0.38 ± 0.10	**2.83** ± 0.21	103 ± 12
	OQ-MOEO	0.0	0.09 ± 0.03	0.66 ± 0.17	13.80 ± 1.32	65 ± 8
	w/o SIPO	**100.0**	0.12 ± 0.03	0.65 ± 0.18	5.18 ± 0.51	69 ± 8
	w/o GRCA	98.0	**0.08** ± 0.02	**0.35** ± 0.09	3.47 ± 0.33	94 ± 11
	w/o TPC	**100.0**	0.14 ± 0.03	0.49 ± 0.13	3.41 ± 0.31	**63** ± 6
	ours	**100.0**	0.09 ± 0.02	0.68 ± 0.19	3.75 ± 0.36	65 ± 7
S2–R5	IFP [5]	58.0	0.25 ± 0.07	**1.07** ± 0.29	**5.77** ± 0.49	169 ± 18
	OQ-MOEO	0.0	0.17 ± 0.05	1.39 ± 0.34	20.62 ± 1.92	132 ± 16
	w/o SIPO	86.0	0.19 ± 0.05	1.46 ± 0.33	8.21 ± 0.78	157 ± 17
	w/o GRCA	80.0	**0.14** ± 0.03	1.13 ± 0.31	6.52 ± 0.59	175 ± 19
	w/o TPC	**94.0**	0.21 ± 0.06	1.13 ± 0.30	6.18 ± 0.57	**122** ± 14
	ours	92.0	0.16 ± 0.04	1.36 ± 0.32	6.87 ± 0.63	130 ± 14
S3–R5	IFP [5]	72.0	0.26 ± 0.06	**0.33** ± 0.09	**3.89** ± 0.31	191 ± 21
	OQ-MOEO	0.0	0.15 ± 0.03	0.87 ± 0.22	14.47 ± 1.55	169 ± 18
	w/o SIPO	92.0	0.17 ± 0.04	0.88 ± 0.21	5.74 ± 0.55	178 ± 19
	w/o GRCA	94.0	**0.13** ± 0.03	0.47 ± 0.12	4.47 ± 0.40	209 ± 23
	w/o TPC	**100.0**	0.20 ± 0.05	0.62 ± 0.16	4.26 ± 0.38	**161** ± 15
	ours	98.0	0.14 ± 0.03	0.86 ± 0.22	4.66 ± 0.43	166 ± 18
S4–R5	IFP [5]	76.0	0.18 ± 0.04	0.36 ± 0.10	**4.19** ± 0.35	175 ± 19
	OQ-MOEO	0.0	0.12 ± 0.03	0.46 ± 0.13	17.13 ± 1.75	129 ± 15
	w/o SIPO	98.0	0.14 ± 0.03	0.59 ± 0.15	6.03 ± 0.57	133 ± 14
	w/o GRCA	86.0	**0.11** ± 0.02	**0.33** ± 0.09	5.07 ± 0.46	168 ± 18
	w/o TPC	**100.0**	0.15 ± 0.05	0.38 ± 0.11	4.76 ± 0.44	**120** ± 15
	ours	**100.0**	**0.11** ± 0.02	0.45 ± 0.12	5.31 ± 0.49	127 ± 13

TABLE VI
CHARACTERISTICS OF FOUR SCENARIOS

Metrics	S1	S2	S3	S4
Obstacle occupancy [%]	11.53	34.67	13.86	25.61
Mean slope [°]	3.35	8.19	6.07	4.19
Mean roughness	0.20	0.35	0.27	0.19
Map size [m]	100×100	100×100	100×100	100×100

TABLE VII
ABLATION STUDY ON TRUST-REGION SHAPING AND FEASIBILITY PROJECTION REPAIR OVER 50 INDEPENDENT RUNS.

Scenario	Metric	w/o TRS	w/o FPR	Ours
S2-R5	Success Rate [%]	88.0	84.0	92.0
	Computation Time [s]	7.54 ± 0.71	8.12 ± 0.79	6.87 ± 0.63
	Average Roughness	0.17 ± 0.04	0.18 ± 0.05	0.16 ± 0.04
S3-R5	Success Rate [%]	94.0	92.0	98.0
	Computation Time [s]	5.18 ± 0.49	5.86 ± 0.56	4.66 ± 0.43
	Average Roughness	0.14 ± 0.04	0.16 ± 0.04	0.14 ± 0.03



(a) (b)

Fig. 10. Computation scalability and deployment success rate with respect to fleet size. (a) Violin–box plots of computation time over 50 independent runs, with mean values (μ) annotated. (b) Success rate over 50 independent runs.

D. Analysis of Scalability and Environmental Adaptability

1) Different Fleet Sizes: As shown in Fig. 10(a), computation time increases approximately linearly with fleet size N: the mean increases from $\mu \approx 1.23\,\mathrm{s}$ at $N = 3$ to $\mu \approx 9.63\,\mathrm{s}$ at $N = 9$, while the variance grows mildly and remains within a manageable range. The success rate is $100\,\%$ for $N \leq 6$, and drops to $98\,\%, 96\,\%, 90\,\%$ for $N = 7, 8, 9$, respectively, as shown in Fig. 10(b). This degradation is attributable to increased spatial crowding under a fixed map size, which makes concurrent satisfaction of all constraints progressively harder. However, the proposed method maintains stable convergence in the medium-size case. From an engineering standpoint given our computational budget and parameter settings, for a 100×100 map at the complexity level of S1, $N \leq 6$ can be considered a stable operating case; further scaling in N would require additional computational resources or relaxed constraints.

2) Environmental Complexity: After discretizing the environment complexity into ten levels, four metrics exhibit significant monotonic relationships with the complexity level, as illustrated in Fig. 11, which also reports Spearman's ρ. Environmental complexity is quantified by a scalar score computed from the normalized scenario statistics in Table VI, namely obstacle occupancy, mean slope, and mean roughness. We first sample candidate sites across the simulation environment and then assign each run to one of ten complexity levels

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2026.3689386

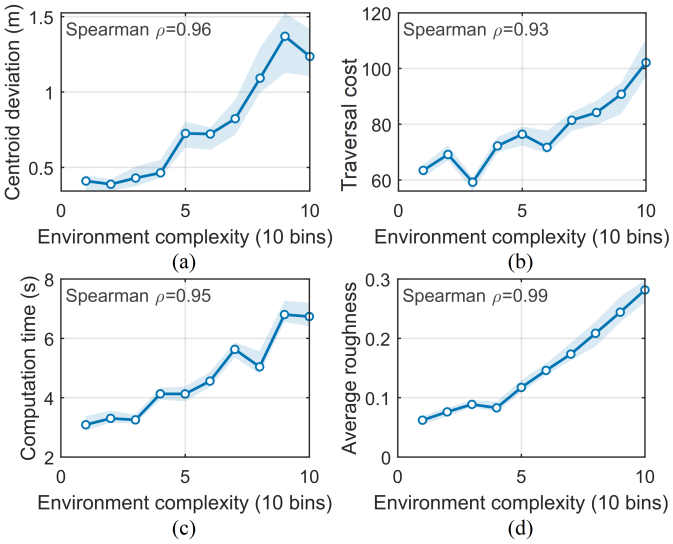


Fig. 11. Influence of environmental complexity on key performance metrics. Complexity is measured from MTAM attributes and divided into ten bins for analysis. For each bin, each circular marker denotes the median over 50 samples and the shaded region indicates the interquartile range (25th–75th percentiles).

by uniformly partitioning this score between the least- and most-complex sampled sites. In particular, centroid deviation, traversal cost, and average roughness increase markedly as complexity increases, and computation time shows a concomitant upward trend. These results indicate that, as the feasible set contracts and reachability deteriorates, the optimization must handle sharper trade-offs among objectives **O1**–**O3**. The observed degradation is primarily due to the environment-induced contraction of the feasible solution space.

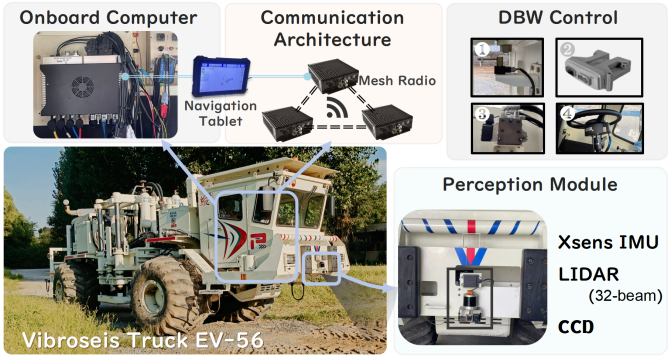


Fig. 12. Overview of the vibroseis truck platform and integrated hardware modules.

VI. Real-World Deployment

A. Field Setup

1) Platform and Environment: We conducted real-world deployments on a fleet of 40-ton vibroseis trucks. The platform was retrofitted to meet the requirements of fully unmanned multi-vehicle operation, as illustrated in Fig. 12, and integrated with the following modules: (i) *Drive-by-Wire (DBW):* steering and powertrain actuation modules enabling closed-loop lateral/longitudinal control and safe braking for precise maneuvering under autonomous operation; (ii) *Perception Module:* a 32-beam LiDAR, an Xsens IMU, and a CCD camera for MTAM construction, state estimation; (iii) *Onboard Computer:* a Miivii APEX AD10 industrial computer (12-core Arm Cortex-A78AE CPU, Ampere GPU, 64 GB RAM) providing online optimization capability in field conditions; (iv) *Fleet Communication Architecture:* a self-organizing mesh radio network that supports low-latency, reliable multi-robot communication with a tablet terminal for task points upload, progress monitoring, and real-time vehicle state visualization.

All tests were conducted at a decommissioned factory site—an unstructured environment featuring unpaved dirt segments, local subsidence, sparse vegetation, and scattered obstacles.

2) Experiment Workflow: Although this work focuses on UOE-MRD optimization, we integrate the module into a full fleet-level operational pipeline tailored for vibroseis applications to validate its practical feasibility. In field operation, the fleet adopts a centralized scheduling strategy. Prior to reaching a task point, the vehicles maintain leader-follower formation via onboard communication, with followers tracking the leader's trajectory. When approaching task point i, the fleet slows down and the lead vehicle constructs an MTAM from real-time sensor data. Based on this map, the proposed UOE-MRD framework computes the optimal deployment poses for all vehicles.

To ensure path-level safety and feasibility of these deployment poses, we employ the conflict-based search (CBS) [35] to generate executable paths for all vehicles. Vehicles then navigate independently to their assigned poses and proceed to the next task point upon completing seismic excitation. This process iterates until all task points are serviced. In our field trial over a $1.27\,\mathrm{km}$ traverse, we scheduled six task points.

B. Experimental Results

1) Analysis of Deployment Results at Representative Sites: Fig. 13 presents representative UOE-MRD optimization results for task points 1, 3, and 4. In all cases, the fleet centroids lie within the task-point tolerance region (satisfying constraint **C2**), and all pairwise separations satisfy the constraint **C3**. As shown in the roughness map layer (Fig. 13) and the field photographs (Fig. 14), the deployed poses clearly avoid non-traversable and high-risk regions and are located on flat terrain. This demonstrates that the deployments ensure operational quality and stability, reflecting the joint optimization of objectives **O1**–**O3** and the satisfaction of constraints **C1**–**C4**.

Fig. 14 further provides time-sequenced photographs for sites 1, 3, and 4, documenting the full cycle from approach and slowdown to onboard mapping, deployment optimization, and arrival at the assigned poses. After the lead vehicle constructs the MTAM and broadcasts the deployment solution, vehicles proceed along their assigned conflict-free paths and reach their designated deployment poses. Then the fleet executes the excitation task. All photographs are spatially aligned with the corresponding roughness maps in Fig. 13.

C. Analysis of Key Metrics

Fig. 15 summarizes deployment metrics across six sites. (i) *Centroid deviation* (**C2**): all six task points remain below the

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2026.3689386

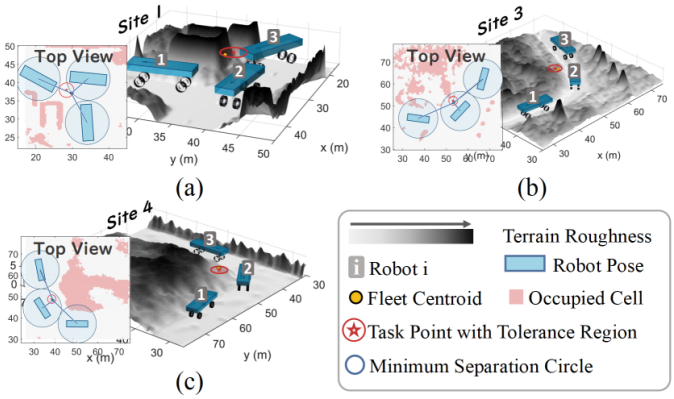


Fig. 13. Representative UOE-MRD optimization results at sites 1, 3, and 4. Site 1 adopts a minimum separation distance of 12 m, whereas sites 3 and 4 use 18 m to reflect differing task constraints. (a) Optimization result at Site 1. (b) Optimization result at Site 3. (c) Optimization result at Site 4.

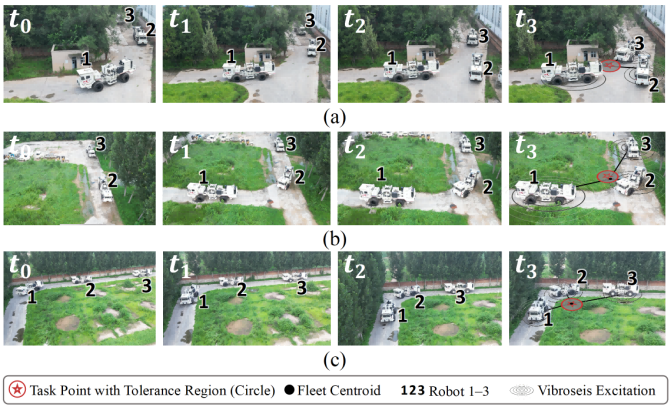


Fig. 14. Time-sequenced field deployment of the vibroseis fleet at sites 1, 3, and 4. Subplots (a)–(c) correspond to Sites 1, 3, and 4, respectively, each showing the full operation cycle (t_0–t_3) from approach and mapping to optimization and deployment.

2.0 m tolerance (max 1.77 m, min 0.36 m), indicating that the deployed poses meet operational accuracy. (ii) *Average roughness* (**O2**): the fleet's deployed poses exhibit markedly lower roughness than the site-wide average (site median ≈ 0.179 vs. fleet median ≈ 0.067), indicating a systematic preference for flatter terrain patches within the feasible deployment region. (iii) *Computation time*: average computation time is 2.35–3.22 s, demonstrating feasibility for on-vehicle deployment. (iv) *Constructive-interference constraint* (**C3**): thresholds are 12 m for task points 1, 2, 5, and 6, and 18 m for 3 and 4; all pairwise separations exceed their respective thresholds, thus satisfying constraint **C3**, as illustrated in Fig. 15(d).

The qualitative and quantitative results show that the proposed UOE-MRD framework achieves both accuracy and robustness in a real, unstructured environment. It consistently delivers feasible multi-vehicle deployments under terrain uncertainty and supports vibroseis operations, demonstrating engineering readiness and potential for wider field adoption.

VII. Conclusion

This work has presented a terrain-coupled, hierarchical optimization framework for UOE-MRD. The proposed method unifies continuous feasibility modeling and execution-aware

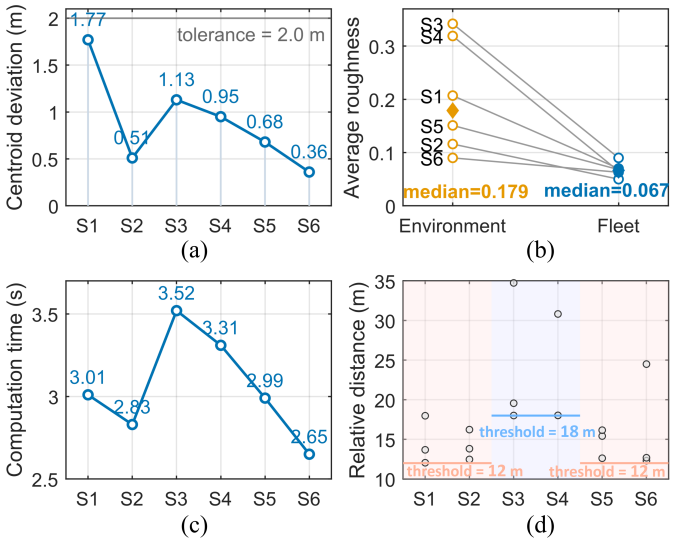


Fig. 15. Deployment performance across six representative sites. (a) Centroid deviation. (b) Average roughness. (c) Computation time. (d) Relative distance.

optimization within a unified deployment framework. An upper-stage SDF-constrained nonlinear program ensures strict geometric feasibility through terrain-informed constraints, while a lower-stage MOEO refines full poses using a precomputed GRCA. Extensive simulation and field results demonstrate that the proposed framework consistently delivers feasible, stable, and efficient deployments across diverse terrains and fleet configurations, achieving real-time on-vehicle performance and validated system-level integration on 40-ton vibroseis trucks.

While the study focuses on vibroseis fleets as an engineering validation case, the framework's formulation and hierarchical structure are broadly applicable to other field-scale UOE-MRD scenarios—such as precision agriculture, construction, and disaster response—where unstructured terrain and execution-aware constraints govern system performance. By addressing the long-standing gap between geometric placement and deployable field execution, this work provides a scalable foundation for robust, terrain-aware MRD in real-world field operations. A current limitation of the GRCA is that it is indexed by position rather than full pose, so it captures location-level reachability and access cost but does not explicitly model heading-dependent terminal feasibility or full pose-level reachability for large nonholonomic vehicles. Future work will focus on extending the proposed framework to achieve tighter integration with vehicle dynamics and richer cost models.

References

[1] L. Zhang, T. Zhang, Y. An, T. Niu, S. Wang, and J. Wang, "4WIDS-MAPF: Multiagent Pathfinding for Four-Wheel Independent Drive/Steering Robots," *IEEE Internet of Things Journal*, vol. 12, no. 18, pp. 38 795–38 803, 2025.

[2] Y. Tian, Y. Chang, F. Herrera Arias, C. Nieto-Granda, J. P. How, and L. Carlone, "Kimera-Multi: Robust, Distributed, Dense Metric-Semantic SLAM for Multi-Robot Systems," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2022–2038, 2022.

[3] Y. Zhang, X. Chen, C. Feng, B. Zhou, and S. Shen, "FALCON: Fast Autonomous Aerial Exploration Using Coverage Path Guidance," *IEEE Transactions on Robotics*, vol. 41, pp. 1365–1385, 2025.

[4] J. Li, M. Ran, and L. Xie, "Efficient Trajectory Planning for Multiple Non-Holonomic Mobile Robots via Prioritized Trajectory Optimization," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 405–412, 2021.

[5] F. Zhang, T. Wang, Q. Li, and J. Xin, "An iterative optimization approach for multi-robot pattern formation in obstacle environment," *Robotics and Autonomous Systems*, vol. 133, no. 6, 11 2020.

[6] H. Zhang, S.-H. Chan, J. Zhong, J. Li, P. Kolapo, S. Koenig, Z. Agioutantis, S. Schafrik, and S. Nikolaidis, "Multi-robot geometric task-and-motion planning for collaborative manipulation tasks," *Autonomous Robots*, vol. 47, no. 8, pp. 1537–1558, 10 2023.

[7] M. Coffey and A. Pierson, "Heterogeneous Coverage and Multi-Resource Allocation in Supply-Constrained Teams," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3447–3453, 2023.

[8] B. Woosley and P. Dasgupta, "Multirobot task allocation with real-time path planning," in *Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference (FLAIRS-26 Conference), St. Pete Beach, FL, USA*, vol. 2224, p. 574579, 2013.

[9] Y. Chen, U. Rosolia, and A. D. Ames, "Decentralized task and path planning for multi-robot systems," 2020.

[10] R. J. Alitappeh, K. Jeddisaravi, and F. G. Guimarães, "Multi-objective multi-robot deployment in a dynamic environment," *Soft Computing*, vol. 21, no. 21, pp. 6481–6497, 6 2016.

[11] S. Kim, M. Santos, L. Guerrero-Bonilla, A. Yezzi, and M. Egerstedt, "Coverage Control of Mobile Robots With Different Maximum Speeds for Time-Sensitive Applications," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3001–3007, 2022.

[12] M. Rudolph, S. Wilson, and M. Egerstedt, "Range limited coverage control using air-ground multi-robot teams," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3525–3530, 2021.

[13] M. Faghihi, M. Yadegar, M. Bakhtiaridoust, N. Meskin, J. Sharifi, and P. Shi, "Distributed optimal coverage control in multi-agent systems: Known and unknown environments," *Automatica*, vol. 173, p. 112031, 2025.

[14] M. A. Kamel, X. Yu, and Y. Zhang, "Formation control and coordination of multiple unmanned ground vehicles in normal and faulty situations: A review," *Annual Reviews in Control*, vol. 49, pp. 128–144, 2020.

[15] W. Liu, J. Hu, H. Zhang, M. Y. Wang, and Z. Xiong, "A novel graph-based motion planner of multi-mobile robot systems with formation and obstacle constraints," *IEEE Transactions on Robotics*, vol. 40, pp. 714–728, 2024.

[16] J. Guo, J. Qi, M. Wang, C. Wu, Y. Ping, S. Li, and J. Jin, "Distributed cooperative obstacle avoidance and formation reconfiguration for multiple quadrotors: Theory and experiment," *Aerospace Science and Technology*, vol. 136, p. 108218, 2023.

[17] T. Zhang, J. Xu, and B. Wu, "Hybrid path planning model for multiple robots considering obstacle avoidance," *IEEE Access*, vol. 10, no. 6, pp. 71 914–71 935, 2022.

[18] D. Thorne and B. T. Lopez, "Multi-Robot Task Assignment and Path Finding for Time-Sensitive Missions with Online Task Generation," *arXiv preprint arXiv:2310.06153*, 2023.

[19] D.-H. Nam and H.-G. Kim, "MPPI-based Integrated Task Assignment, Path Planning, and Control for Unmanned Ground Vehicles," *International Journal of Control, Automation and Systems*, vol. 22, no. 12, pp. 3641–3652, 12 2024.

[20] P. D. Cen Cheng, M. Indri, C. Possieri, M. Sassano, and F. Sibona, "Path planning in formation and collision avoidance for multi-agent systems," *Nonlinear Analysis: Hybrid Systems*, vol. 47, p. 101293, 2023.

[21] H. Hosseini, M. Taleai, and S. Zlatanova, "NSGA-II based optimal Wi-Fi access point placement for indoor positioning: A BIM-based RSS prediction," *Automation in Construction*, vol. 152, p. 104897, 2023.

[22] A. J. I. Foster, M. Gianni, A. Aly, and H. Samani, "An Efficient NSGA-II-Based Algorithm for Multi-Robot Coverage Path Planning," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5401–5407, 2025.

[23] Y. Shen and J. Meng, "Elite strategy-based improved nsga-ii algorithm for multi-robot task allocation in orbital bolting operations," *Journal of Combinatorial Mathematics and Combinatorial Computing*, vol. 127a, pp. 147–162, 4 2025.

[24] S.-C. Wang and T.-C. Chen, "Using NSGA-II to solve multi-objective competitive location problem with cooperative coverage for distance-based attractiveness," *Journal of Intelligent Fuzzy Systems*, vol. 40, pp. 7723–7734, 1 2021.

[25] N. T. Tam, T. H. Hung, H. T. T. Binh, and L. T. Vinh, "A decomposition-based multi-objective optimization approach for balancing the energy consumption of wireless sensor networks," *Applied Soft Computing*, vol. 107, p. 107365, 2021.

[26] A. Renzaglia, J. Dibangoye, V. L. Doze, and O. Simonin, "A Common Optimization Framework for Multi-Robot Exploration and Coverage in 3D Environments," *Journal of Intelligent Robotic Systems*, vol. 100, no. 3-4, pp. 1453–1468, 9 2020.

[27] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[28] E. Yakıcı and M. Karatas, "Solving a multi-objective heterogeneous sensor network location problem with genetic algorithm," *Computer Networks*, vol. 192, p. 108041, 2021.

[29] K. Pongsirijinda, Z. Cao, K. Bhowmik, M. Shalihan, B. P. L. Lau, R. Liu, C. Yuen, and U.-X. Tan, "Distributed multi-robot potential-field-based exploration with submap-based mapping and noise-augmented strategy," *Robotics and Autonomous Systems*, vol. 179, p. 104752, 2024.

[30] J. Yu and S. M. LaValle, "Optimal Multirobot Path Planning on Graphs: Complete Algorithms and Effective Heuristics," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1163–1177, 2016.

[31] A. Leininger, M. Ali, H. Jardali, and L. Liu, "Gaussian Process-based Traversability Analysis for Terrain Mapless Navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10 925–10 931, 2024.

[32] T. Niu, S. Yu, L. Wang, H. Yuan, S. Wang, and J. Wang, "Real-time terrain assessment and Bayesian-based path planning for off-road navigation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 11 700–11 706, 2024.

[33] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.

[34] Y. Li, Y. Zhang, A. Razmjoo, and S. Calinon, "Representing robot geometry as distance fields: Applications to whole-body manipulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15 351–15 357, 2024.

[35] G. Sharon, R. Stern, A. Felner, and N. R. Sturtevant, "Conflict-based search for optimal multi-agent pathfinding," *Artificial Intelligence*, vol. 219, pp. 40–66, 2015.