Computer Modeling in Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.085689

ARTICLE

OGU: Near-Optimal Group Selection of Heterogeneous Sensing UAVs via Capability Modeling and Aggregation

Xiao-Juan Li, Yu Zhang*, Xing-She Zhou, Meng-Jie Li and Xin-Yue Liu

School of Computer Science, Northwestern Polytechnical University, Xi'an, China

*Corresponding Author: Yu Zhang. Email: zhangyu@nwpu.edu.cn

Received: 15 May 2026; Accepted: 31 July 2026; Published: 28 August 2026

ABSTRACT: Sensor-equipped Unmanned Aerial Vehicles (UAVs) are increasingly deployed for collaborative aerial sensing, yet selecting an optimal subgroup from a heterogeneous fleet remains challenging. Existing approaches rank individual UAVs by fixed, isolated metrics (e.g., sensor type, residual energy) and deploy them sequentially, failing to quantify task-specific performance under coupled operational uncertainties arising from platform heterogeneity, sensor configuration, and environmental dynamics. To address this, we propose Near-Optimal Group UAV Selection (OGU), a capability-driven modeling method. Rather than directly manipulating raw, heterogeneous hardware parameters, OGU aggregates each UAV–sensor unit into a capability entity characterized by intrinsic task-oriented attributes that are computed for the specific task context. This reformulation transforms group UAV selection into capability matching, reducing the optimization complexity from a high-dimensional coupled parameter space to a compact capability space. For an area coverage task, each heterogeneous UAV–sensor pair is abstracted as a unified Sweep Capability with two attributes—coverage benefit and energy cost—computed via sub-capability aggregation. An energy-efficient greedy heuristic algorithm then identifies a near-optimal UAV subgroup that fulfills task requirements with minimal total energy expenditure by leveraging these capability attributes. Empirically, the selected subgroup deviates from the exact MILP solution by less than a 5% optimality gap. Extensive simulations demonstrate that OGU outperforms the state-of-the-art selectivity-based Opti-U method and a random-selection baseline. OGU achieves a 131.3% higher energy efficiency and a 11.2% higher area coverage ratio while deploying fewer UAVs and reduces task execution time by approximately 7× through simultaneous group deployment. The proposed capability-oriented paradigm provides a flexible and scalable framework for managing heterogeneous sensing UAVs across diverse tasks.

KEYWORDS: UAV; sensor; capability aggregation; group UAV selection; energy efficiency; area coverage; random forest regressor

1 Introduction

Sensor-equipped UAVs provide a flexible, on-demand sensing platform for applications including air pollution monitoring [1], forest fire detection [2], precision agriculture [3], industrial inspection [4], and disaster management [5]. As sensing UAV deployments expand, platform managers must select suitable UAVs from the available fleet to satisfy diverse sensing requirements.

Existing approaches assess individual UAV performance based on metrics such as sensor utilization, reliability, and hovering capability [6,7], selecting the single "best" UAV with the highest score. However, these rank-and-select methods become inadequate as task complexity and multi-UAV collaboration requirements increase. For large-scale tasks, they must execute sequentially—selecting and deploying UAVs one at a time and awaiting each deployment's completion before choosing the next. This sequential dependency neglects the benefits of coordinated teams and increases overall execution time. Group UAV selection overcomes this limitation by enabling simultaneous task execution and leveraging complementary capabilities.

While pre-defined metrics enable unified comparison of individual platforms, they are fundamentally inadequate for group selection. Group selection requires quantifying the task-specific contribution of each UAV and evaluating collective effectiveness, rather than merely ranking isolated features such as sensor modality or residual energy. Quantifying such task-specific performance is particularly challenging because it is governed by multi-dimensional uncertainty including task objectives, sensor characteristics, path-planning strategies, environmental conditions, and especially UAV mobility. Unlike static attributes, mobility dynamically determines sensing range, data quality, and energy expenditure during execution. Most critically, UAV motion is coupled with other factors, generating a vast trajectory solution space. This introduces significant uncertainty in predicting performance, rendering exhaustive evaluation of all possible group compositions computationally prohibitive.

This paper addresses the following problem: Given a fleet of sensing UAVs whose performance is subject to mobility-related uncertainties arising from the interplay of sensors, environmental conditions, and task objectives, how can we accurately predict UAV performance under dynamic conditions and simultaneously select a subset to maximize task fulfillment?

To address this challenge, we propose OGU (Near-Optimal Group UAV Selection), a near-optimal group selection scheme that simultaneously identifies a UAV group to maximize task performance. The core idea is to abstract the task-specific performance of each sensor-UAV unit into a unified capability descriptor through a formal aggregation model, thereby decoupling low-level hardware heterogeneity from high-level group selection. Rather than manipulating raw, tightly coupled platform parameters directly, OGU transforms low-level hardware resources and environmental constraints into high-level, task-oriented capability metrics. It comprises (i) a capability aggregation model that predicts the task-specific performance—including both task achievement and associated costs—of each heterogeneous UAV, and (ii) an energy-efficient greedy heuristic algorithm (OGU-greedy) that leverages these pre-aggregated capability metrics to rapidly select a UAV group satisfying the task requirements. We demonstrate OGU in a representative area coverage task whose objectives are full area coverage and minimizing energy consumption (Fig. 1).

OGU operates in three stages. First, heterogeneous UAVs and their onboard sensors are abstracted into unified, standardized Imaging Capability and Flight Capability descriptors that account for environmental factors. Second, each sensor-UAV pair is further abstracted into a Sweep Capability—characterized by coverage benefit and coverage cost attributes—-derived from the sub-capabilities through area-maximizing path-optimization. Finally, the greedy algorithm efficiently assembles a UAV group that provides full coverage at minimal energy cost by iteratively selecting the unit with the highest coverage-to-energy ratio based on the pre-aggregated Sweep Capabilities.

The main contributions of this paper are summarized as follows:

1. **Problem Formulation.** We identify and formally define the optimal group UAV selection problem under coupled operational uncertainties arising from platform heterogeneity, sensor configuration, and environmental dynamics. We demonstrate that direct optimization over the raw, coupled parameter space incurs prohibitive computational cost due to exponential combinatorial growth in the fleet size and the tightly coupled nature of motion, sensing, and energy constraints.
2. **Capability Modeling and Aggregation Framework.** We propose a novel capability abstraction framework that transforms the group selection problem from device-centric parameter computing into

task-oriented capability matching. The framework introduces (i) a unified capability entity structure $\langle \mathrm{ID}, \mathcal{A}, \mathcal{X} \rangle$ that abstracts each heterogeneous UAV–sensor unit as a task-oriented performance entity, independent of its underlying hardware implementation; and (ii) a capability aggregation operator Φ that compresses the high-dimensional, coupled interactions among UAV motion, sensor configuration, and environmental dynamics into compact, task-level attributes (e.g., coverage benefit and energy cost). This abstraction achieves dimensionality reduction and shields the selection logic from raw parameter complexity.

3. **OGU Methodology: From Abstraction to Near-Optimal Selection.** We instantiate the proposed framework into a complete methodology—near-optimal group UAV selection (OGU)—for single-camera area coverage tasks. This includes: (a) formal definition of Imaging and Flight sub-capabilities with geometric imaging models and data-driven Random Forest energy regression; (b) synergistic aggregation into Sweep Capability via area-maximizing path optimization; and (c) the OGU-greedy algorithm that selects UAVs by ranking pre-computed coverage-to-energy ratios.
4. **Experimental Validation and Robustness Analysis.** We evaluate the OGU pipeline through extensive simulations across diverse UAV–sensor configurations and target scales. Statistical analyses confirm the framework's superior robustness ($p < 0.001$) and consistent advantages over multiple baselines (selectivity-based Opti-U and Random). Furthermore, benchmarking against exact MILP solvers demonstrates near-optimality (within a 5% gap) and microsecond-level execution latency, ensuring real-time feasibility for dynamic missions.

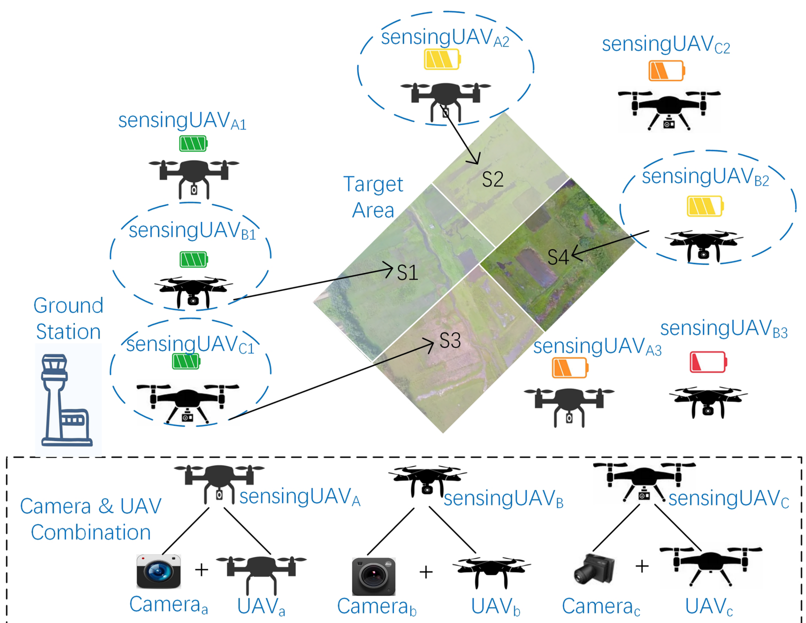


Figure 1: Group sensing UAV selection: sensing $UAV_{Ai,Bi,Ci}$ represent combinations of heterogeneous UAVs and cameras with different positions and battery levels. Sensing $UAV_{A2,B1,B2,C1}$ are selected to cover the target area with minimal energy consumption.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 presents the background and motivation for group UAV selection. Section 4 formulates the problem and details the capability aggregation model. Section 5 describes the OGU method for the area coverage task. Section 6 reports the experiments, and Section 7 concludes the paper.

2 Related Works

UAV selection for aerial sensing extends sensor selection to mobile platforms with three-dimensional maneuverability and energy-constrained continuous operation. Unlike static sensor deployment, the problem must account for dynamic flight constraints that couple sensing performance with platform motion. To contextualize this distinction, we review mobile sensor selection in sensor networks and opportunistic mobile sensing, then examine existing UAV selection approaches that partially bridge these directions yet remain limited for task-driven aerial operations.

2.1 Sensor Selection under Constrained Mobility

Sensor selection in Wireless Sensor Networks (WSN) optimizes energy efficiency and data quality [8]. In this context, mobile sensor selection typically concerns sensors mounted on movable devices (e.g., robotic platforms, relocatable ground nodes) whose mobility range is physically constrained and their mobility is treated only as a deployment-tuning mechanism.

Among the studies that do address mobility, Wang et al. [9] proposed a geometry-based method that optimizes mobile sensor selection for localization accuracy by maximizing the determinant of the Fisher Information Matrix (FIM) and strategically relocating sensors. Álvarez et al. [10] integrate sensor selection into the Node Location Problem for TDOA localization, demonstrating that combining selection with placement reduces Cramér-Rao uncertainties compared with decoupled approaches. Ghazalian et al. [11] introduced Quality of Experience (QoE) as a selection criterion for cameras with adjustable angles, considering both target coverage and visual data quality.

A common limitation across these approaches is that mobility is exploited only to improve initial geometry; sensors remain effectively static during actual operation. Consequently, they do not address continuous, task-driven mobility where the platform must sustain movement to fulfill mission objectives.

2.2 Opportunistic Mobile Sensing and Participant Selection

Mobile sensing leverages smartphones or vehicle-mounted sensors for large-scale urban data collection, where mobility is opportunistic and uncontrolled. Participant selection has been studied to optimize cost, coverage, and data quality.

Ding et al. optimized participant selection based on historical data quality to maximize cost-efficiency and platform profit, but their method overlooked variations in individual contributions [12]. Tan et al. focused on maximizing coverage with minimal participants by utilizing location-based data [13], while Sun et al. emphasized data reliability and proposed a cost-reduction strategy using historical records [14]. Zhu et al. ranked sensors based on spatio-temporal coverage, data utility, and energy cost, selecting those with the highest composite scores [15]. Zhang et al. employed a Greedy heuristic to maximize sensing coverage by evaluating worker potential through coverage functions [16]. A common limitation of these approaches is the underlying assumption that participants move relatively slowly and that sensing performance remains invariant during the selection window.

A related line of research exploits vehicle mobility to extend spatial coverage and sensing modality diversity. Toledo et al. [17] developed stochastic approaches to minimize the number of sensors in cooperative spectrum sensing for randomly moving nodes. Zhu et al. [18] predicted vehicle mobility with deep learning and applied a Greedy algorithm for vehicle selection under budget constraints. Hu et al. [19] utilized driving patterns to predict sensing coverage and enable effective fleet selection.

While these methods acknowledge mobility's impact on coverage, they fundamentally rely on opportunistic, unplanned motion. This stands in stark contrast to deliberately planned, task-driven mobility in UAV-based aerial sensing, where trajectories are explicitly designed to satisfy sensing objectives—a controllability that demands a distinct selection paradigm.

2.3 Task-Driven UAV Selection

One line of research focuses on selecting UAVs for communication [20–23] and computation tasks [24,25]. For example, Bithas and Moustakas [21] selected UAVs based on channel state information to ensure optimal link quality in distributed communication systems. Zhu et al. [26] employed a utility-based function to optimize UAV positioning for improved service delivery. Bousbaa et al. [27] adopted a game-theoretic approach to study service selection in UAV cloud computing. Hadjkouider et al. [28] proposed a Stackelberg game-based model for UAV selection, taking into account residual energy and delay time to allocate services at competitive prices.

These works optimize network throughput or computation load, with UAVs treated as static or quasi-static infrastructure nodes. Because their objective functions and platform models are designed for communication and computation scenarios rather than sensing task fulfillment, they are not methodologically compatible as baselines for our sensing-centric group selection problem.

Another research direction explores UAV selection for various sensing applications. Mufalli et al. [29] selected sensors with lighter weight for UAV assembly and routed UAVs through target fields to maximize intelligence gain. Motlagh et al. [6] formulated linear integer programming models considering equipment type, residual energy, distance, and speed. Roy and Bouvry [7] proposed Opti-U, which calculates a "selectivity" score based on sensor type, utilization rate, energy level, and failure rate for UAV-as-a-Service platforms. Gao [30] employed dual hesitant fuzzy sets, the best-worst method, and MULTIMOORA to rank UAV alternatives under uncertainty using expert assessments.

However, these sensing-UAV selection studies employ fixed, isolated metrics that evaluate sensor characteristics and UAV capabilities separately, thereby neglecting the group selection requirements and the impact of UAV motion on sensing performance. To address these limitations, this paper proposes a novel group UAV selection method that explicitly models the interaction between UAV mobility and sensing performance through a capability aggregation model.

3 Background and Motivation

As aerial sensing tasks grow in scale and complexity, deploying multiple UAVs in a coordinated manner has become essential [31,32]. However, existing swarm management strategies, including collaborative task allocation [33], path planning [34,35], UAV deployment [36] and area decomposition [37], often adopt an all-available policy, dispatching every UAV in the fleet without discrimination. While this maximizes parallel task execution, it ignores the fundamental trade-off between task completion time and resource utilization, leading to excessive cumulative energy consumption and unnecessarily complicated task assignment and data processing [38].

Motivation

The fixed-metric baseline ranks candidates by a weighted composite score combining sensor type, residual energy, and geographical proximity (following the selectivity paradigm of [6]).

To quantitatively expose the structural limitations of fixed-metric sequential selection, we conducted controlled preliminary simulations using a heterogeneous fleet of 5 rotary-wing UAVs with varying battery capacities (2590–5500 mAh), maximum speeds (12–20 m/s), and camera resolutions (812 × 620 to

3840 × 2160 pixels). The flight power data is computed from an open data set [39] using a machine learning method detailed in Section 5.4.2. The target areas include a 500 × 400 m^2 rectangle (Area 1) and a 1000 × 100 m^2 rectangle (Area 2), both requiring full coverage at a uniform GSD of 10 cm/pixel. We test three cases: 5 UAVs with (1) same battery level (80%), same start position; (2) same battery level and different start position; (3) different battery level and start position. We first perform a task-time simulation. In this sequential paradigm, each iteration selects only the single "best" UAV based on predefined indicators. As shown in Fig. 2, the selected UAV executes its planned path until reaching its energy threshold, after which the next UAV resumes from the preceding end position. This process repeats until the entire area is covered.

However, this paradigm creates a strict sequential dependency: a new selection cannot commence until the previous UAV completes its assignment and objectives are reassessed. As illustrated in Fig. 3, this dependency causes the total task duration to increase linearly with the number of iterations, significantly degrading efficiency. To overcome this inefficiency, simultaneous group selection is necessary to transition from serial execution to parallel deployment, thereby significantly reducing overall task duration.

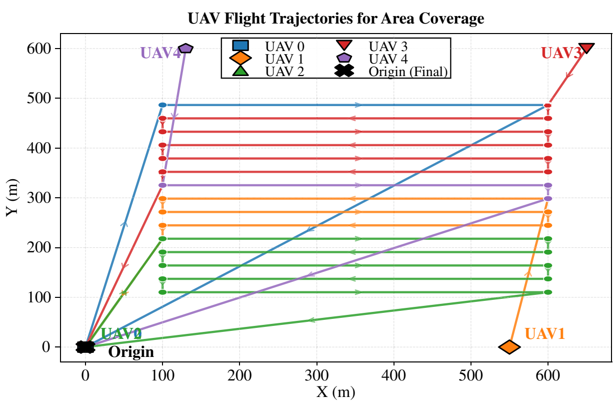


Figure 2: Flight path of selected UAVs for area coverage.

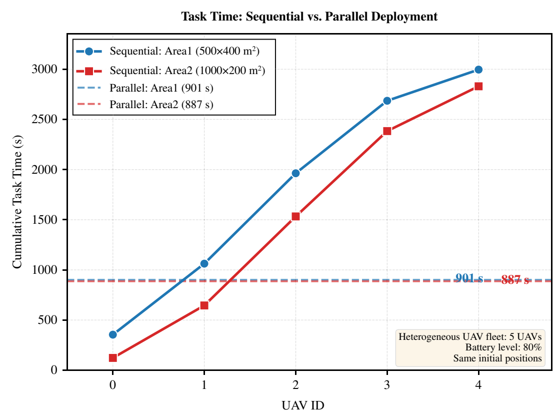


Figure 3: Task time of sequential selection.

Realizing such simultaneous group selection, however, requires accurate quantification of each UAV–sensor unit's task-specific performance under dynamic conditions. A closer examination of the fixed metrics employed by existing methods reveals two fundamental limitations:

1. **Neglecting the effect of UAV motion on sensing performance.** Current selection approaches consider sensor-task compatibility, residual energy [7], and UAV location [6]. However, these metrics are inherently static and fail to capture how UAV flight dynamics affect actual sensing effectiveness. As shown in Figs. 4 and 5, the UAV ranked highest by fixed metrics did not yield the maximum coverage area or the highest energy efficiency. Early-selected UAVs exhibited inferior coverage performance compared to later-selected ones, indicating a mismatch between the selection score and the actual achievable coverage.
2. **Neglecting the influence of sensor configuration on UAV motion.** Existing methods treat sensor parameters as independent selection criteria, overlooking the fact that sensor characteristics significantly affect flight-path planning. As Fig. 6 illustrates, when the same UAV carries cameras of different resolutions during a monitoring task, the resulting path coverage area differs markedly; the highest-resolution camera enables 386% more coverage than the lowest-resolution one under the same energy budget and GSD constraint. Sensor parameters must therefore be integrated into motion evaluation for accurate performance prediction.

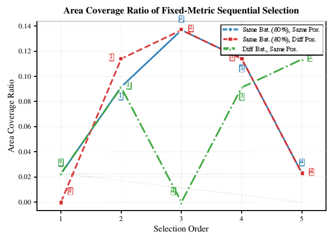


Figure 4: Suboptimal selection in covered area.

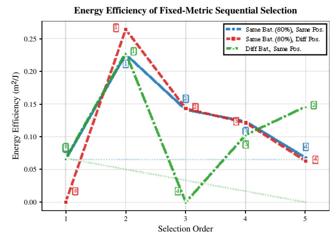


Figure 5: Suboptimal selection in energy efficiency.

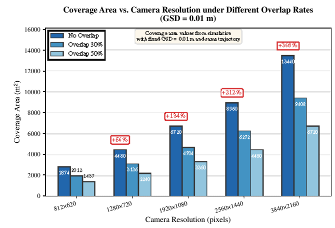


Figure 6: Coverage difference with varying camera resolution.

Fixed-metric approaches fail to quantify the task-specific performance of individual UAV–sensor units. In collaborative missions, the inability to assess dynamic contributions prevents identifying the optimal combination for collective effectiveness. Furthermore, the selection problem is inherently combinatorial, a challenge exacerbated by coupled uncertainties in UAV mobility, sensor heterogeneity, and environmental dynamics. These factors lead to an exponential expansion of the search space, rendering exhaustive evaluation computationally prohibitive. Addressing this requires a unified methodology that can: (i) accurately assess heterogeneous units under dynamic conditions, and (ii) facilitate efficient group selection—a gap that current methods have yet to fill.

4 Problem Formulation and Capability Aggregation Model

This section formalizes the group selection problem for heterogeneous UAV–sensor fleets and introduces a capability aggregation model to translate composite performance into task-oriented descriptors.

4.1 Group UAV Selection Problem Formulation

Let $U = \{u_1, \ldots, u_n\}$ denote the set of candidate UAVs. Each UAV u_i carries a sensor suite $S_i = \{s^i_1, \ldots, s^i_{m_i}\}$, where $m_i = |S_i|$ is the number of sensors on UAV u_i. A task set $T = \{T_1, \ldots, T_t\}$ is given. For a specific task T_l, we define two types of binary decision variables:

- $y_i \in \{0, 1\}$: whether UAV u_i is selected into the subgroup ($y_i = 1$) or not ($y_i = 0$);
- $x_{i,j} \in \{0, 1\}$: whether sensor s^i_j (the j-th sensor on UAV u_i) is activated for task T_l.

The objective is to identify a subset $U_{\text{sel}} \subseteq U$ that minimizes total energy expenditure while satisfying the task performance requirement. This yields the following formulation:

$$\min_{x_{i,j}, y_i} \quad \sum_{u_i \in U} \left[C_{\text{uav}}(u_i, T_l) \cdot y_i + \sum_{s^i_j \in S_i} C_{\text{sensor}}(u_i, s^i_j, T_l) \cdot x_{i,j} \right] \tag{1a}$$

$$\text{s.t.} \quad \sum_{u_i \in U} \sum_{s^i_j \in S_i} B(u_i, s^i_j, T_l) \cdot x_{i,j} \geq O(T_l) \tag{1b}$$

$$\sum_{s^i_j \in S_i} C_{\text{sensor}}(u_i, s^i_j, T_l) \cdot x_{i,j} + C_{\text{uav}}(u_i, T_l) \cdot y_i \leq E^{\max}_i \cdot y_i, \quad \forall u_i \in U \tag{1c}$$

$$\sum_{s^i_j \in S_i} D(u_i, s^i_j, T_l) \cdot x_{i,j} \leq D^{\max}_i \cdot y_i, \quad \forall u_i \in U \tag{1d}$$

$$y_i \leq \sum_{j=1}^{m_i} x_{i,j}, \quad \forall i \in \{1, \ldots, n\} \tag{1e}$$

$$\sum_{j=1}^{m_i} x_{i,j} \le M \cdot y_i, \quad \forall i \in \{1, \ldots, n\} \tag{1f}$$

$$x_{i,j} \le y_i, \quad \forall i \in \{1, \ldots, n\}, \ \forall j \in \{1, \ldots, m_i\} \tag{1g}$$

$$\sum_{i=1}^{n} y_i \ge 1, \tag{1h}$$

$$x_{i,j} \in \{0,1\}, \ y_i \in \{0,1\}, \quad \forall i \in \{1, \ldots, n\}, \ \forall j \in \{1, \ldots, m_i\} \tag{1i}$$

Here, $C_{\text{uav}}(u_i, T_l)$ denotes the platform-level energy cost of UAV u_i for task T_l (including flight, communication, and avionics), while $C_{\text{sensor}}(u_i, s_j^i, T_l)$ denotes the incremental energy cost of operating sensor s_j^i, $B(\cdot)$ denotes the task performance benefit, and $O(T_l)$ denotes the minimum performance requirement for task type T_l. $E_i^{\max}$ and $D_i^{\max}$ are the battery capacity and onboard disk storage limit of UAV u_i, respectively. Constraint (1b) ensures aggregate task satisfaction. Constraint (1c) and (1d) enforce per-UAV battery and storage limits, where the right-hand side is written as $E_i^{\max} \cdot y_i$ and $D_i^{\max} \cdot y_i$ to ensure that inactive UAVs ($y_i = 0$) consume no resources. Constraints (1e)–(1f) are the sensor cardinality bounds. Constraint (1e) prevents "empty" UAV selection (a UAV is selected but carries no active sensor). Constraint (1f) limits the number of concurrently active sensors on each UAV, where $M \in \mathbb{Z}^+$ is the platform-specific maximum number of simultaneously activatable sensors per UAV. Constraint (1g) enforces the physical coupling: sensors cannot be activated independently of their host UAVs. Constraint (1h) guarantees that at least one UAV is assigned to the task. (1i) enforce binary decisions.

Both $B(\cdot)$ and $C(\cdot)$ are high-dimensional, tightly coupled mappings of UAV physical attributes $\{\mathcal{P}_{u_i}\}$ (mass, battery, propulsion), UAV states $\{\mathcal{P}_{us_i}\}$ (position, velocity, attitude), sensor parameters $\{\mathcal{P}_{s_j^i}\}$ (FOV, resolution, focal length), and task requirements $\{\mathcal{P}_{T_l}\}$ (coverage area, spatial resolution, altitude).

Directly optimizing Problem (1) over this raw, heterogeneous parameter space is computationally prohibitive for three reasons: *(i)* the benefit and cost functions depend on heterogeneous device parameters that vary widely across platforms; *(ii)* flight dynamics, sensor characteristics, and environmental conditions introduce uncertainty to the sensing performance; and *(iii)* exhaustive evaluation of all 2^n subsets incurs exponential complexity. For $n > 10$, each subset evaluation further requires high-dimensional trajectory optimization subject to coupled sensor-motion-energy constraints, rendering exhaustive search computationally impractical.

4.2 Capability Aggregation Model

To make Problem (1) tractable, we propose a capability aggregation model that maps the heterogeneous, tightly coupled parameter space into unified, task-oriented performance descriptors. Inspired by the virtualization paradigm in sensor networks [40] and UAV systems [41,42], our model treats "capability" as a high–level abstraction—focusing on functional outcomes rather than physical implementation [43]. This approach effectively decouples selection logic from hardware specifics while encapsulating the complex interplay between UAV mobility, sensor configurations, and environmental dynamics into compact capability attributes.

4.2.1 Capability Definition

Definition 1 (Capability): *A capability $Cap_{i,j}^{(l)}$ is an entity with attributes, that quantifies the task-specific performance of the j-th functional component of device u_i for T_l.*

$$Cap_{i,j}^{(l)} = \left\langle \mathrm{ID}(Cap_{i,j}^{(l)}), \mathcal{A}(Cap_{i,j}^{(l)}), \mathcal{X}(Cap_{i,j}^{(l)}) \right\rangle, \tag{2}$$

where:

- $\mathrm{ID}(Cap_{i,j}^{(l)}) = \{name, identifier\}$ *is the unique designation;*
- $\mathcal{A}(Cap_{i,j}^{(l)}) = \{Attri_k^{C_{i,j}^{T_l}}\}$ *is a set of quantitative attributes such as range, accuracy, and energy consumption;*
- $\mathcal{X}(Cap_{i,j}^{(l)}) = \{Env_k^{C_{i,j}^{T_l}}, Sys_k^{C_{i,j}^{T_l}}\}$ *represents external environmental factors (e.g., wind, illumination) and internal system configurations.*

The specific instantiation depends on device functionality and task type; for example, a camera provides Imaging Capability for visual tasks, while a LiDAR provides Ranging Capability for spatial mapping.

4.2.2 Capability Aggregation

Individual constituent capabilities, termed **sub-capabilities**, are aggregated to form a **composite capability** $Cap_{\mathrm{aggr}}^{(l)}(u_i)$, representing the joint task performance of a UAV–sensor pair. This aggregation is formulated as:

$$Cap_{\mathrm{aggr}}^{(l)}(u_i) = \Phi\Big(\{Cap_{\mathrm{sub},k}^{(l)}(u_i)\}_{k=1}^{n_s}; \mathcal{W}_i\Big), \tag{3}$$

where $\{Cap_{\mathrm{sub},k}^{(l)}\}_{k=1}^{n_s}$ denotes the set of n_s sub-capabilities, and $\mathcal{W}_i = \{w_{i,1}, w_{i,2}, \ldots\}$ represents the weight coefficients reflecting their relative importance. The aggregation operator Φ is designed to satisfy the zero-property: the composite capability nullifies (degrades to zero) if any essential sub-capability is zero.

The aggregation process consists of two sequential steps:

Step 1: Sub-Capability Attribute Computation. The attributes of each sub-capability are derived from UAV physical parameters $\{\mathcal{P}_{u_i}\}$, UAV state $\{\mathcal{P}_{us_i}\}$, sensor parameters $\{\mathcal{P}_{s_j^i}\}$, and contextual factors Env_i and Sys_i, expressed as $Attri_k^{C_{i,j}^{T_l}} = \mathrm{AttriFunc}_k\big(\{\mathcal{P}_{u_i}\}, \{\mathcal{P}_{us_i}\}, \{\mathcal{P}_{s_j^i}\}, Env_i, Sys_i\big)$. This function serves as a generalized mapping that is instantiated based on specific capability and task requirements. To accurately evaluate effectiveness, we establish tailored models for different capability attributes, which are further elaborated in Section 5.

Step 2: Task-Specific Optimization. Sub-capability attributes are fused through the aggregation function Φ to yield the composite capability attributes. This step solves a task-specific optimization problem (e.g., path optimization for area coverage) that determines the best achievable performance of the UAV–sensor pair under the prevailing constraints, producing a compact, task-level descriptor.

By replacing the raw, high-dimensional parameter mappings $B(u_i, s_j^i, T_l)$ and $C(u_i, s_j^i, T_l)$ with composite capability attributes, the proposed model fundamentally restructures the group selection problem in three critical ways. First, it decouples performance assessment from hardware heterogeneity by representing disparate UAV–sensor pairs through unified, task-level capability descriptors. Second, it encapsulates complex motion dynamics—coupled with sensor configuration, environmental conditions, and task objectives—into a structured interplay of sub-capabilities. Third, it enables the selection algorithm to operate on compact attributes rather than the vast combinatorial parameter space, thereby facilitating efficient greedy optimization.

5 The OGU Approach

This section instantiates the capability aggregation model for an area coverage task and details the OGU group selection method.

5.1 Capability-Based Area Coverage Formulation

UAV sensing tasks can be categorized into three modes based on the coupling between UAV motion and sensor action:

1. **Instantaneous Point Coverage:** Sensors collect data (e.g., temperature, air quality) as the UAV passes over discrete locations [6].
2. **Persistent Point Coverage:** The UAV hovers above targets, enabling continuous data capture [44,45].
3. **Continuous Area Coverage:** Sensors must sweep every position within a large target area while the UAV is in motion [46].

In Types 1 and 2, UAV flight and sensor operation are loosely coupled, so their task contributions can be evaluated separately. Continuous area coverage (Type 3) presents the most significant selection challenge because the sensing effects are directly governed by the UAV's flight path, speed, and altitude—all of which are tightly coupled with sensor configuration and environmental conditions. This creates a vast, complex trajectory search space that exemplifies the mobility-related uncertainty addressed in Section 3.

We consider a heterogeneous rotary-wing UAV fleet tasked with monitoring a large target area $\mathcal{A}$ (e.g., a forest or urban region). All UAVs are centrally controlled, though they differ in platform specifications, sensor configurations, battery levels, and storage capacities. Some may be stationed at a base, while others are already airborne from prior missions. The objective is to select a UAV subgroup that achieves full coverage of $\mathcal{A}$ with minimal total energy expenditure.

5.1.1 Model Specialization: From UAV–Sensor Pairs to Single-UAV Selection

The general formulation (1) accommodates heterogeneous UAV fleets where each platform may carry multiple sensors ($m_i \geq 1$) and the optimizer decides which subset to activate, subject to the coupling and cardinality constraints. However, for the area coverage tasks addressed in this paper, we focus on a specific yet prevalent configuration:

Assumption 1 (Single-Sensor-per-UAV): *Each UAV u_i is equipped with a single fixed-focus RGB camera, $S_i = \{s_1^i\}$, and $m_i \triangleq |S_i| = 1, \forall u_i \in U$. Furthermore, the platform restricts the number of active sensors to $M = 1$.*

Under Assumption 1, the sensor activation variable $x_{i,j}$ has only one admissible index $j = 1$. Substituting $m_i = 1$ and $M = 1$ into the cardinality bounds (1e)–(1g) yields: $y_i \leq x_{i,\,1} \leq y_i$, which implies the equality: $x_{i,1} = y_i, \quad \forall u_i \in U$. Introducing the scalar UAV-selection indicator:

$$x_i \triangleq x_{i,1} = y_i \in \{0,1\}, \quad \forall u_i \in U. \tag{4}$$

The sensor activation variable $x_{i,j}$ and the UAV selection variable y_i collapse into a single binary indicator $x_i \in \{0,1\}$, where $x_i = 1$ denotes that UAV u_i and its sole camera are selected. The linking constraint (1g) becomes redundant and the double summations in (1b)–(1d) which enforce (i) no sensors can be activated if the UAV is not selected, (ii) at least one sensor is activated whenever a UAV is selected, and

(iii) no more than M sensors on the same UAV are active simultaneously, collapse to single terms, allowing the double summations in the general model (1) to be reduced to scalar terms.

5.1.2 Capability Instantiation and Nominal Assumptions

For the area coverage task T_{cov}, we encapsulate the raw, high-dimensional parameters of each UAV–sensor pair into a task-oriented composite capability descriptor with attributes (α_i, β_i), where:

- α_i denotes the **net effective coverage area** accumulated by UAV u_i and its sensors s_j^i along its planned path within the target area;
- β_i denotes the **total energy cost** consumed for coverage task, including in-strip flight, camera operation, communication, and the reach/return transit.

To ensure the tractability of these pre-computed metrics, we adopt the following *nominal assumptions*: (i) quasi-steady level flight with negligible wind; (ii) obstacle-free airspace over flat terrain; and (iii) UAV dynamics that support boustrophedon trajectories without actuator saturation. Under these conditions, (α_i, β_i) serve as reliable nominal proxies for task performance, internalizing footprint overlaps and motion-induced consumption.

5.1.3 Problem Formulation

By substituting the composite capability attributes into the capability-based formulation, the area coverage group selection problem is expressed as:

$$\min_{x_i} \sum_{u_i \in U} \beta_i \cdot x_i \tag{5a}$$

$$\text{s.t.} \quad \sum_{u_i \in U} \alpha_i \cdot x_i \geq \Omega \tag{5b}$$

$$\beta_i \cdot x_i \leq E_i^{\max}, \quad \forall u_i \in U \tag{5c}$$

$$D_i \cdot x_i \leq D_i^{\max}, \quad \forall u_i \in U \tag{5d}$$

$$x_i \in \{0,1\}, \quad \forall u_i \in U \tag{5e}$$

where Ω is the total area coverage requirement, $E_i^{\max}$ and $D_i^{\max}$ are battery and storage limit of UAV u_i, respectively. Constraint (5b) ensures the selected UAVs meets the target area requirement Ω. Constraint (5c) enforces per-UAV energy limits; constraint (5d) limits onboard storage. The transition from these pre-computed scalar estimates to physical sub-region assignments and sweep paths is handled by the deployment procedure in Section 5.5.2.

Compared with the raw formulation (1), the capability-based model (5) replaces the complex, coupled parameter mappings with pre-computed capability attributes (α_i, β_i), thereby enabling efficient group selection. Table 1 provides a detailed mapping of variables from the general framework to this specialized area coverage model.

Table 1: Mapping from general formulation (1) to capability-based formulation (5).

General Model (1)	Area Coverage Model (5)	Reduction Condition
y_i (UAV selection)	x_i	Assumption 1
$x_{i,j}$ (sensor selection)	x_i	Degenerates via Eq. (4)
$y_i \leq \sum_j x_{i,j}$ in (1e)	—	Degenerate under Assumption 1

(Continued)

Table 1 (continued)

General Model (1)	Area Coverage Model (5)	Reduction Condition
$\sum_j x_{i,j} \le My_i$ in (1f)	—	Degenerate under Assumption 1 ($M = 1$)
$x_{i,j} \le y_i$ in (1g)	—	Redundant under Assumption 1
$B(u_i, s_j^i, T_l)$	α_i	Capability aggregation
$C(u_i, s_j^i, T_l)$	β_i	Capability aggregation
$O(T_l)$	Ω	Single task ($T_l = T_{\text{cov}}$)
$\sum_{j=1}^{m_i}(\cdot)$ in (1e)–(1f)	Single term $(\cdot)$	$m_i = 1$ via Assumption 1

5.2 OGU Overview

OGU selects an energy-efficient group of sensing UAVs for area coverage by modeling the interplay between sensing effectiveness and UAV motion (flight path, speed, altitude, sensor configuration, and environmental conditions). The method comprises three stages illustrated in Fig. 7:

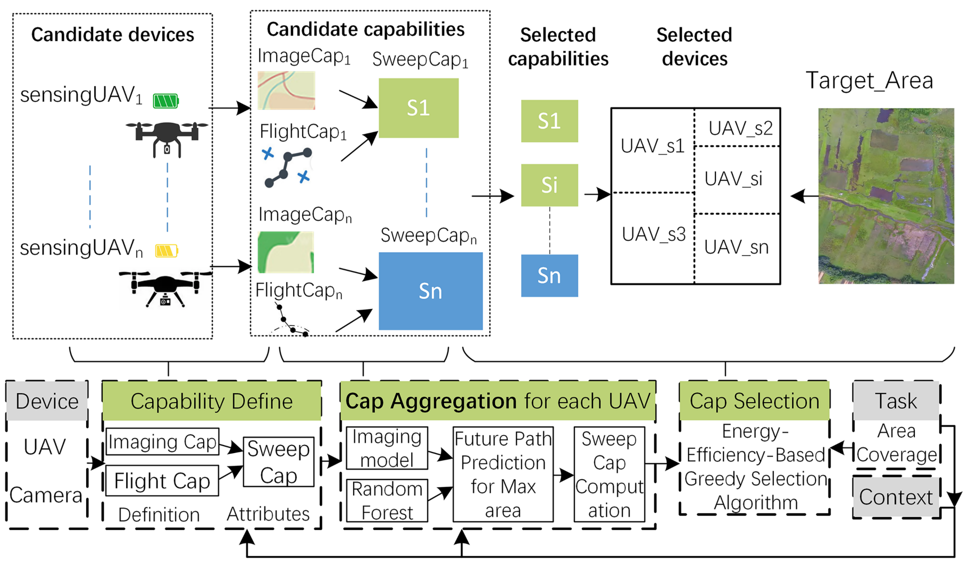


Figure 7: Overview of the capability-based selection method.

Stage 1—Capability Definition. We formally define two sub-capabilities—**Imaging Capability** for the camera payload and **Flight Capability** for the UAV platform—and aggregate them into the **Sweep Capability**, characterized by *effective coverage area* α_i (benefit) and *energy cost* β_i (cost).

Stage 2—Capability Computation. We construct device-specific models to compute the attributes of each sub-capability. The Imaging Capability is derived from the camera's intrinsic characteristics (resolution, focal length, field of view). The Flight Capability, particularly energy consumption, is predicted by a Random Forest regressor accounting for wind, payload, and platform dynamics. These sub-capabilities are then fused through an area-maximizing path optimization to determine the Sweep Capability attributes (α_i, β_i).

Stage 3—Group Selection. Using the pre-computed Sweep Capability attributes, we formulate the group selection as a 0–1 integer programming problem and solve it via the **OGU-greedy algorithm, which iteratively selects UAVs by the coverage-to-energy ratio** $\rho_i = \alpha_i/\beta_i$.

5.3 Capability Definition

5.3.1 Imaging Capability

The **Imaging Capability** Cap_i^{img} characterizes the visual sensing performance of the camera payload on UAV u_i in terms of coverage footprint, spatial resolution, capture frequency, and power consumption. It is determined by the camera's intrinsic parameters and the task resolution requirement. The ground sample distance GSD^{img_i} and effective coverage footprint $L_h^{'img_i}$, $L_v^{'img_i}$ are different from the hardware parameters.

Definition 2 (Imaging Capability): *The Imaging Capability of UAV u_i is defined as*

$$Cap_i^{\text{img}} = \langle \text{ID}^{\text{img}_\text{i}}, \mathcal{A}_k^{\text{img}_\text{i}}, \mathcal{X}_k^{\text{img}_\text{i}} \rangle,$$

- $\text{ID}^{\text{img}_\text{i}} = \{$"*imaging*", $i\}$;
- $\mathcal{A}_k^{\text{img}_\text{i}} = \{L_h^{'img_i}, L_v^{'img_i}, h^{img_i}, GSD^{img_i}, Fr^{img_i}, P^{img_i}, \}$ *comprises:*
  - $L_h^{'img_i}$, $L_v^{'img_i}$: *effective field of view (ground coverage horizontal/vertical at operating altitude);*
  - GSD^{img_i}: *ground sample distance (cm per pixel), determined by focal length, pixel size, and imaging distance;*
  - Fr^{img_i}: *frame capture rate (frames per second);*
  - h^{img_i}: *distance from camera to target (m);*
  - P^{img_i}: *imaging power consumption (W).*
- $\mathcal{X}_k^{\text{img}_\text{i}} = \{Rg^{T_l}\}$: *the task-required ground resolution.*

5.3.2 Flight Capability

The Flight Capability Cap_i^{flt} characterizes the platform-level flight and energy performance of UAV u_i under realistic operational conditions, abstracting away the complexity of specific flight paths, which is essential for predicting coverage endurance.

Definition 3 (Flight Capability): *The Flight Capability of UAV $u_i Cap_i^{\text{flt}}$ is defined as*

$$Cap_i^{\text{flt}} = \langle \text{ID}^{\text{flt}_\text{i}}, \mathcal{A}_k^{\text{flt}_\text{i}}, \mathcal{X}_k^{\text{flt}_\text{i}} \rangle,$$

- $\text{ID}^{\text{flt}_\text{i}} = \{$"*flight*", $i\}$;
- $\mathcal{A}_k^{\text{flt}_\text{i}} = \{Ei^{flt_i}, Pa^{flt_i}, Sm^{flt_i}, Er^{flt_i}\}$ *comprises:*
  - Ei^{flt_i}: *initial battery energy (J);*
  - Pa^{flt_i}: *action power consumption or energy consumtion for turning, hover, and cruise;*
  - Er^{flt_i}: *residual energy (J) at the current operational state;*
  - Sm^{flt_i}: *maximum allowable flight speed (m/s).*
- $\mathcal{X}_k^{\text{flt}_\text{i}} = \{w_{\text{payload}}^{flt_i}, v_{\text{wind}}^{flt_i}\}$: *payload weight and wind speed.*

5.3.3 Sweep Capability

The task-oriented capability aggregation is illustrated in Fig. 8. For area coverage, we aggregate Imaging and Flight Capabilities into the **Sweep Capability** Cap_i^{sweep}, which depends on both image quality and navigation efficiency. Imaging Capability determines the instantaneous coverage achievable at any given moment, while Flight Capability determines the cumulative coverage over the flight duration. The context influencing Sweep Capability includes the sub-capabilities: Imaging Capability Cap_i^{img} and Flight Capability Cap_i^{flt}.

Definition 4 (Sweep Capability): *The output of the aggregation function* Φ*:*

$$Cap_i^{\mathrm{sweep}} = \Phi\left(Cap_i^{\mathrm{img}}, Cap_i^{\mathrm{flt}}; \mathcal{W}_i\right) = \langle \alpha_i, \beta_i \rangle,$$

where $\mathcal{W}_i$ *denotes the aggregation weights. The composite attributes are* α_i *and* β_i*, same as in Eq.(5).*

The Sweep Capability Cap_i^{sweep} thus provides a compact, task-level performance descriptor that encapsulates the joint influence of imaging quality and flight endurance on area coverage effectiveness. These attributes directly populate the group selection model (5), decoupling selection criteria from low-level hardware parameters.

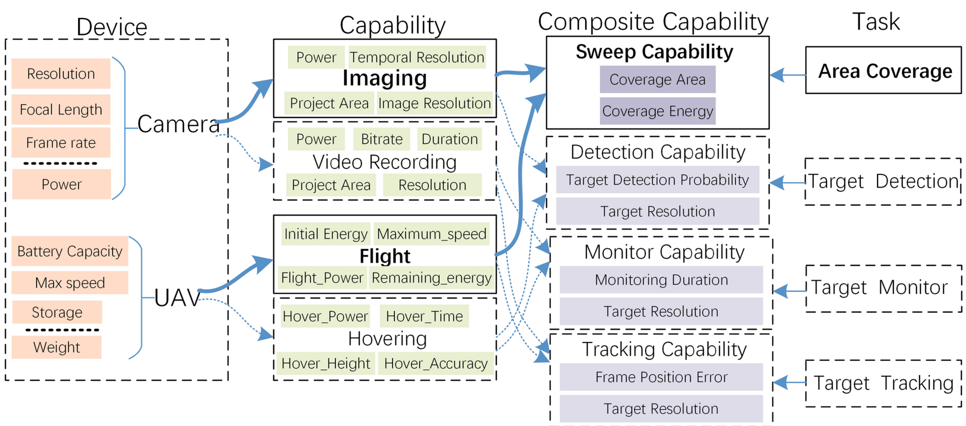


Figure 8: Task-oriented capability abstraction.

5.4 Computation of Aggregation for a Single UAV

Capability attributes vary with operational context and the computation involves different models. This section details the calculation of the capability attributes for a sensing UAV u_i. For simplicity, the subscript i has been omitted.

5.4.1 Imaging Capability Computation

Fig. 9 depicts the geometric imaging model employed to derive the imaging capability attributes. The meaning of symbols is shown in Table 2.

The primary contextual factor for Imaging Capability Cap_i^{img} is the task resolution requirement Rg^{T_l}, specified as the minimum Ground Sample Distance (GSD). The GSD requirement dictates the operating altitude and effective coverage footprint of the camera. The required camera altitude (h^{img_i}) can be calculated using Eqs. (6)–(8):

$$FOV_h = 2\arctan\left(\frac{P_h}{2f}\right) \quad FOV_v = 2\arctan\left(\frac{P_v}{2f}\right) \tag{6}$$

$$L_h(h) = 2h\tan\left(\frac{FOV_h}{2}\right) \quad L_v(h) = 2h\tan\left(\frac{FOV_v}{2}\right) \tag{7}$$

$$GSD = \frac{L_h}{P_h} = \frac{2h\tan\left(FOV_h/2\right)}{P_h} \tag{8}$$

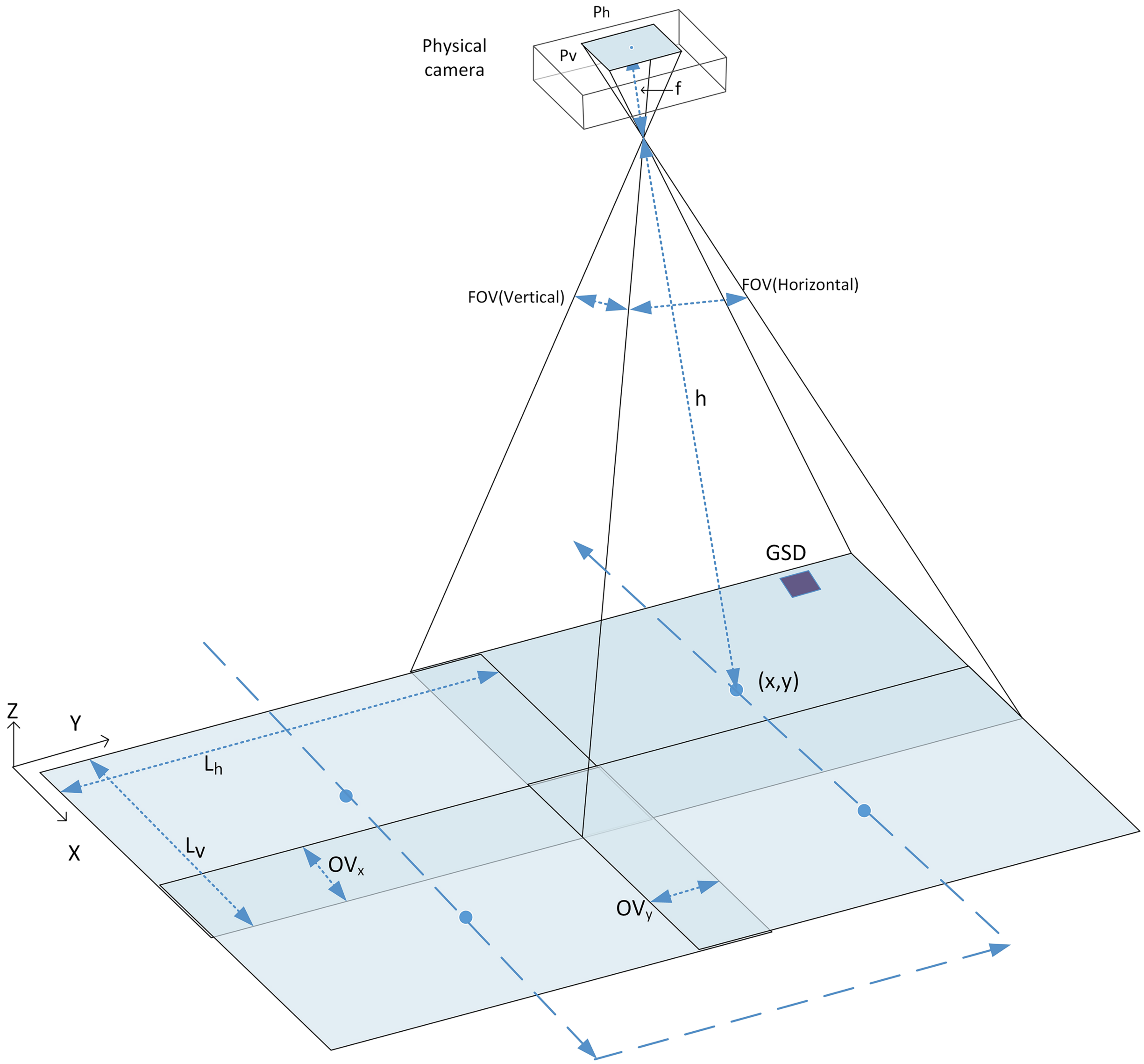


Figure 9: Geometric imaging model illustrating the relationship among flight altitude h, camera focal length f, sensor size, pixel dimensions, ground sampling distance (GSD), ground coverage width L, and image overlap ratio OV. This is a physics-based geometry model used to compute imaging capability parameters.

Table 2: Main notation table.

Notation	Description
f	Focal Length of the camera
FOV	Field of view of camera
P_h, P_v	Horizontal and vertical pixel dimensions of the camera
FR	Frame rate of the camera
h	Flight Altitude of the UAV
GSD	Ground Sampling Distance in cm/pixel
$L_h(h)$, $L_v(h)$	The length and width of the projected area of the camera from height h

(Continued)

Table 2 (continued)

Notation	Description
ROV_x, ROV_y	Horizontal and Vertical Image Overlap Rate
OV_x, OV_y	Overlap distance along and perpendicular to the direction of UAV movement.
$L'_h(h)$, $L'_v(h)$	Effective horizontal and vertical ground coverage distances at altitude h.
P_{cam}	Power consumption of the camera.

Given the image overlap rates (ROV_x, ROV_y), the effective projected area of i-th Imaging Capability ($S_h^{'img_i}$) from altitude (h) of the camera can be determined using Eqs. (10) to (12):

$$OV_y = L_h(h) \cdot ROV_y, \quad OV_x = L_v(h) \cdot ROV_x \tag{9}$$

$$L_h^{'img_i} = L'_h(h) = L_h(h) - OV_y, \quad L_v^{'img_i} = L'_v(h) = L_v(h) - OV_x \tag{10}$$

$$S_h^{'img_i} = L_h^{'img_i} \cdot L_v^{'img_i}, \tag{11}$$

$$h^{img_i} = h. \tag{12}$$

In addition to the altitude (h^{img_i}) and effective coverage dimensions ($L_h^{'img_i}$, $L_v^{'img_i}$) calculated using the imaging model, the following attributes of Imaging Capability can be directly mapped from task attributes or camera specifications:

- **Ground resolution** (GSD^{img_i}): = Rg^{T_l}, the task-specified GSD;
- **Frame rate** (Fr^{img_i}): = FR, the camera's nominal frame rate;
- **Imaging power** (P^{img_i}): = P_{cam}, the camera's rated power consumption.

5.4.2 Flight Capability Computation

UAV are energy-constrained, the energy consumption of flight actions directly affects their flight duration and, consequently, their coverage area. Flight Capability focus on attributes related to flight energy consumption. The Initial_Energy, Residual_Energy, and Maximum_Speed attributes of Flight Capability are derived from the UAV's manufacturer specifications and its current operational state.

- **Initial_Energy** (Ei^{flt_i}): the total stored energy of the battery, calculated from its rated voltage V_{bat} (V) and capacity C_{bat} (Ah): $Ei^{flt_i} = V_{bat} \times C_{bat} \times 3600$ [J].
- **Residual_Energy** (Er^{flt_i}): the remaining energy at the current operational state, determined by the initial energy and the battery state-of-charge ratio $P_{bat} \in [0, 1]$: $Er^{flt_i} = Ei^{flt_i} \times P_{bat}$.
- **Maximum_Speed** (Sm^{flt_i}): the platform's rated maximum flight speed, obtained from manufacturer specifications.

The Action_Power attribute (Pa^{flt_i}) captures the power consumption of individual flight actions under varying flight states and environmental conditions. The total energy consumption along a flight path is obtained by summing the energy expenditures of the constituent actions. Because manufacturers do not provide action-specific power tables, we train Random Forest models using an open UAV flight dataset [39], which contains multi-UAV flight records (velocity, position, power) under diverse conditions (altitude, speed, payload weight, wind speed, wind direction). The modeling pipeline is shown in Fig. 10. The main two stages are flight action recognition and action energy regression.

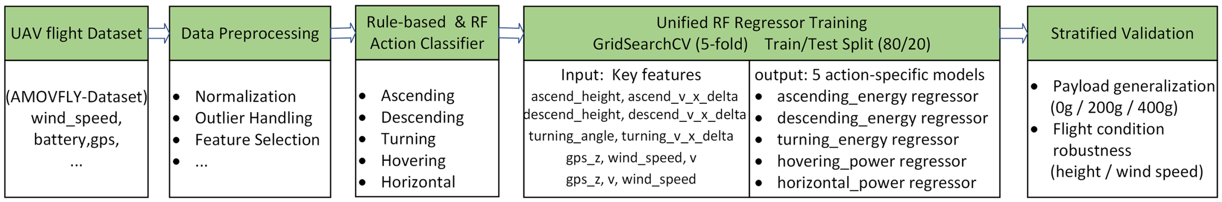


Figure 10: Action power/energy prediction model pipeline. (1) Raw flight data from the AMOVFLY dataset are preprocessed. (2) Rule-based segmentation and an RF classifier recognize flight segments into five regimes. (3) Action-specific RF regressors are trained. (4) Stratified validation assesses generalization across payload configurations (0–400 g) and flight conditions (altitude, wind speed).

Stage 1—Flight Action Recognition. We categorize UAV motion into five actions: hovering, ascending, descending, turning, and horizontal flying, not limited to hovering and forward flight [47–49], each characterized by distinct power profiles.

Two approaches are explored for action labeling: (i) a physics-based rule set exploiting deterministic kinematic boundaries, and (ii) a Random Forest classifier trained on the rule-generated labels as ground truth.

Rule-based segmentation. The primary labeling method employs kinematic thresholds on vertical speed v_z, horizontal speed $v_{xy} = \sqrt{v_x^2 + v_y^2}$, altitude change rate $\dot{h}$, and heading change rate $\dot{\psi}$:

$$\hat{y}_{\text{rule}}(t) = \begin{cases} \text{ascending}, & \text{if } v_z(t) > v_{z,\text{th}} \text{ and } \dot{h}(t) > 0, \\ \text{descending}, & \text{if } v_z(t) < -v_{z,\text{th}} \text{ and } \dot{h}(t) < 0, \\ \text{hovering}, & \text{if } v_{xy}(t) < v_{\text{hover}} \text{ and } |v_z(t)| < v_{z,\text{th}}, \\ \text{turning}, & \text{if } |\dot{\psi}(t)| > \dot{\psi}_{\text{th}} \text{ and } v_{xy}(t) > v_{\text{hover}}, \\ \text{horizontal}, & \text{otherwise}, \end{cases} \tag{13}$$

where $v_{z,\text{th}} = 0.5$ m/s, $v_{\text{hover}} = 1.0$ m/s, and $\dot{\psi}_{\text{th}} = 15°$/s. These thresholds are empirically validated against the flight dynamics of all three UAV types.

RF classifier (validation). A Random Forest classifier is additionally trained to cross-check the rule-based labels. The classifier achieves more than 95% accuracy on a held-out test set, confirming that the kinematic boundaries are learnable. To prevent classification error propagation, all energy regressors in Stage 2 are trained on rule-generated labels.

The feature set for both approaches includes wind speed, wind angle, UAV position (gps_x, gps_y), altitude (gps_z), orientation (o_x, o_y, o_z, o_w), ground speed (v_x, v_y, v_z), and ground acceleration (la_x, la_y, la_z).

The Random Forest classifier is an ensemble of B CART trees trained to minimize the Gini impurity. For a node containing samples from K action classes, the Gini impurity is defined as: $G = 1 - \sum_{k=1}^{K} p_k^2$, where p_k is the proportion of samples belonging to class k. The final action label $\hat{y}_{action}$ is determined by majority voting across all trees: $\hat{y}_{action} = \arg\max_k \sum_{b=1}^{B} \mathbb{I}\big(\text{CART}_b(\mathbf{f}) = k\big)$, where $\mathbf{f}$ denotes the feature vector and $\mathbb{I}(\cdot)$ is the indicator function.

Stage 2—Action Power Regression. For each flight action, an independent Random Forest regressor is trained using the augmented feature set: action label, wind speed, wind angle, UAV position (gps_x, gps_y), altitude (gps_z), ground speed (v_x, v_y, v_z), battery voltage V_{bat}, and battery current I_{bat}. The output of the five regressors provides the action-specific energy or power: E_{ascend} and E_{descend} (integrated energy in J), E_{turn} (turning energy in J), P_{hover} and P_{horiz} (instantaneous power in W), under varying conditions.

Each CART regression tree is trained to minimize the mean squared error: $\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$, where N is the number of training samples. The prediction of the b-th tree for a leaf node with N_b samples is the mean target value $\bar{y}_b = \frac{1}{N_b}\sum_{i\in\mathrm{leaf}_b} y_i$. The final Random Forest prediction is the average over all B trees: $\hat{y}_{RF} = \frac{1}{B}\sum_{b=1}^{B}\bar{y}_b$.

The RF model is trained and validated on real-world flight logs. All random forest regressors are implemented with scikit-learn RandomForestRegressor (v1.7.2). The following hyperparameters are held fixed across all models at their package default values or explicitly set for reproducibility and parallel execution:

- *criterion = 'squared_error'* (mean squared error);
- *max_features = 1.0* (all features considered at each split);
- *bootstrap = True* (bootstrap sampling);
- *min_impurity_decrease = 0.0*;
- *min_weight_fraction_leaf = 0.0*;
- *max_leaf_nodes = None*;
- *random_state = 42* (explicitly set for reproducibility);
- *n_jobs = −1* (parallel execution using all cores).

Four hyperparameters are optimized per model via 5-fold GridSearchCV (*scoring = 'neg_mean_squared_error'*) over the grid: $n\_estimators \in \{50, 100, 200\}$, $max\_depth \in \{10, 20, \text{None}\}$, $min\_samples\_split \in \{5, 10\}$, and $min\_samples\_leaf \in \{2, 4\}$. This yields $3 \times 3 \times 2 \times 2 = 36$ hyperparameter combinations per model; with 5-fold cross-validation, each model requires $36 \times 5 = 180$ fitted estimators. The dataset is split into 80% training and 20% testing by flight missions. Table 3 reports the optimized (n_t, d, mss, msl) for each trained model.

Table 3: Random forest action-specific energy model performance.

Regime	UAV	Samples	RMSE	MAE	R^2	Optimized (n_t, d, mss, msl)
Ascending	G	92	273.28	208.87	0.977	(100, 10, 5, 2)
Ascending	R	1305	231.38	132.84	0.917	(200, 20, 5, 2)
Ascending	Y	211	252.60	160.84	0.969	(100, 10, 5, 2)
Descending	G	106	360.65	254.49	0.991	(100, None, 5, 4)
Descending	R	1028	311.28	133.58	0.955	(100, None, 5, 4)
Descending	Y	231	447.15	274.89	0.963	(200, 10, 10, 2)
Horizontal	G	66,514	6.93	4.13	0.944	(200, None, 5, 2)
Horizontal	R	50,905	5.35	3.21	0.972	(200, None, 5, 2)
Horizontal	Y	140,194	4.60	2.95	0.961	(200, None, 5, 2)
Hovering	G	7770	22.90	8.67	0.917	(200, 20, 5, 2)
Hovering	R	3712	25.92	14.04	0.940	(100, None, 5, 2)
Hovering	Y	6555	26.30	13.59	0.943	(200, None, 5, 2)
Turning	G	5029	149.76	62.93	0.927	(200, None, 5, 2)
Turning	R	3989	778.24	178.30	0.382	(200, 20, 5, 4)
Turning	Y	6315	441.68	97.444	0.562	(100, 20, 5, 4)

The Turning R/Y models are excluded for their best R^2 remained below 0.6 across all configurations, indicating insufficient predictive reliability for energy estimation. Therefore, the UavG turning model is adopted as the universal turning-energy proxy, with post-hoc adjustment coefficients applied to account for inter-type differences. Specifically, the turning-energy estimate for a UAV of type $t \in \{\mathrm{R}, \mathrm{Y}\}$ is computed as $\hat{E}^{(t)}_{\mathrm{turn}} = \eta_t \cdot \hat{E}^{(\mathrm{G})}_{\mathrm{turn}}$, where $\hat{E}^{(\mathrm{G})}_{\mathrm{turn}}$ is the output of the UavG turning model and η_t is the type-specific adjustment coefficient. The coefficients are calibrated from a small hold-out set of R/Y flight segments: $\eta_{\mathrm{R}} = 1.15$ and $\eta_{\mathrm{Y}} = 0.87$. This approximation approach does not explicitly account for the distinct aerodynamic or powertrain characteristics of R-type and Y-type UAVs. As a result, the turning-energy estimates for R- and Y-type tasks carry a larger model-induced uncertainty compared with those for G-type tasks. This simplification is a limitation of the current task-energy estimation framework.

Diagnostic Analysis. To validate model reliability, each regressor undergoes residual analysis, learning curve inspection, and feature importance ranking. Fig. 11 exemplifies the regression model of horizontal uniform action for G-type UAV ($R^2 = 0.944$, $n = 66,514$): (a) the actual-vs-predicted scatter shows tight alignment along the ideal line; (b) residuals are randomly distributed around zero without systematic bias; (c) the residual histogram approximates a Gaussian distribution; (d) the learning curve confirms convergence without overfitting; (e) feature importance reveals that velocity and payload dominate power prediction, consistent with aerodynamic drag and weight-induced energy power.

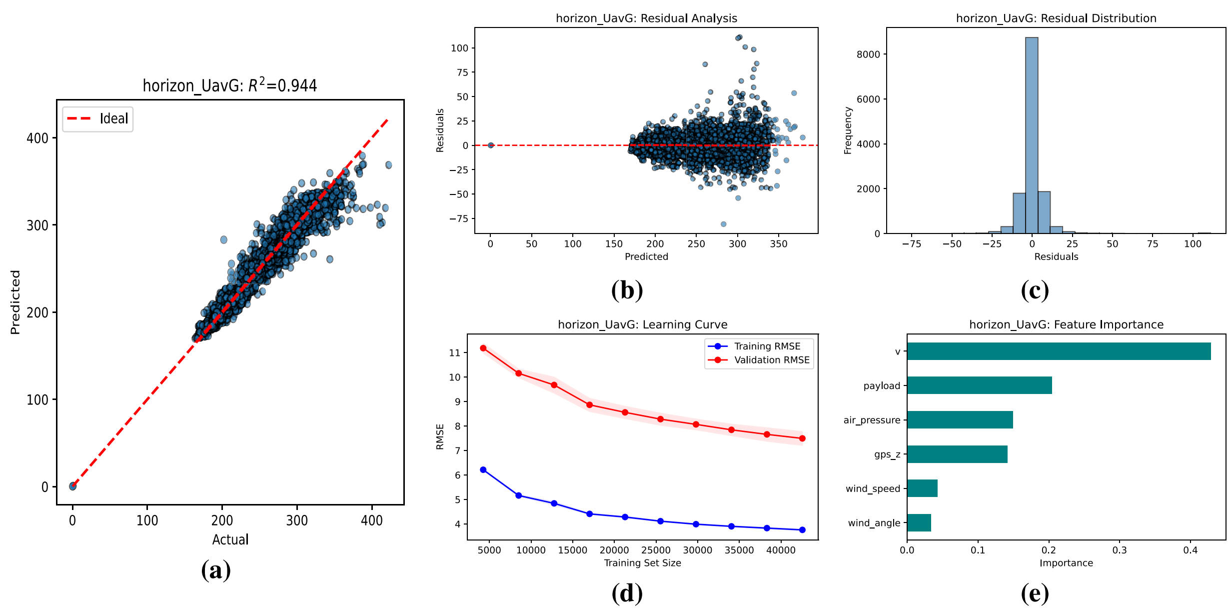


Figure 11: Diagnostic analysis of the horizontal uniform energy model (UavG, unified payload). (**a**) Actual vs. predicted power. (**b**) Residuals vs. predicted. (**c**) Residual distribution. (**d**) Learning curve. (**e**) Feature importance.

Table A1 reports the residual diagnostics for all 15 trained models. For the retained models ($R^2 > 0.9$), the residual means are generally bounded within ±6% of the RMSE, with the exception of the Ascending-G model (34%), which is primarily attributable to its limited sample size $n = 92$. The Durbin-Watson (D-W) statistics for these models hover near 2.0 (1.88–2.04), indicating a negligible lag-1 autocorrelation in residuals. Positive skewness (0.5–4.4) and excess kurtosis (3.5–63.9), such heavy-tailed behavior is consistent with long-tail power-consumption physics (gust-induced transient loads, actuator saturation) and do not invalidate the models for aggregate task-energy budgeting. Validation curves were constructed for all 15 models. For models with satisfactory baseline performance (test $\mathrm{R}^2 > 0.60,\ \mathrm{n} = 13$), the curves exhibit clear plateaus (Table A2).

Stratified Validation. To assess generalization across payloads and flight conditions, we perform stratified testing. Fig. 12 shows that: (a) the ascending model (UavR) maintains $R^2 > 0.88$ across all payload configurations (0–400 g); (b) the horizontal model (UavR) achieves $R^2 > 0.96$ across low, medium, and high altitudes; (c) the hovering model (UavR) maintains $R^2 > 0.93$ across low, medium, and high wind speeds. Identical stratified tests are performed for every action–UAV combination. Table A3 reports the R^2 by stratum.

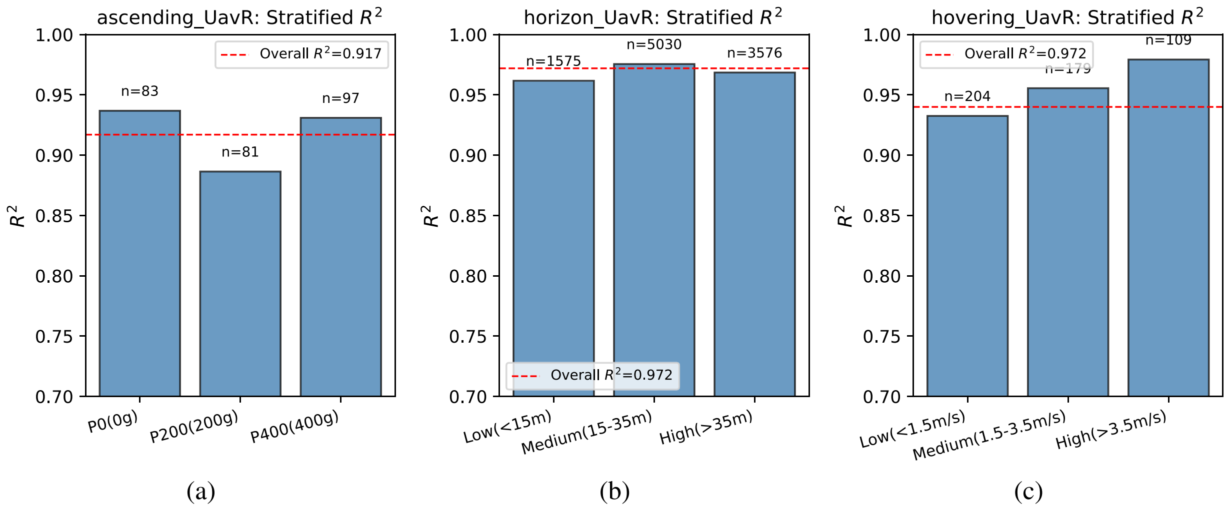


Figure 12: Stratified validation results. (**a**) Ascending model (UavR) across payload configurations. (**b**) Horizontal model (UavR) across altitude levels. (**c**) Hovering model (UavR) across wind speed levels. Red dashed lines indicate overall R^2.

The regression models enable the computation of action-specific energy consumption under varying flight conditions (speed, altitude) and payload weights, as illustrated in Fig. 13.

5.4.3 Sweep Capability Aggregation

This section formulates a path-optimization problem that, given the Imaging Capability Cap_i^{img} and Flight Capability Cap_i^{flt}, maximizes the effective coverage area of a single UAV, thereby deriving its Sweep Capability $Cap_i^{\text{sweep}} = \langle \alpha_i, \beta_i \rangle$.

The coverage benefit α_i and total energy cost β_i are intrinsically linked to the UAV's flight path. To characterize the performance potential of a UAV–sensor pair, we define its Sweep Capability as the maximum area it can cover given its current energy state. The trajectory that yields this maximum is termed the **Optimal Sweep Path**.

Trajectory Strategy and Pattern Selection. A common approach to path planning involves decomposing the target area into cells that must be visited [50]. While this problem is NP-hard in general, flying in straight-line patterns offers an efficient means to cover terrain. To reduce the infinite search space of potential trajectories and simplify image processing, we adopt the Boustrophedon (lawnmower) pattern—a sequence of straight, parallel flight lanes connected by turns [51].

To maximize coverage, we adopt the methodology from [52]. We orient the flight lanes parallel to the longest edge of the target area to minimize the number of energy-intensive turns [53].

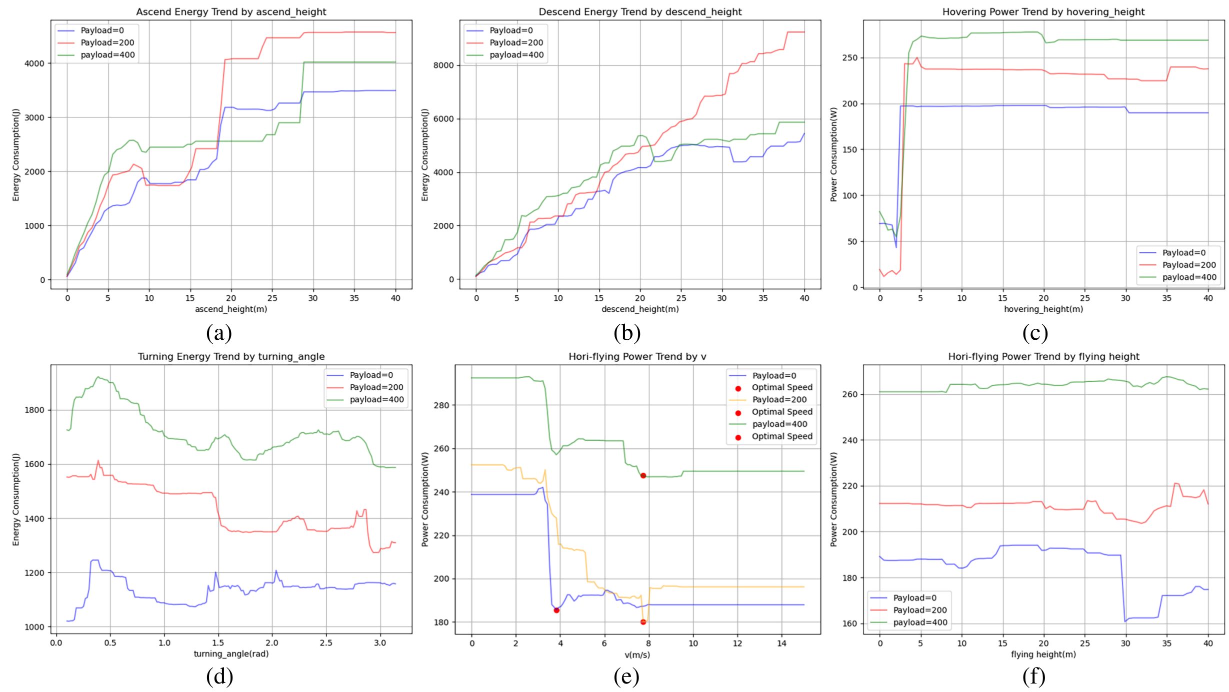


Figure 13: Energy consumption for different flight actions. (**a**) Ascending energy at different heights. (**b**) Descending energy at different heights. (**c**) Hovering energy at different heights. (**d**) Turning energy at varying angles. (**e**) Horizontal flight energy at varying velocities. (**f**) Horizontal flight energy at different heights.

As established in Section 5.4.1, the Imaging Capability for the required GSD yields an effective single-frame coverage footprint of dimensions $L_h^{'img} \cdot L_v^{'img}$. A target area of size $L_{\text{area}} \times W_{\text{area}}$ is partitioned into $n = \lceil W_{\text{area}}/L_h^{'img} \rceil$ lanes, denoted R_k $(k = 1, \ldots, n)$, each with a length $d_k = L_{\text{area}}$. The optimization then determines how many such lanes a UAV can traverse before exhausting its battery.

Path-Optimization Formulation. The Sweep Capability is computed by solving the following lane-based optimization problem:

$$\max_{z_k} \quad \alpha_i = \sum_{k=1}^{n} L_{\text{area}} \cdot L_h^{'img} \cdot z_k \tag{14a}$$

$$\text{s.t.} \quad \sum_{k=1}^{n} \left(E_{\text{cam},k} + E_{\text{flt},k} \right) \cdot z_k + E_{\text{reach}} + E_{\text{return}} \le Er^{flt} - E_{\text{safe}}, \tag{14b}$$

$$E_{\text{flt},k} = E_{\text{horiz}}(d_k) + E_{\text{turn}}(R_{k-1}, R_k) + E_{\text{turn}}(R_k, R_{k+1}) + E_{\text{horiz}}(d_{turn}), \tag{14c}$$

$$E_{\text{reach}} = \begin{cases} E_{\text{ascend}}(h_i) + E_{\text{horiz}}(d_{\text{reach}}), & \text{if starting from base,} \\ E_{\text{horiz}}(d_{\text{reach}}), & \text{otherwise,} \end{cases} \tag{14d}$$

$$E_{\text{return}} = E_{\text{horiz}}(d_{\text{return}}) + E_{\text{descend}}(h_i), \tag{14e}$$

$$h_i \le h^{img}, \quad v \le Sm^{flt}, \tag{14f}$$

$$z_k \in \{0, 1\}, \quad k = 1, \ldots, n. \tag{14g}$$

where $z_k \in \{0,1\}$ indicates if lane k is covered. $E_{\text{cam},k} = t_k \cdot P^{img}$ is the camera operation energy per lane.

Computational Assumptions and Refinement. To ensure the capability metrics α_i and β_i are robust, conservative nominal estimates, apart from the nominal assumptions in Section 5.1.2, the following constraints are applied:

- *Spatial Approximations:* Both the global area and coverage sub-regions are treated as rectangles. This ensures that the sweep path is regular and that the coverage area is exactly the product of the lane width and the total path length, eliminating overlap or gap ambiguity at the nominal design stage.
- *Energy Budgeting:* Constraint (14b) limits total energy consumption by the residual energy Er^{flt} minus a safety margin E_{safe}. The total energy comprises camera operation energy $E_{\text{cam},k}$, flight energy along each lane $E_{\text{flt},k}$, the reach energy from UAV's current position to target area arrival point E_{reach} and the return energy from target leaving point to base. Generally, the energy required for communication can be considered negligible compared to propulsion energy, given that data collected by the UAV is processed post-flight [54].
- *Center-point approximation for reach/return energy:* To mitigate position bias, the transit distances d_{return} and d_{reach} are estimated using the geometric center of the target region as a representative arrival/departure point.
- *In-trip energy:* The in-trip flight energy $E_{\text{flt},k}$ in (14c) includes horizontal flight along lane R_k $E_{\text{horiz}}(d_k)$, turning energy at its endpoints $E_{\text{turn}}(R_{k-1}, R_k)$ $E_{\text{turn}}(R_k, R_{k+1})$ and the horizontal flight during turns $E_{\text{horiz}}(d_{turn})$.
- *Transit energy:* The reach energy (14d) depends on whether the UAV starts from the base (requiring ascent) or is already airborne. If the UAV starts from the base, it includes ascending energy, $E_{ascend}(h_i)$, where h_i is the flight altitude, and horizontal flight energy to the target lane $E_{horiz}(d_{Ri})$. Otherwise, only horizontal flight energy is required. The return energy (14e) includes horizontal flight to the landing point at the base and descent.
- *Optimal Motion Parameters:* Constraint (14f) limits altitude h_i and speed v. The flight velocity is optimized based on the power-regression models from Section 5.4.2 to minimize energy consumption per meter.

Once the optimal path is solved, the resulting coverage α_i (constrained to an integer multiple of the lane width $(W_i = n \cdot w_i,\ n \in \mathbb{Z}^+)$), and the total energy expenditure β_i $(= \sum_k (E_{\text{cam},k} + E_{\text{flt},k}) + E_{\text{reach}} + E_{\text{return}})$ are stored as the Sweep Capability attributes. This aggregation transforms the complex interplay of motion dynamics and camera intrinsics into a compact, task-level descriptor, enabling efficient selection in the subsequent group-optimization phase.

Furthermore, α_i is the maximum net effective coverage that already excludes internal frame-to-frame overlap along the UAV's own trajectory and its dimension is constrained to integer multiples of the lane width, ensuring that the number of turns and the straight-line distance are predetermined and integer-valued. Although heterogeneous cameras that have different intrinsics (focal length, pixel size) must operate at different altitudes to satisfy the same GSD, yielding heterogeneous lane widths across UAVs, only the lane count that fits within the UAV's energy budget matters within the scope of single-UAV capability computation. These pre-computed attributes are designed to be region-invariant to facilitate the subsequent group selection phase. Although the specific sub-region assigned to a UAV during deployment may vary in its exact coordinates, the deviation between pre-computed capability and actual task execution remains minimal.

5.5 Sensing UAV Selection Using Sweep Capabilities

5.5.1 OGU-Greedy Selection

The group selection problem is formulated as a scalar minimum-cost coverage formulation. Given a target area Ω and a set of candidate UAVs $U = \{u_1, \ldots, u_n\}$, each UAV u_i is characterized by a pre-computed capability tuple $\langle \alpha_i, \beta_i \rangle$, where α_i is the effective coverage area and β_i is the total energy cost of the in-strip sweep and the nominal reach/return transit under idealized lane-sweep conditions.

The objective is to select a minimum-cost subset such that the aggregate coverage capability meets the target Ω. The scalar minimum-cost coverage problem is formulated as the following 0–1 Integer Program. Let U be the set of candidate UAVs, and x_i be the decision variable indicating whether u_i is selected:

$$\min_{x_i} \quad \sum_{u_i \in U} \beta_i \cdot x_i \tag{15}$$

$$\text{s.t.} \quad \sum_{u_i \in U} \alpha_i \cdot x_i \geq \Omega, \tag{16}$$

$$\beta_i \cdot x_i \leq E_i^{\max}, \quad \forall u_i \in U, \tag{17}$$

$$D_i \cdot x_i \leq D_i^{\max}, \quad \forall u_i \in U, \tag{18}$$

$$x_i \in \{0, 1\}, \quad \forall u_i \in U, \tag{19}$$

To solve the scalar minimum-cost coverage formulation efficiently, we propose OGU-greedy (Algorithm 1), a **cost-efficiency ranking heuristic**. The algorithm employs a cost-efficiency ranking heuristic: UAVs are sorted by the pre-computed coverage-to-energy ratio $\rho_i = \alpha_i / \beta_i$ in descending order, and iteratively selected until the cumulative nominal coverage meets the target. Because the coverage contributions are scalar values added linearly, OGU-greedy does not rely on diminishing marginal returns as in submodular set-covering; the ratio $\rho_i = \alpha_i / \beta_i$ therefore serves as a static cost-efficiency ranking metric rather than a dynamic marginal-utility score. The practical solution quality is validated empirically in Section 6.5, where OGU-greedy achieves objective values within **5% of the exact optimum** obtained by MILP and Branch-and-Bound methods.

Algorithm 1: OGU-greedy: cost-efficiency ranking heuristic for UAV group selection

Require: Target area Ω; set of candidate UAVs $U = \{u_1, \ldots, u_n\}$ with pre-computed Sweep Capabilities $\{Cap_i^{\text{sweep}} = \langle \alpha_i, \beta_i \rangle\}_{i=1}^{n}$.
Ensure: Selected UAV subgroup $\mathcal{U}_{\text{sel}} \subseteq U$.

Compute efficiency ratio $\rho_i \leftarrow \alpha_i / \beta_i$ for each $u_i \in U$.
Sort U by ρ_i in descending order.
$S_{\text{cov}} \leftarrow 0$ {Cumulative coverage}
$\mathcal{U}_{\text{sel}} \leftarrow \varnothing$ {Selected UAVs}
$E_{\text{tot}} \leftarrow 0$ {Total energy cost}
for u_i in sorted U **do**
  $S_{\text{cov}} \leftarrow S_{\text{cov}} + \alpha_i$
  $\mathcal{U}_{\text{sel}} \leftarrow \mathcal{U}_{\text{sel}} \cup \{u_i\}$
  $E_{\text{tot}} \leftarrow E_{\text{tot}} + \beta_i$
  if $S_{\text{cov}} \geq \Omega$ **then**
    break
  end if

(Continued)

Algorithm 1 (continued)

13: **end for**
14:
15: **return** $\mathcal{U}_{\text{sel}}$

5.5.2 Constructive Geometric Partitioning and Deployment

Lane-based area decomposition and task assignment. After the greedy selection returns the ordered subgroup $\mathcal{U}_{\text{sel}} = \{u_1, u_2, \ldots, u_k\}$, the central planner executes a **constructive geometric partitioning procedure** to carve out sub-regions and assign sweep paths. This procedure is constructive in the sense that it yields an explicit, executable deployment layout under nominal operating assumptions and Sweep Capability constraints.

Sequential strip-peeling partition. The target area Ω is approximated as a rectangular region of dimensions $L_{area} \times W_{area}$, where the x-axis is aligned with the longest edge (the sweep direction). The planner peels off strips sequentially along the y-axis (the short edge) in the order of the selected UAVs. The area is partitioned into non-overlapping strips $\{\Omega_1, \ldots, \Omega_i\}$ along the y-axis:

1. **Nominal strip geometry.** For each UAV u_i $(i = 1, \ldots, k-1)$, the assigned strip Ω_i has the same length as the target area (L_{area}) and a width W_i that is an integer multiple of the UAV's ground footprint:

$$W_i = n_i \cdot w_i, \quad n_i \in \mathbb{Z}^+, \tag{20}$$

where the integer n_i is chosen such that the strip area matches the UAV's pre-computed coverage capability $\text{Area}(\Omega_i) = L_{area} \times W_i = \alpha_i$. Because the sweep capability α_i for the u_i was computed under the same lane width $w_i = L'^{img}_{h,i}$ and the same sweep length L_{area}, the strip Ω_i is geometrically identical to the nominal sub-region used during the α_i and β_i estimation.
2. **Trajectory translation.** The pre-computed Boustrophedon sweep path for UAV u_i is translated rigidly from the nominal reference frame to the actual strip Ω_i. The lane count, lane width, turning count, and straight-line distance are preserved exactly; only the spatial origin of the path changes. Consequently, under the nominal assumptions stated above, the *in-strip* flight trajectory of u_i is a pure geometric translation of its nominal trajectory, and the in-strip energy consumption is still the subcomponent of the pre-computed capability cost β_i.
3. **Sequential peeling.** The strips are peeled off sequentially along the y-axis without overlap: $\Omega_i \cap \Omega_j = \varnothing, \quad \forall i \neq j$. The i-th strip occupies the interval $[y_{i-1}, y_{i-1} + W_i)$ along the short edge, where $y_0 = 0$ and $y_i = \sum_{j=1}^{i} W_j$.
4. **Closing the Partition (The Last Strip):** For the last UAV u_k, the remaining uncovered width is $W_{\text{rem}} = W_{area} - \sum_{j=1}^{k-1} W_j$. Since the greedy selection ensures $\sum_{j=1}^{k} \alpha_j \geq \Omega = L_{area} \times W_{area}$, and each $W_j = \alpha_j / L_{area}$, u_k is guaranteed to have sufficient capability to cover the residual strip $\Omega_k = L_{area} \times W_{\text{rem}}$. Specifically, u_k traverses $n_k = \lceil W_{\text{rem}} / L'^{img}_{h,k} \rceil$ lanes to cover the residual width; any surplus capability beyond the actual residual area is treated as an operational margin.

Reach and return path concatenation. For each UAV, *reach* (base → strip entry) and *return* (strip exit → base) are embedded in the pre-computed capability cost β_i. The reach and return distances are estimated using the geometric center of the assigned strip as the representative operating point. This center-point approximation is intended to minimize the maximum deviation across the strip and yields a nominally conservative estimate for the actual transit energy under the stated idealized conditions. The total capability cost β_i, which combines the pre-computed in-strip sweep cost with this transit estimate, is therefore nominally representative of the actual deployment energy under the same assumptions.

Residual deviations and runtime correction. Because the constructive partitioning relies on nominal estimates, coverage gaps may arise from wind drift, positioning errors, or energy depletion before the UAV completes its strip. After each UAV returns to base, its actual covered area is compared against the nominal α_i. If a gap is detected, OGU-greedy is re-executed on the remaining available UAVs with the updated target area Ω_{gap}. This online replanning does not disturb ongoing operations and compensates for residual deviations that are not captured by the pre-computed nominal model.

This formulation decouples the combinatorial selection from detailed UAV-sensor parameters and geometric path planning: α_i, β_i represent nominal (pre-computed) estimates for a UAV operating under the idealized lane-sweep conditions, while actual task costs are monitored and corrected during execution.

6 Experimental Validation

This section presents simulation results to validate the effectiveness of OGU for the area coverage task.

6.1 Experimental Setup

All simulations are conducted on a desktop computer with an Intel Core i5-9400F 2.90 GHz CPU and 16 GB RAM, running Windows 10.

The experimental setup comprises a heterogeneous fleet of UAVs and camera payloads. Five camera types and five UAV platform types are considered, as summarized in Tables 4 and 5, with parameters are based on manufacturer datasheets. The flight dynamics and energy consumption are modeled using a data-driven Random Forest regressor trained on real-world flight logs. Both homogeneous and heterogeneous UAV–camera pairings are evaluated. Six target areas of varying dimensions are defined in Table 6 for testing. The default scenario uses a target area of 500×400 m^2 (20×10^4 m^2), a GSD of 0.010 m, and a candidate pool of 25 UAVs unless otherwise stated.

Table 4: Camera specifications.

ID	Resolution (pixels)	Element Size (um)	Frame Rate (fps)	Image Size (MB)	Power (W)	Weight (g)
C1	812 × 620	6.9	282	0.9	4.1	65
C2	1280 × 720	5.3	61	2.6	3.0	75
C3	1920 × 1080	3.45	70.8	6	3.6	62
C4	2560 × 1440	2.2	14	11	3.4	40
C5	3840 × 2160	1.67	10.4	24	3.8	47

Table 5: UAV specifications.

ID	Battery (mAh)	Voltage (V)	Max Speed (m/s)	Battery Left (%)
U1	2590	7	12	60%–80%
U2	2800	8	13	60%–80%
U3	3000	11	15	60%–80%
U4	5000	15.4	20	60%–80%
U5	5500	15	17	60%–80%

Table 6: Area specifications.

AreaID	Length (m)	Width (m)	Area (m^2)
Area1	1000	100	10×10^4
Area2	500	400	20×10^4
Area3	800	250	20×10^4
Area4	1000	300	30×10^4
Area5	1000	400	40×10^4
Area6	900	500	45×10^4

We compare the capability-based approach (OGU) with other sensor selection methods to validate its effectiveness and efficiency. The comparison metrics used in this evaluation are as follows, which are the Y-axis in figures:

1. **Number of selected UAVs:** the cardinality of the selected subgroup. A smaller number reduces operational complexity, communication overhead, and collision risk.
2. **Area coverage ratio:** the ratio of actual covered area to the required target area Ω. A value of 1 indicates exact coverage; values below 1 denote under-coverage, while values significantly above 1 indicate over-coverage.
3. **Energy ratio:** the ratio of energy consumed for effective coverage to the total energy expended. Higher values indicate more efficient energy utilization.
4. **Energy efficiency:** the ratio of swept area to total energy consumption (m^2/J), reflecting the coverage output per unit energy.
5. **Selection time:** the computational time for the selection algorithm to output the UAV subgroup.
6. **Task completion time:** the wall-clock time from UAV deployment to full area coverage, reflecting the task duration.

6.2 Comparison with Selectivity-Based Method (Opti-U)

We first compare OGU, Selectivity, and Random under the default configuration (20×10^4 m^2, 25 candidates, GSD = 0.010 m).

- **OGU:** the proposed capability-oriented group selection method with pre-computed capability aggregation.
- **Selectivity (Opti-U):** the state-of-the-art selectivity-based single-UAV selection approach [7], extended to iterative multi-UAV selection by repeatedly choosing the highest-scoring candidate.
- **Random:** a baseline that stochastically selects UAVs from the candidate pool. To ensure an equitable comparison, the number of selected UAVs is set to be identical to that determined by the OGU method.

As shown in Table 7, OGU achieves the highest mean energy efficiency (0.458 ± 0.033) and area coverage ratio (1.121 ± 0.079), while selecting only 4.9 ± 0.5 UAVs—less than half the number selected by Selectivity (10.3 ± 2.2). Random performs comparably to OGU in UAV count (4.9 ± 0.5) but falls short in area coverage (0.592 ± 0.054) and energy efficiency (0.264 ± 0.024). Wilcoxon signed-rank tests ($n = 30$) confirm that OGU's advantages over both baselines are statistically noticeable for all metrics ($p < 0.001$) except UAV count vs. Random, which is constrained to match OGU by design ($p = 1.0$).

The most striking difference lies in task time: OGU completes the mission in 729 ± 75 s, whereas Selectivity requires 5212 ± 615 s—nearly 7× longer. This gap stems from their fundamentally different deployment paradigms: OGU performs simultaneous group selection, enabling parallel UAV deployment, while Selectivity iteratively selects one UAV at a time, forcing sequential execution. Random also benefits from parallel deployment (all selected UAVs fly concurrently), but its lack of capability-aware matching leads to suboptimal coverage and energy utilization.

Table 7: Baseline comparison (20×10^4 m^2, 25 candidates, GSD = 0.010 m, Mean ± SD, 95% CI, n = 50).

Metric	OGU	Selectivity	Random
Energy Ratio	0.715 ± 0.034 [0.705, 0.724]	0.592 ± 0.072 [0.571, 0.612]	0.642 ± 0.046 [0.629, 0.655]
Area Ratio	1.121 ± 0.079 [1.098, 1.143]	1.008 ± 0.000 [1.008, 1.008]	0.592 ± 0.054 [0.576, 0.607]
Energy Eff.	0.458 ± 0.033 [0.448, 0.467]	0.198 ± 0.029 [0.190, 0.207]	0.264 ± 0.024 [0.257, 0.271]
Task Time (s)	729 ± 75 [707, 750]	5212 ± 615 [5037, 5386]	667 ± 56 [651, 683]
Sel. Time (s)	$<10^{-4}$	0.022 ± 0.003	$<10^{-4}$
Prep. Time (s)	2.99 ± 0.02 [2.98, 2.99]	0.12 ± 0.00 [0.12, 0.12]	0.54 ± 0.05 [0.52, 0.55]
UAV Count	4.9 ± 0.5 [4.8, 5.0]	10.3 ± 2.2 [9.6, 10.9]	4.9 ± 0.5 [4.8, 5.0]

Note: Wilcoxon signed-rank: OGU vs. Sel. $p < 0.001$; OGU vs. Rnd. $p < 0.001$ (UAV Count $p = 1.0$, by design).

6.2.1 Sensitivity Analysis

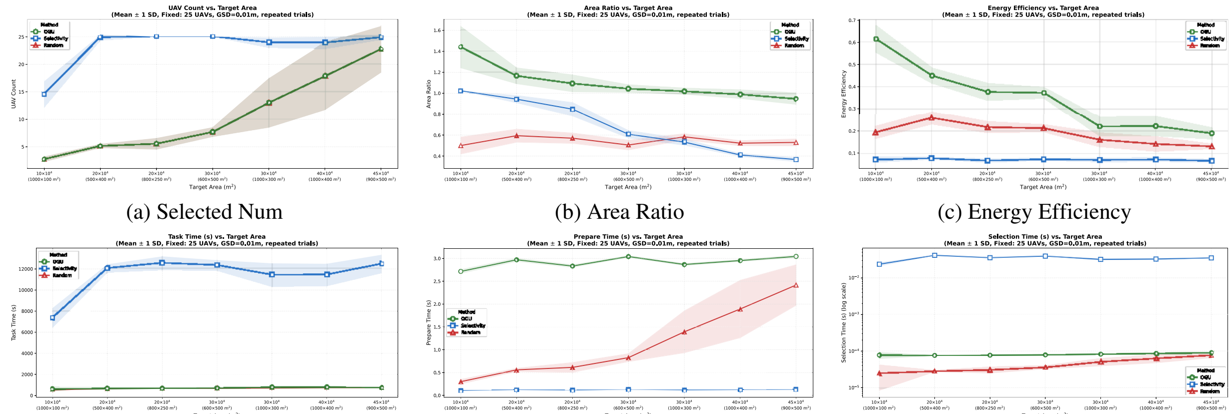
a. Sensitivity to Target Area and Shape

To systematically evaluate OGU's behavior under varying task parameters, we conduct four sensitivity analyses. In each dimension, the non-varied parameters are fixed to the default configuration.

We vary the target area from 10×10^4 m^2 to 45×10^4 m^2 while keeping the candidate fleet size at 25. Two areas of identical size but different aspect ratios are included: 20×10^4 m^2 (500 m × 400 m vs. 800 m × 250 m) and 30×10^4 m^2 (600 m × 500 m vs. 1000 m × 300 m).

As shown in Fig. 14a–c, OGU consistently outperforms both baselines in energy efficiency and area coverage across all scales. For the smallest configuration (10×10^4 m^2), OGU achieves a mean area ratio of 1.440 ± 0.198 (95% CI: [1.366, 1.514]), exceeding Selectivity's exact-coverage value (1.023 ± 0.000, $p < 0.001$) and Random's under-coverage (0.501 ± 0.081, $p < 0.001$). At the largest scale (45×10^4 m^2), OGU's mean area ratio drops to 0.949 ± 0.056 but remains higher than Selectivity (0.367 ± 0.020, $p < 0.001$) and Random (0.531 ± 0.034, $p < 0.001$). Notably, Selectivity's area ratio exhibits zero variance (1.023 ± 0.000 for 10×10^4 m^2) because its fixed sweep path guarantees exact coverage, whereas OGU's over-coverage reflects its capability-aware redundancy. Full statistical validation across all seven area configurations is reported in Table A4.

As shown in Fig. 14d–e, selectivity exhibits a dramatic increase in task time with area—reaching 12,307 ± 497 s for 30×10^4 m^2 and 12,454 ± 860 s for 45×10^4 m^2—because its sequential single-UAV selection scales poorly with task scale. In contrast, OGU's task time remains stable (679 ± 195 to 864 ± 6 s) and is significantly shorter than Selectivity's at all scales ($p < 0.001$, Wilcoxon signed-rank). OGU achieves a selection time on the order of ten microseconds ($<10^{-4}$ s), whereas selectivity-based Opti-U requires approximately 0.023–0.041 s. This substantial gap stems from their fundamentally different selection paradigms: OGU performs a single, simultaneous group selection using pre-computed capability attributes, while selectivity-based Opti-U must repeatedly execute its single-UAV selection routine throughout the task—re-initiating whenever a UAV completes its assignment.



(a) Selected Num (b) Area Ratio (c) Energy Efficiency

(d) Task Time (e) Prepare Time (f) Selection Time

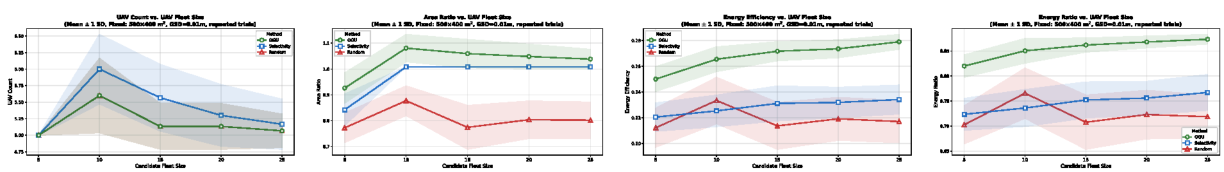
Figure 14: Sensitivity to target area and shape.

Fig. 14f compares the preparation time, defined as the time required to compute the selection metrics for each candidate UAV. OGU requires 2.72–3.04 s, longer than Selectivity (0.11–0.14 s, $p < 0.001$) because of the computational overhead of capability aggregation, including path optimization and Random Forest inference. This pre-computation, however, is amortized across the task: it enables near-instantaneous group selection at runtime. Statistical validation at representative settings is reported in Table A4. Shaded bands denote ±1 SD over $n = 30$ repeated trials.

b. Sensitivity to UAV Fleet Size

We vary the candidate fleet size from 5 to 25 while fixing the target area at 20×10^4 m^2.

As shown in Fig. 15, OGU's energy efficiency improves monotonically with fleet size, from 0.250 ± 0.010 (95% CI: $[0.247, 0.254]$) at 5 candidates to 0.279 ± 0.006 ($[0.277, 0.281]$) at 25, with statistically noticeable advantages over both Selectivity and Random at all scales ($p < 0.001$, Wilcoxon signed-rank, $n = 30$). This trend reflects the increased diversity of the candidate pool: with more UAVs available, OGU can identify better-matched subgroups that deliver higher coverage per unit energy.



(a) Selected Num (b) Area Ratio (c) Energy Efficiency (d) Energy Ratio

Figure 15: Sensitivity to UAV fleet size.

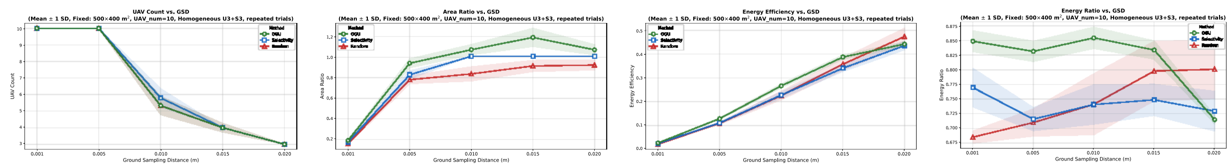
Selectivity remains lower than OGU's at all scales ($p < 0.001$), because its static scoring function cannot exploit the heterogeneous capability space for subgroup optimization. Random shows no clear monotonic trend (0.212 ± 0.015 to 0.217 ± 0.016), fluctuating because stochastic selection lacks the structural awareness to leverage increased candidate diversity.

A key observation is that OGU deploys approximately 5 UAVs regardless of fleet size, while maintaining higher area ratios than both baselines. This demonstrates that OGU's advantage lies not in reducing the deployed count, but in selecting more capable subgroups from a larger pool. Full statistical validation across all fleet sizes is reported in Table A5.

c. Sensitivity to Ground Sampling Distance (GSD)

We vary the GSD from 0.001 m (high resolution) to 0.020 m (low resolution) under fixed area (20×10^4 m^2) and 10 homogeneous UAVs (U3 + S3).

As shown in Fig. 16, all three methods show improved energy efficiency as GSD coarsens from 0.001 to 0.020 m, because coarser resolution reduces the required flight path length. At the finest resolution (0.001 m), all methods suffer from extremely low energy efficiency (<0.03) due to the excessive flight path required for high-resolution coverage. OGU maintains a observed advantage even in this regime (0.026 ± 0.001 vs. Selectivity 0.024 ± 0.001, $p < 0.001$; vs. Random 0.021 ± 0.000, $p < 0.001$).



(a) Selected Num (b) Area Ratio (c) Energy Efficiency (d) Energy Ratio

Figure 16: Sensitivity to GSD.

At 0.01 m, OGU's energy efficiency reaches 0.266 ± 0.008, exceeding both Selectivity (0.228 ± 0.013, $p < 0.001$) and Random (0.226 ± 0.018, $p < 0.001$). At 0.015 m, OGU's area ratio peaks at 1.191 ± 0.088, indicating robust over-coverage compared to Selectivity's exact-coverage behavior (1.008 ± 0.000) and Random's under-coverage (0.914 ± 0.054).

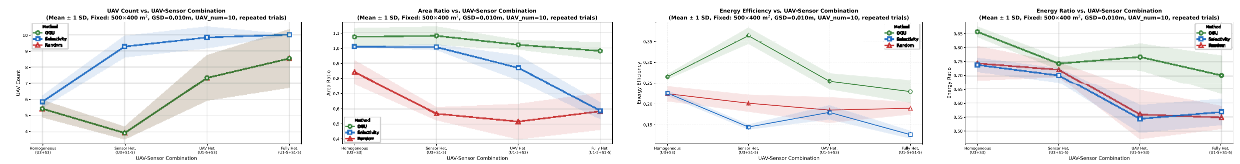
At GSD = 0.020 m, Random achieves the highest energy efficiency (0.474 ± 0.035), slightly exceeding OGU (0.442 ± 0.010) and Selectivity (0.436 ± 0.017). This behavior is attributed to the fact that OGU does not perform path truncation in the current implementation. OGU assigns flight regions based on UAV sweep capability without clipping paths that exceed the target boundary.

Even when the candidate pool is restricted to a homogeneous fleet (U3 + S3), OGU achieves better energy efficiency and coverage ratio than Selectivity. OGU's capability-constrained selection still exploits task-level differences in UAV positioning, remaining energy, and expected coverage effectiveness—factors that Selectivity's static, sensor-centric scoring function does not account for. Full statistical validation across all GSD configurations is reported in Table A6.

d. Sensitivity to UAV–Sensor Heterogeneity

We evaluate four UAV-sensor combination scenarios under fixed conditions (20×10^4 m^2, 0.010 m GSD, 10 candidates): (1) *Homogeneous*: U3 + S3; (2) *Sensor Heterogeneous*: U3 + S1–5; (3) *UAV Heterogeneous*: U1–5 + S3; (4) *Fully Heterogeneous*: U1–5 + S1–5.

As shown in Fig. 17, OGU achieves the highest mean energy efficiency under the Sensor Heterogeneous configuration (0.364 ± 0.019, 95% CI: [0.357, 0.371]), outperforming both the Homogeneous (0.265 ± 0.006, $p < 0.001$) and UAV Heterogeneous (0.254 ± 0.019, $p < 0.001$) configurations. This suggests that sensor diversity contributes more to task performance than UAV platform diversity in area coverage scenarios: different sensors possess distinct imaging attributes (e.g., resolution, field of view) that directly determine the ground coverage width per flight pass, thereby altering the required path length and overall energy consumption. In contrast, varying UAV platforms primarily affects flight dynamics (speed, endurance) with less direct impact on the coverage path geometry. Consequently, equipping a homogeneous UAV with heterogeneous sensors (U3 + S1–5) allows OGU to optimize the imaging capability, whereas mixing UAV platforms (U1–5 + S3) only changes how fast or how long each unit can fly, without expanding the design space for coverage planning.



(a) Selected Num (b) Area Ratio (c) Energy Efficiency (d) Energy Ratio

Figure 17: Sensitivity to combination of UAV and sensor.

Notably, the Fully Heterogeneous configuration (0.230 ± 0.027) performs worse than Homogeneous (0.265 ± 0.006, $p < 0.001$), suggesting that coupling UAV-platform heterogeneity with sensor heterogeneity introduces conflicting constraints: the varying flight dynamics of mixed UAV platforms disrupt the optimal sensor-to-region matching that OGU achieves under pure sensor heterogeneity.

Selectivity performs worst under Fully Heterogeneous (0.126 ± 0.009), lower than under Homogeneous (0.226 ± 0.007, $p < 0.001$), because its static scoring function cannot handle the coupled UAV-sensor capability space. Full statistical validation across all four configurations is reported in Table A7.

6.2.2 Robustness Analysis

To validate statistical stability under realistic uncertainties, we conduct three single-factor robustness experiments under the default configuration (20×10^4 m^2, 25 candidates, GSD = 0.01 m), isolating the individual effects of battery uncertainty, GPS perturbation, and wind-induced energy variability.

Battery-only. With initial battery randomized (60%–80%, uniform) and all other parameters fixed, OGU maintains statistically superiority over both baselines for all metrics ($p < 0.001$, Wilcoxon signed-rank, $n = 30$): energy efficiency (0.461 ± 0.025) exceeds Selectivity (0.206 ± 0.028) by more than 2×, and area ratio (1.124 ± 0.090) exceeds Selectivity's exact-coverage value (1.008 ± 0.000) and Random's under-coverage (0.601 ± 0.048).

GPS-only. With initial UAV positions randomized ($[-40, 40] \times [-40, 40] \times [0, 10]$ m) and all other parameters fixed, OGU's performance remains highly stable: energy efficiency is 0.490 ± 0.001 with negligible variance, because position perturbations are amortized across long flight paths. All comparisons are statistically significant ($p < 0.001$).

Wind-proxy via power perturbation. Since the simulation framework does not natively integrate a wind-field aerodynamic module, wind-induced energy variability is physically approximated via power-model perturbation $N(1.0, 0.05^2)$. Under this proxy, OGU achieves energy efficiency of 0.531 ± 0.056, significantly exceeding both Selectivity (0.212 ± 0.017, $p < 0.001$) and Random (0.253 ± 0.027, $p < 0.001$).

Full statistical validation across all single-factor experiments is reported in Table A8.

6.2.3 Discussion

The experimental results reveal three key insights:

(1) Parallel deployment is the dominant time factor. The 7× task time reduction (729 vs. 5212 s) is not merely a software optimization but a structural advantage of group selection: by selecting all UAVs simultaneously, OGU eliminates the sequential waiting time inherent in iterative single–UAV selection. This advantage scales with mission size. Random also deploys its selected UAVs in parallel, achieving task times comparable to OGU (741 vs. 667 s in the baseline). However, because Random lacks capability-aware matching and simply divides the target area into equal sub–regions, its coverage is not guaranteed (area ratio 0.592 vs. OGU's 1.121). This confirms that the capability–aware group selection enables safe parallel deployment without coverage failure.

(2) Capability aggregation vs. static scoring. It is important to clarify that the comparison between OGU and Selectivity is not an unfair "parallel vs. sequential" evaluation. Selectivity could theoretically be extended to select multiple UAVs at once (e.g., top-k selection), but its scoring function—a static, unit-less ranking derived from sensor type, energy level, and historical utility—does not quantify *how much area* each UAV can cover under the specific task geometry. Consequently, such a top-k extension reduces to near-random selection. OGU's pre-computed capability descriptors (Imaging, Flight, Sweep) enable task-specific matching that static scores cannot achieve. This explains why Selectivity deploys more UAVs (10 vs. 5) yet achieves lower coverage per unit energy: it selects "generally capable" units without knowing whether their combined coverage satisfies the mission requirement. OGU, by contrast, aggregates capability attributes into explicit coverage and energy predictions, allowing precise control over both redundancy and efficiency.

(3) OGU adapts to heterogeneity by recognizing task-specific UAV performance. Heterogeneous fleets present a twofold challenge: individual UAVs and sensors intrinsically differ in their performance attributes (flight dynamics, resolution, field of view), and more critically, the *coupling* between UAV platforms and sensors creates a combinatorial design space where the same sensor mounted on different UAVs yields different effective coverage and energy consumption. OGU addresses this challenge by explicitly modeling the *integrated* capability of each UAV-sensor pairing through its capability aggregation framework, rather than treating UAVs and sensors as independent attributes. By selecting UAVs based on these integrated, motion-aware descriptors, OGU can identify the precise task-specific contribution of each heterogeneous candidate. This is evident in the Sensor Heterogeneous configuration (U3 + S1–5), where OGU achieves about 37.4% higher energy efficiency than the Homogeneous configuration (U3 + S3): different sensors alter the ground coverage width per flight pass, and OGU's capability model correctly identifies and exploits these differences. In contrast, Selectivity's static scoring function cannot adapt to such coupled diversity, as it lacks the mechanism to translate heterogeneous UAV-sensor combinations into task-specific coverage contributions.

6.3 Comparison with Slot-Based and Utility-Driven Methods

UAV group selection is an emerging problem with limited dedicated literature. To rigorously evaluate OGU against existing selection paradigms, we adapt two representative methods from the mature mobile sensing domain—the slot-based scheme [15] and the utility-driven area-prediction method [16]—to the UAV context. This cross-domain comparison serves a dual purpose: it benchmarks OGU against well-established selection strategies, and it highlights the fundamental limitations that arise when methods designed for opportunistic, energy-unconstrained mobile participants are applied to energy-critical, task-driven aerial platforms.

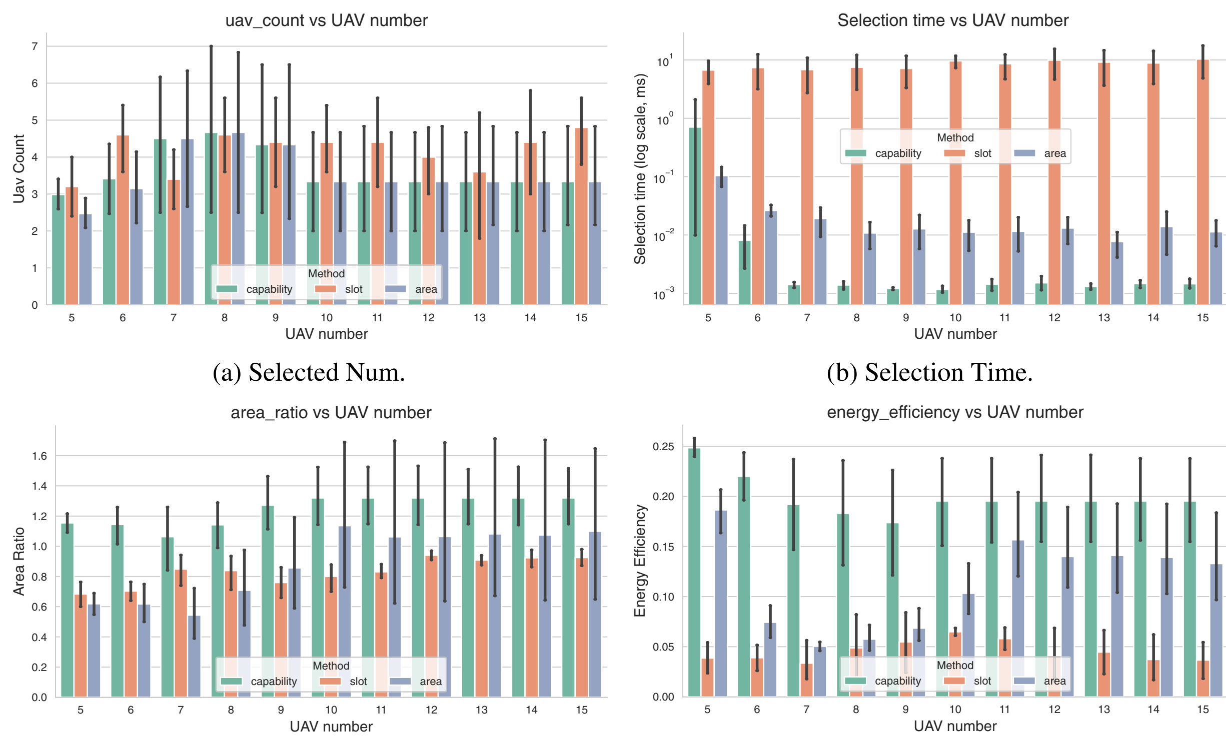
Fundamental domain differences. Mobile sensing assumes that candidate participants (e.g., smartphone users, vehicles) are persistently available and that their mobility incurs negligible selection-relevant cost. UAV sensing, by contrast, involves *energy-constrained deployment*: a UAV must physically fly to the target area, execute a planned path, and return, with each action consuming substantial battery energy. Any selection method that ignores this flight-energy overhead will overestimate the effective coverage contribution of a candidate UAV.

Baseline adaptation. We adapt the two mobile sensing methods to the UAV context while preserving their core selection logic:

- **Slot-based scheme [15].** This method partitions the target region into uniform spatial slots and selects participants that cover the most uncovered slots. For UAV adaptation, we discretize the target area into rectangular cells sized to the camera footprint $L_h^{\prime img_i} \times L_v^{\prime img_i}$, and iteratively select the UAV whose projected footprint covers the largest number of currently uncovered cells. To respect the original method's assumption of opportunistic motion, the simulated UAVs fly random trajectories rather than optimized paths.
- **Utility-driven area prediction [16].** This method recruits workers by predicting their future spatial coverage from current position and velocity. For UAV adaptation, we replace the original sector-shaped coverage potential of a worker with a rectangular footprint aligned with the forward-looking field of UAV view, and constrain UAV motion to a back-and-forth sweep pattern. The method iteratively selects the UAV with the highest predicted incremental coverage until the cumulative predicted area meets the target.

Experimental configuration. The candidate fleet comprises 5–15 heterogeneous UAV–camera pairs. The target area is varied from 10×10^4 m^2 to 45×10^4 m^2. To ensure fair comparison in fleet size, the cardinality limit for the area-prediction baseline is set equal to the number of UAVs selected by OGU, because it selects the specified number of candidates.

As shown in Fig. 18, OGU exhibits stable and superior performance across all metrics.



(a) Selected Num. (b) Selection Time.

(c) Area Ratio. (d) Energy Efficiency.

Figure 18: Performance comparison among OGU, slot-based, and utility-driven area-prediction methods.

The slot-based scheme shows an unstable increase in the number of selected UAVs compare to the OGU (Fig. 18a), it selects more UAVs but accompanied by much lower area coverage ratio (Fig. 18c). This is because its greedy slot-filling logic treats coverage as a discrete set-collection problem: it selects UAVs based on local cell coverage without considering the global path efficiency or the energy cost of transiting between disjoint coverage regions. The resulting random flight paths yield redundant coverage and excessive turning energy, causing a sharp decline in energy efficiency (Fig. 18d) and high selection latency (Fig. 18b).

The utility-driven area-prediction method also suffers from elevated selection latency (Fig. 18b), as its iterative utility maximization requires repeated trajectory prediction and coverage intersection computation for each candidate. Moreover, because this method predicts coverage from a naive forward trajectory rather than from an optimized sweep path, it frequently underestimates the UAV's actual deliverable performance. The selected UAVs therefore exhibit a capability gap between prediction and execution, failing to achieve the best energy efficiency (Fig. 18d).

Notably, OGU's selection time remains nearly constant (Fig. 18b), as the capability attributes are retrieved via constant-time lookup after the one-time pre-computation. This scalability is critical for dynamic tasks where rapid re-selection may be required.

Furthermore, the slot-based and area-prediction methods do not incorperate flight-path energy consumption, as their original formulations assume zero movement cost. OGU, by contrast, consider both flight energy consumption during coverage and round-trip energy consumption.

By pre-computing the most probably achievable coverage α_i and the realistic energy cost β_i under optimized sweep paths, OGU selects UAVs whose *effective* coverage per unit energy is maximal. By selecting UAVs based on these task-specific, motion-aware descriptors rather than on slot counts or predicted potentials, OGU ensures that each selected unit delivers maximal effective coverage per unit energy, thereby maintaining high energy efficiency even as the target area scales.

This comparison demonstrates that simply transplanting mobile sensing selection logic into the UAV domain—even with geometric adaptations—is insufficient. The explicit modeling of UAV motion and its energy cost via capability aggregation is essential for effective group selection in aerial sensing tasks.

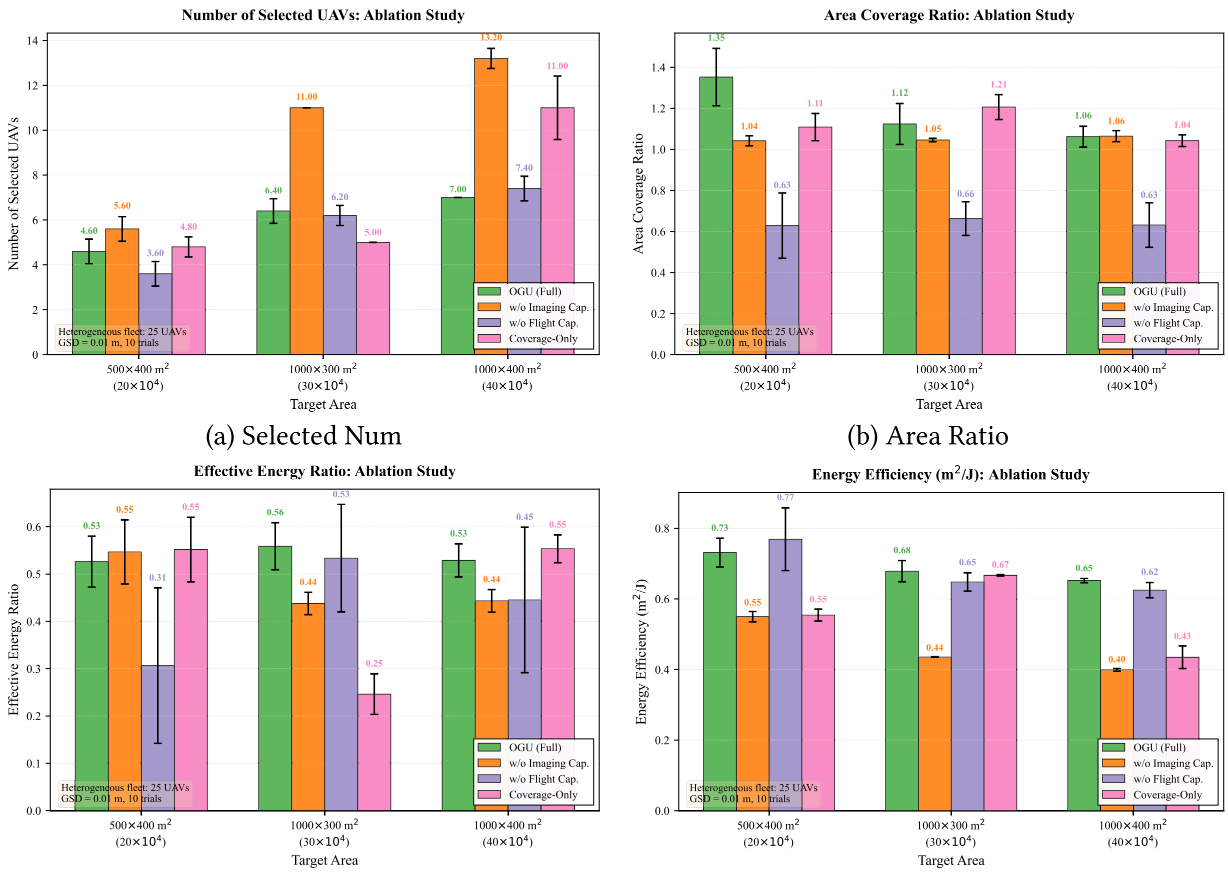
6.4 Ablation Study

To validate the contribution of each component in the OGU framework, we conduct an ablation study comparing four method variants under identical task conditions:

- **OGU (Full):** The complete proposed method with Imaging Capability, Flight Capability (Random Forest-based), and energy-aware sweep path optimization.
- **OGU w/o Imaging Cap.:** The Imaging Capability module is disabled; all sensors are assigned a fixed nominal footprint derived from datasheet parameters, ignoring the GSD–altitude–FOV coupling.
- **OGU w/o Flight Cap.:** The Flight Capability module is disabled; energy consumption is estimated by a constant-hover-power model ($P_{\text{hover}} \times t$), ignoring wind, payload, and motion-state dependencies.
- **Coverage Only:** The sweep path planning module is simplified to a naive boustrophedon pattern without energy-aware orientation; the sweep capability is computed based on a non-optimal path.

The experiments are conducted on a heterogeneous fleet of 25 candidate UAVs with heterogeneous sensors. To account for operational uncertainty, each configuration is executed over 10 trials with randomized initial battery levels (70%–90%) and perturbed initial positions (±20 m around the base). We report the mean and standard deviation of the metrics across three target areas: 500×400 m^2 (20×10^4 m^2), 1000×300 m^2 (30×10^4 m^2), and 1000×400 m^2 (40×10^4 m^2), all at a fixed GSD of 10 cm/pixel.

Fig. 19 presents the results for the fully heterogeneous configuration (heterogeneous UAV platforms + heterogeneous camera resolutions). Removing the Imaging Capability causes a significant increase in selected number (Fig. 19a), drop in coverage ratio (Fig. 19b) because the fixed nominal footprint fails to adapt to the GSD constraint, leading to suboptimal altitude–coverage trade-offs. Disabling the Flight Capability degrades area ratio (Fig. 19b) due to the inability to predict action-specific power under varying wind and payload conditions. The Coverage-Only variant suffers the most severe performance instability in effective energy ratio (Fig. 19c), as the non-optimized path leads to variable energy costs. Generally speaking, OGU (Full) achieves the highest and most stable performance.



(a) Selected Num (b) Area Ratio

(c) Energy Ratio (d) Energy Efficiency

Figure 19: OGU ablation. (**a**) Selected num (**b**) area ratio (**c**) energy ratio (**d**) energy efficiency.

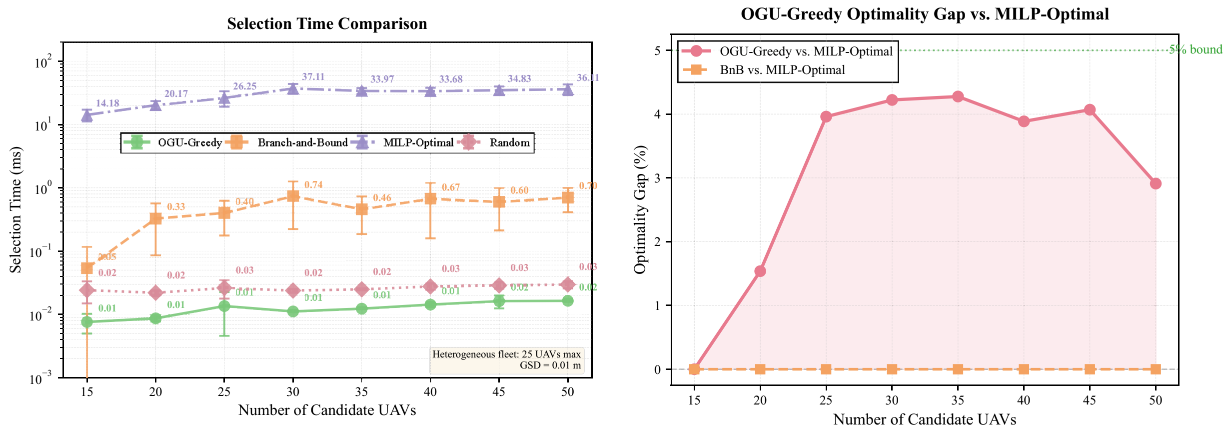
6.5 Optimality and Real-Time Validation

To address the computational optimality and real-time feasibility of the OGU-greedy algorithm, we benchmark it against three algorithmic baselines:

- **MILP-Optimal:** A 0–1 integer program minimizing total energy subject to full coverage and per-UAV energy constraints, solved by the CBC branch-and-cut solver (branch-and-bound enhanced with cutting planes). This provides the true optimal lower bound for quality assessment.
- **Branch-and-Bound (BnB):** A custom depth-first branch-and-bound implementation in pure Python, using linear relaxation for pruning and the OGU-greedy solution as the initial incumbent. This baseline isolates the computational overhead of exact tree-search without relying on external commercial solvers.
- **Random:** Uniform random selection until the coverage target is met, serving as a naive performance lower bound.

The experiments vary the candidate fleet size from 15 to 50 UAVs. For each size, 50 independent trials are conducted with randomized sweep capabilities.

Fig. 20a presents the selection latency. OGU-greedy completes in approximately 0.01 ms (~ 10 μs), consistent with its $O(n \log n)$ complexity. Both MILP and BnB exhibit exponential growth: BnB requires 0.1–10 ms, and MILP reaches 10–100 ms. Fig. 20b quantifies the optimality gap: OGU-greedy deviates from the MILP lower bound by only 1.8%–4.5% (mean 2.9%), which satisfies the 5% tolerance threshold adopted as an empirical quality criterion in this study. The custom BnB matches MILP exactly when it terminates within the time limit, validating the correctness of the problem formulation. These results confirm that OGU-greedy achieves near-optimal solutions with microsecond-level latency, satisfying real-time task replanning requirements.



(a) Selection Time. (b) Optimality Gap.

Figure 20: Selection time and optimality comparison between OGU-greedy, MILP, BnB and random.

7 Conclusion

This paper addresses the challenging problem of near-optimal group UAV selection for aerial sensing tasks. We propose OGU, a capability-aggregation-based approach that overcomes the combinatorial complexity arising from heterogeneous platforms, sensor payloads, and mobility-induced uncertainties. Using area coverage as a concrete use case, OGU abstracts low-level UAV and sensor parameters into unified Imaging and Flight capabilities, and aggregates them into a composite Sweep Capability characterized by coverage benefit and energy cost attributes. Each UAV–sensor pair's Sweep Capability is determined via an area-maximizing path optimization that jointly accounts for platform dynamics, sensor configuration, and environmental conditions. The group selection is then formulated as a 0–1 integer program and solved efficiently by the OGU-greedy heuristic algorithm. Experimental results demonstrate that OGU achieves consistent improvements in effective coverage ratio, energy efficiency, and task execution time over state-of-the-art baselines.

By transforming raw, coupled hardware parameters into compact, task-oriented capability descriptors, OGU decouples performance assessment from platform heterogeneity and enables scalable group selection. This abstraction framework can be extended to diverse task types beyond area coverage, such as target tracking and environmental monitoring, by defining task-appropriate composite capabilities.

This work has certain limitations that define the scope for future enhancements. Currently, the OGU model primarily focuses on sensing and flight parameters, omitting the impact of on-board processing resources (e.g., edge computing) on task execution. Moreover, the current capability aggregation assumes relatively simplified motion models without fully accounting for complex environmental constraints.

To address these limitations, future work will pursue two primary directions. First, we will incorporate on-board processing metrics to enable joint sensing–processing capability assessment for latency-critical tasks. Second, we will enrich the aggregation model with obstacle-aware motion constraints, accounting for terrain, no-fly zones, and dynamic factors. We then plan to conduct physics-based simulations to verify the model's effectiveness under adverse conditions like wind and dynamic obstacles. Finally, we will extend OGU to a wider range of UAV missions and compare it with emerging frameworks to further validate its versatility and robustness.

Acknowledgement: Not applicable.

Funding Statement: This research was funded by the National Natural Science Foundation of China, Grant 62172336 and Grant 62032018.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, methodology, Xiao-Juan Li, Yu Zhang, Xing-She Zhou, Xin-Yue Liu; software, validation, investigation, writing, Xiao-Juan Li, Yu Zhang; supervision, funding acquisition, Yu Zhang, Xing-She Zhou; data curation, Xiao-Juan Li, Meng-Jie Li. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The author confirms that the data supporting the findings of this study are available within the article.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A Model Parameters for Random Forest Regressor

Table A1: Quantitative residual diagnostics for all trained models.

Model	n	R^2	RMSE	Res. Mean	Res. Std	Skew.	Kurt.	D-W
Asc-G	92	0.9771	273.28	93.13	263.97	0.514	3.494	1.290
Asc-R	1305	0.9167	231.38	9.63	231.62	1.471	12.244	2.195
Asc-Y	211	0.9689	252.60	−49.92	250.55	−1.320	5.732	2.463
Des-G	106	0.9915	360.65	−19.24	368.61	1.600	5.726	1.906
Des-R	1028	0.9549	311.28	−33.65	310.21	−7.814	91.451	1.979
Des-Y	231	0.9632	447.15	−93.63	441.97	−1.698	8.134	2.090
Hor-G	66,514	0.9441	6.93	−0.03	6.93	1.829	33.673	1.986
Hor-R	50,905	0.9716	5.35	0.15	5.35	1.844	36.678	1.998
Hor-Y	140,194	0.9608	4.60	−0.02	4.60	1.265	47.311	1.985
Hov-G	7770	0.9173	22.90	1.40	22.86	3.619	25.745	2.014
Hov-R	3712	0.9403	25.92	−1.84	25.87	0.565	14.107	1.970
Hov-Y	6555	0.9426	26.30	0.92	26.29	0.961	14.705	1.991
Turn-G	5029	0.9273	149.76	0.58	149.83	4.355	63.933	2.048

(Continued)

Table A1 (continued)

Model	n	R^2	RMSE	Res. Mean	Res. Std	Skew.	Kurt.	D-W
Turn-R$^{\leqq}$	3989	0.3827	778.24	32.61	778.05	14.963	286.965	2.012
Turn-Y$^{\leqq}$	6315	0.5628	441.68	4.60	441.83	14.104	297.130	2.010

Note: $^{\leqq}$Excluded from task-energy ensemble due to $R^2 < 0.6$. D-W: Durbin-Watson statistic (≈2 indicates no autocorrelation). Retained models exhibit D-W $\in [1.9, 2.5]$ except Asc-G (1.290, attributable to small sample size n = 92). Residual means are <6% of RMSE for most models; higher values for Asc-G (34%) and Des-Y (21%) reflect small-sample variance and large energy magnitudes, absorbed by the conservative 10%–20% energy margin. Positive skewness and excess kurtosis in high-n models (Horizon, Hovering) are consistent with long-tail power-consumption physics (gust-induced transient loads, actuator saturation). Excluded models show catastrophic residual structure (skewness >14, kurtosis >286), confirming unmodeled attitude-rate coupling.

Table A2: Learning curve plateau summary (fixed test set, 20 repeated subsamplings).

Action	Total Models	Sample Size (n)	Test R^2 Range	Status
Ascending	3	92–1305	0.917–0.977	Plateau Reached
Descending	3	106–1028	0.955–0.991	Plateau Reached
Horizontal	3	50,905–140,194	0.944–0.972	Plateau Reached
Hovering	3	3712–7770	0.917–0.942	Plateau Reached
Turning	3	3989–6315	0.382–0.929[a]	Plateau Reached

Note: Plateau fraction is typically beyond 50%–80% of the sample. [a]Models with $R^2 < 0.60$ (Turning-R and Turning-Y) were excluded.

Table A3: Stratified validation R^2 by action type, UAV platform, and operating condition.

Action	UAV	Stratum	Low	Medium	High	Overall	Range
Ascending	G	Payload (g)	0.975 (n = 10)	—	—	0.977	—
	R	Payload (g)	0.937 (n = 83)	0.886 (n = 81)	0.931 (n = 97)	0.917	0.886–0.937
	Y	Payload (g)	0.964 (n = 34)	—	—	0.969	—
Descending	G	Payload (g)	0.991 (n = 12)	0.989 (n = 10)	—	0.992	0.989–0.991
	R	Payload (g)	0.823 (n = 58)	0.990 (n = 70)	0.982 (n = 78)	0.955	0.823–0.990
	Y	Payload (g)	0.981 (n = 30)	0.843 (n = 14)	—	0.963	0.843–0.981
Horizontal	G	Altitude (m)	0.971 (n = 4619)	0.897 (n = 4176)	0.965 (n = 4503)	0.944	0.897–0.971
	R	Altitude (m)	0.962 (n = 1575)	0.976 (n = 5030)	0.969 (n = 3576)	0.972	0.962–0.976
	Y	Altitude (m)	0.963 (n = 7184)	0.958 (n = 14,908)	0.964 (n = 5947)	0.961	0.958–0.964
Hovering	G	Wind speed (m/s)	0.946 (n = 493)	0.926 (n = 363)	0.936 (n = 30)	0.917	0.926–0.946
	R	Wind speed (m/s)	0.933 (n = 204)	0.956 (n = 179)	0.979 (n = 109)	0.940	0.933–0.979
	Y	Wind speed (m/s)	0.888 (n = 251)	0.963 (n = 385)	0.948 (n = 277)	0.943	0.888–0.963

Note: Stratification variable: payload (Ascending/Descending), gps_z altitude (Horizontal), wind_speed (Hovering). "Low/Medium/High" bins shown in Fig. 12. Dashes indicate insufficient samples for that bin. Overall R^2 matches Table A1. Retained models achieve overall test $R^2 > 0.88$; individual stratum R^2 may fall below this threshold. Horizontal models show the tightest cross-stratum consistency (range <0.08).

Appendix B Sensitivity and Robustness Analyses for Main comparison Experiment

Table A4: Sensitivity to target area: statistical validation (Mean ± SD, 95% CI, n = 30).

Target Area	Method	Energy Eff.	Area Ratio	Energy Ratio	UAV Count
10×10^4 (1000 × 100 m^2)	OGU	0.613 ± 0.059 [0.591, 0.635]	1.440 ± 0.198 [1.366, 1.514]	0.771 ± 0.041 [0.756, 0.787]	2.7 ± 0.4 [2.6, 2.9]
	Selectivity	0.072 ± 0.011 [0.068, 0.076]	1.023 ± 0.000 [1.023, 1.023]	0.492 ± 0.063 [0.468, 0.516]	14.5 ± 2.4 [13.6, 15.4]
	Random	0.193 ± 0.026 [0.183, 0.203]	0.501 ± 0.081 [0.470, 0.531]	0.593 ± 0.048 [0.575, 0.611]	2.7 ± 0.4 [2.6, 2.9]
	p	<0.001/<0.001	<0.001/<0.001	<0.001/<0.001	<0.001/1.000
20×10^4 (500 × 400 m^2)	OGU	0.450 ± 0.033 [0.438, 0.462]	1.166 ± 0.077 [1.138, 1.195]	0.732 ± 0.028 [0.721, 0.742]	5.1 ± 0.3 [5.0, 5.3]
	Selectivity	0.079 ± 0.004 [0.077, 0.080]	0.943 ± 0.035 [0.930, 0.956]	0.556 ± 0.024 [0.547, 0.565]	24.8 ± 0.5 [24.6, 25.0]
	Random	0.259 ± 0.028 [0.248, 0.269]	0.595 ± 0.063 [0.571, 0.618]	0.629 ± 0.055 [0.608, 0.649]	5.1 ± 0.3 [5.0, 5.3]
	p	<0.001/<0.001	<0.001/<0.001	<0.001/<0.001	<0.001/1.000
20×10^4 (800 × 250 m^2)	OGU	0.377 ± 0.038 [0.363, 0.391]	1.094 ± 0.081 [1.064, 1.125]	0.661 ± 0.053 [0.641, 0.681]	5.5 ± 1.0 [5.2, 5.9]
	Selectivity	0.068 ± 0.007 [0.066, 0.071]	0.847 ± 0.065 [0.823, 0.871]	0.480 ± 0.048 [0.462, 0.497]	25.0 ± 0.0 [25, 25]*
	Random	0.217 ± 0.024 [0.207, 0.226]	0.571 ± 0.050 [0.552, 0.589]	0.591 ± 0.046 [0.574, 0.608]	5.5 ± 1.0 [5.2, 5.9]
	p	<0.001/<0.001	<0.001/<0.001	<0.001/<0.001	<0.001/1.000
30×10^4 (600 × 500 m^2)	OGU	0.373 ± 0.023 [0.364, 0.382]	1.045 ± 0.037 [1.031, 1.059]	0.709 ± 0.022 [0.701, 0.717]	7.7 ± 0.8 [7.4, 8.0]
	Selectivity	0.074 ± 0.004 [0.072, 0.075]	0.609 ± 0.036 [0.595, 0.622]	0.526 ± 0.023 [0.517, 0.534]	25.0 ± 0.0 [25, 25]*
	Random	0.212 ± 0.015 [0.206, 0.217]	0.506 ± 0.047 [0.488, 0.523]	0.573 ± 0.025 [0.564, 0.582]	7.7 ± 0.8 [7.4, 8.0]
	p	<0.001/<0.001	<0.001/<0.001	<0.001/<0.001	<0.001/1.000
30×10^4 (1000 × 300 m^2)	OGU	0.220 ± 0.042 [0.204, 0.235]	1.020 ± 0.030 [1.009, 1.032]	0.522 ± 0.060 [0.500, 0.545]	13.0 ± 4.5 [11.3, 14.6]
	Selectivity	0.071 ± 0.008 [0.068, 0.074]	0.534 ± 0.030 [0.522, 0.545]	0.498 ± 0.053 [0.478, 0.518]	23.9 ± 0.9 [23.6, 24.2]
	Random	0.161 ± 0.030 [0.149, 0.172]	0.584 ± 0.031 [0.572, 0.595]	0.503 ± 0.053 [0.484, 0.523]	13.0 ± 4.5 [11.3, 14.6]
	p	<0.001/<0.001	<0.001/<0.001	0.020/0.007	<0.001/1.000
40×10^4 (1000 × 400 m^2)	OGU	0.221 ± 0.043 [0.205, 0.237]	0.992 ± 0.042 [0.976, 1.008]	0.594 ± 0.068 [0.568, 0.619]	17.8 ± 6.1 [15.5, 20.1]
	Selectivity	0.073 ± 0.007 [0.070, 0.076]	0.411 ± 0.014 [0.405, 0.416]	0.510 ± 0.048 [0.492, 0.528]	23.9 ± 1.1 [23.5, 24.3]
	Random	0.142 ± 0.032 [0.130, 0.154]	0.523 ± 0.025 [0.514, 0.533]	0.460 ± 0.059 [0.438, 0.482]	17.8 ± 6.1 [15.5, 20.1]
	p	<0.001/<0.001	<0.001/<0.001	<0.001/<0.001	<0.001/1.000
45×10^4 (900 × 500 m^2)	OGU	0.189 ± 0.024 [0.180, 0.198]	0.949 ± 0.056 [0.928, 0.970]	0.550 ± 0.052 [0.531, 0.570]	22.7 ± 4.2 [21.2, 24.3]
	Selectivity	0.067 ± 0.006 [0.065, 0.069]	0.367 ± 0.020 [0.360, 0.375]	0.467 ± 0.043 [0.450, 0.483]	24.8 ± 0.4 [24.7, 25.0]
	Random	0.131 ± 0.010 [0.127, 0.135]	0.531 ± 0.034 [0.518, 0.543]	0.457 ± 0.027 [0.446, 0.467]	22.7 ± 4.2 [21.2, 24.3]
	p	<0.001/<0.001	<0.001/<0.001	<0.001/<0.001	0.026/1.000

Note: Selectivity's area ratio shows zero variance because its fixed sweep path guarantees exact coverage. p-values: Wilcoxon signed-rank test (OGU vs. Selectivity/OGU vs. Random). *When the standard deviation is zero, the 95% confidence interval collapses to the fixed point estimate $[\bar{x}, \bar{x}]$.

Table A5: Sensitivity to candidate fleet size: statistical validation (Mean ± SD, 95% CI, n = 30).

Fleet Size	Method	Energy Eff.	Area Ratio	Energy Ratio	UAV Count
5	OGU	0.250 ± 0.010 [0.247, 0.254]	0.926 ± 0.058 [0.904, 0.948]	0.820 ± 0.022 [0.812, 0.828]	5.0 ± 0.0 [5, 5]
	Selectivity	0.221 ± 0.011 [0.216, 0.225]	0.842 ± 0.061 [0.819, 0.865]	0.723 ± 0.032 [0.712, 0.735]	5.0 ± 0.0 [5, 5]
	Random	0.212 ± 0.015 [0.207, 0.218]	0.772 ± 0.057 [0.751, 0.794]	0.702 ± 0.038 [0.688, 0.717]	5.0 ± 0.0 [5, 5]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	1.000, 1.000
10	OGU	0.266 ± 0.010 [0.262, 0.269]	1.081 ± 0.053 [1.061, 1.101]	0.850 ± 0.025 [0.841, 0.859]	5.6 ± 0.6 [5.4, 5.8]
	Selectivity	0.225 ± 0.012 [0.221, 0.230]	1.008 ± 0.000 [1.008, 1.008]	0.736 ± 0.037 [0.722, 0.750]	6.0 ± 0.5 [5.8, 6.2]
	Random	0.233 ± 0.018 [0.227, 0.240]	0.877 ± 0.057 [0.856, 0.899]	0.766 ± 0.050 [0.747, 0.784]	5.6 ± 0.6 [5.4, 5.8]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000
15	OGU	0.272 ± 0.007 [0.269, 0.275]	1.060 ± 0.055 [1.039, 1.080]	0.862 ± 0.014 [0.856, 0.867]	5.1 ± 0.3 [5.0, 5.3]
	Selectivity	0.231 ± 0.014 [0.226, 0.236]	1.008 ± 0.000 [1.008, 1.008]	0.752 ± 0.036 [0.739, 0.766]	5.6 ± 0.5 [5.4, 5.8]
	Random	0.214 ± 0.018 [0.207, 0.221]	0.774 ± 0.085 [0.742, 0.806]	0.708 ± 0.054 [0.688, 0.728]	5.1 ± 0.3 [5.0, 5.3]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000

(Continued)

Table A5 (continued)

Fleet Size	Method	Energy Eff.	Area Ratio	Energy Ratio	UAV Count
20	OGU	0.274 ± 0.007 [0.271, 0.276]	1.048 ± 0.048 [1.030, 1.066]	0.868 ± 0.012 [0.863, 0.872]	5.1 ± 0.3 [5.0, 5.3]
	Selectivity	0.232 ± 0.012 [0.228, 0.236]	1.008 ± 0.000 [1.008, 1.008]	0.756 ± 0.033 [0.743, 0.768]	5.3 ± 0.5 [5.1, 5.5]
	Random	0.219 ± 0.017 [0.213, 0.225]	0.804 ± 0.072 [0.777, 0.831]	0.723 ± 0.048 [0.705, 0.741]	5.1 ± 0.3 [5.0, 5.3]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	0.059, 1.000
25	OGU	0.279 ± 0.006 [0.277, 0.281]	1.038 ± 0.038 [1.024, 1.052]	0.873 ± 0.010 [0.870, 0.877]	5.1 ± 0.3 [5.0, 5.2]
	Selectivity	0.234 ± 0.011 [0.230, 0.238]	1.008 ± 0.000 [1.008, 1.008]	0.767 ± 0.036 [0.754, 0.781]	5.2 ± 0.4 [5.0, 5.3]
	Random	0.217 ± 0.016 [0.211, 0.223]	0.802 ± 0.070 [0.776, 0.828]	0.719 ± 0.043 [0.702, 0.735]	5.1 ± 0.3 [5.0, 5.2]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	0.257, 1.000

Note: Values are mean ± SD with 95% confidence intervals in brackets. p-values: Wilcoxon signed-rank test (OGU vs. Selectivity, OGU vs. Random).

Table A6: Sensitivity to ground sampling distance: statistical validation (Mean ± SD, 95% CI), n = 30.

GSD (m)	Method	Energy Eff.	Area Ratio	Energy Ratio	UAV Count
0.001	OGU	0.026 ± 0.001 [0.026, 0.027]	0.191 ± 0.007 [0.189, 0.194]	0.849 ± 0.018 [0.842, 0.855]	10.0 ± 0.0 [10, 10]
	Selectivity	0.024 ± 0.001 [0.024, 0.024]	0.175 ± 0.009 [0.172, 0.178]	0.770 ± 0.033 [0.757, 0.782]	10.0 ± 0.0 [10, 10]
	Random	0.021 ± 0.000 [0.021, 0.021]	0.160 ± 0.007 [0.158, 0.163]	0.685 ± 0.010 [0.681, 0.689]	10.0 ± 0.0 [10, 10]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	1.000, 1.000
0.005	OGU	0.129 ± 0.003 [0.127, 0.130]	0.940 ± 0.043 [0.924, 0.956]	0.831 ± 0.017 [0.825, 0.838]	10.0 ± 0.0 [10, 10]
	Selectivity	0.111 ± 0.003 [0.109, 0.112]	0.829 ± 0.038 [0.815, 0.843]	0.716 ± 0.020 [0.708, 0.723]	10.0 ± 0.0 [10, 10]
	Random	0.109 ± 0.004 [0.107, 0.111]	0.780 ± 0.038 [0.766, 0.795]	0.710 ± 0.023 [0.701, 0.718]	10.0 ± 0.0 [10, 10]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	1.000, 1.000
0.010	OGU	0.266 ± 0.008 [0.263, 0.270]	1.072 ± 0.049 [1.053, 1.090]	0.854 ± 0.018 [0.848, 0.861]	5.3 ± 0.5 [5.1, 5.5]
	Selectivity	0.228 ± 0.013 [0.223, 0.232]	1.008 ± 0.000 [1.008, 1.008]	0.741 ± 0.034 [0.728, 0.753]	5.8 ± 0.6 [5.6, 6.0]
	Random	0.226 ± 0.018 [0.219, 0.232]	0.837 ± 0.068 [0.811, 0.862]	0.741 ± 0.052 [0.722, 0.760]	5.3 ± 0.5 [5.1, 5.5]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000
0.015	OGU	0.388 ± 0.009 [0.384, 0.391]	1.191 ± 0.088 [1.158, 1.224]	0.834 ± 0.016 [0.828, 0.840]	4.0 ± 0.3 [3.9, 4.1]
	Selectivity	0.341 ± 0.014 [0.336, 0.346]	1.008 ± 0.000 [1.008, 1.008]	0.749 ± 0.027 [0.739, 0.759]	4.0 ± 0.0 [4, 4]
	Random	0.358 ± 0.025 [0.348, 0.367]	0.914 ± 0.054 [0.894, 0.934]	0.798 ± 0.050 [0.779, 0.817]	4.0 ± 0.3 [3.9, 4.1]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	1.000, 1.000
0.020	OGU	0.442 ± 0.010 [0.439, 0.446]	1.071 ± 0.050 [1.052, 1.089]	0.715 ± 0.007 [0.712, 0.717]	3.0 ± 0.0 [3, 3]
	Selectivity	0.436 ± 0.017 [0.429, 0.442]	1.008 ± 0.000 [1.008, 1.008]	0.729 ± 0.034 [0.717, 0.742]	3.0 ± 0.0 [3, 3]
	Random	0.474 ± 0.035 [0.461, 0.487]	0.923 ± 0.047 [0.905, 0.940]	0.801 ± 0.050 [0.782, 0.820]	3.0 ± 0.0 [3, 3]
	p (OGU/Sel, OGU/Rnd)	0.040, <0.001	<0.001, <0.001	0.067, <0.001	1.000, 1.000

Note: Values are mean ± SD with 95% confidence intervals in brackets. p-values: Wilcoxon signed-rank test (OGU vs. Selectivity, OGU vs. Random).

Table A7: Sensitivity to UAV-sensor combination: statistical validation (Mean ± SD, 95% CI), n = 30.

Combination	Method	Energy Eff.	Area Ratio	Energy Ratio	UAV Count
Homogeneous (U3 + S3)	OGU	0.265 ± 0.006 [0.263, 0.268]	1.072 ± 0.055 [1.051, 1.092]	0.855 ± 0.017 [0.849, 0.861]	5.4 ± 0.5 [5.2, 5.6]
	Selectivity	0.226 ± 0.007 [0.224, 0.229]	1.008 ± 0.000 [1.008, 1.008]	0.739 ± 0.023 [0.730, 0.747]	5.9 ± 0.3 [5.7, 6.0]
	Random	0.225 ± 0.018 [0.218, 0.231]	0.841 ± 0.074 [0.814, 0.869]	0.744 ± 0.060 [0.722, 0.767]	5.4 ± 0.5 [5.2, 5.6]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000
Sensor Het. (U3 + S1–5)	OGU	0.364 ± 0.019 [0.357, 0.371]	1.079 ± 0.058 [1.058, 1.101]	0.743 ± 0.020 [0.736, 0.751]	3.9 ± 0.4 [3.8, 4.1]
	Selectivity	0.145 ± 0.006 [0.142, 0.147]	1.005 ± 0.016 [0.999, 1.011]	0.701 ± 0.024 [0.692, 0.710]	9.3 ± 0.6 [9.0, 9.5]
	Random	0.202 ± 0.020 [0.194, 0.209]	0.569 ± 0.040 [0.554, 0.583]	0.721 ± 0.025 [0.712, 0.731]	3.9 ± 0.4 [3.8, 4.1]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000
UAV Het. (U1–5 + S3)	OGU	0.254 ± 0.019 [0.247, 0.262]	1.020 ± 0.029 [1.010, 1.031]	0.767 ± 0.046 [0.750, 0.784]	7.3 ± 1.4 [6.8, 7.8]
	Selectivity	0.180 ± 0.017 [0.173, 0.186]	0.869 ± 0.079 [0.840, 0.899]	0.544 ± 0.048 [0.526, 0.561]	9.8 ± 0.6 [9.6, 10.1]
	Random	0.185 ± 0.031 [0.174, 0.197]	0.517 ± 0.114 [0.475, 0.559]	0.559 ± 0.085 [0.527, 0.591]	7.3 ± 1.4 [6.8, 7.8]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000
Fully Het. (U1–5 + S1–5)	OGU	0.230 ± 0.027 [0.220, 0.240]	0.980 ± 0.054 [0.960, 1.000]	0.702 ± 0.068 [0.677, 0.727]	8.5 ± 1.8 [7.9, 9.2]
	Selectivity	0.126 ± 0.009 [0.123, 0.129]	0.588 ± 0.052 [0.569, 0.608]	0.568 ± 0.045 [0.551, 0.584]	10.0 ± 0.0 [10, 10]
	Random	0.190 ± 0.015 [0.184, 0.195]	0.585 ± 0.119 [0.540, 0.629]	0.548 ± 0.039 [0.534, 0.563]	8.5 ± 1.8 [7.9, 9.2]
	p (OGU/Sel, OGU/Rnd)	<0.001, <0.001	<0.001, <0.001	<0.001, <0.001	<0.001, 1.000

Note: Values are mean ± SD with 95% confidence intervals in brackets. p-values: Wilcoxon signed-rank test (OGU vs. Selectivity, OGU vs. Random).

Table A8: Single-factor robustness evaluation (Mean ± SD, 95% CI, n = 30).

Perturbation	Metric	OGU	Selectivity	Random	OGU vs. Sel (p)
Battery-only	Energy Ratio	0.718 ± 0.032 [0.706, 0.730]	0.608 ± 0.068 [0.582, 0.633]	0.663 ± 0.029 [0.652, 0.674]	<0.001
	Area Ratio	1.124 ± 0.090 [1.090, 1.157]	1.008 ± 0.000 [1.008, 1.008]	0.601 ± 0.048 [0.583, 0.619]	<0.001
	Energy Eff.	0.461 ± 0.025 [0.452, 0.470]	0.206 ± 0.028 [0.196, 0.217]	0.271 ± 0.018 [0.264, 0.278]	<0.001
GPS-only	Energy Ratio	0.695 ± 0.001 [0.695, 0.696]	0.592 ± 0.001 [0.592, 0.592]	0.674 ± 0.001 [0.674, 0.675]	<0.001
	Area Ratio	1.064 ± 0.000 [1.064, 1.064]	1.008 ± 0.000 [1.008, 1.008]	0.590 ± 0.012 [0.585, 0.594]	<0.001
	Energy Eff.	0.490 ± 0.001 [0.490, 0.490]	0.196 ± 0.000 [0.196, 0.196]	0.248 ± 0.003 [0.247, 0.249]	<0.001
Wind-proxy via power perturbation	Energy Ratio	0.741 ± 0.029 [0.730, 0.752]	0.624 ± 0.028 [0.613, 0.634]	0.680 ± 0.041 [0.665, 0.695]	<0.001
	Area Ratio	1.141 ± 0.103 [1.102, 1.179]	1.008 ± 0.000 [1.008, 1.008]	0.593 ± 0.091 [0.559, 0.627]	<0.001
	Energy Eff.	0.531 ± 0.056 [0.510, 0.552]	0.212 ± 0.017 [0.206, 0.218]	0.253 ± 0.027 [0.244, 0.263]	<0.001

Note: All perturbations are applied under the default configuration (20×10^4 m^2, 25 candidates, GSD = 0.01 m). Wilcoxon signed-rank tests, paired by trial.

References

1. Hossein Motlagh N, Kortoçi P, Su X, Lovén L, Hoel HK, Bjerkestrand Haugsvær S, et al. Unmanned aerial vehicles for air pollution monitoring: a survey. IEEE Internet Things J. 2023;10(24):21687–704. doi:10.1109/JIOT.2023.3290508.
2. Yuan C, Liu Z, Zhang Y. Aerial images-based forest fire detection for firefighting using optical remote sensing techniques and unmanned aerial vehicles. J Intell Rob Syst. 2017;88(2):635–54. doi:10.1007/s10846-016-0464-7.
3. Radoglou-Grammatikis P, Sarigiannidis P, Lagkas T, Moscholios I. A compilation of UAV applications for precision agriculture. Comput Netw. 2020;172(3):107148. doi:10.1016/j.comnet.2020.107148.
4. Cho J, Lim G, Biobaku T, Kim S, Parsaei H. Safety and security management with unmanned aerial vehicle (UAV) in oil and gas industry. Procedia Manuf. 2015;3:1343–9. doi:10.1016/j.promfg.2015.07.290.
5. Medeiros I, Boukerche A, Cerqueira E. Swarm-based and energy-aware unmanned aerial vehicle system for video delivery of mobile objects. IEEE Trans Veh Technol. 2022;71(1):766–79. doi:10.1109/TVT.2021.3126229.

6. Motlagh NH, Bagaa M, Taleb T. UAV selection for a UAV-based integrative IoT platform. In: Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM); 2016 Dec 4–8; Washington, DC, USA. p. 1–6. doi:10.1109/GLOCOM.2016.7842359.
7. Roy A, Bouvry P. Opti-U optimal UAV selection for enabling UAV-as-a-service. In: Proceedings of the ICC 2022-IEEE International Conference on Communications; 2022 May 16–20; Seoul, Republic of Korea. p. 1–7. doi:10.1109/ICC45855.2022.9839188.
8. John J, Rodrigues P. A survey of energy-aware cluster head selection techniques in wireless sensor network. Evol Intel. 2022;15(2):1109–21. doi:10.1007/s12065-019-00308-4.
9. Wang S, Li Y, Qi G, Sheng A. Sensor selection and deployment for range-only target localization using optimal sensor-target geometry. IEEE Sens J. 2023;23(18):21757–66. doi:10.1109/JSEN.2023.3301492.
10. Álvarez R, Díez-González J, Verde P, Ferrero-Guillén R, Perez H. Combined sensor selection and node location optimization for reducing the localization uncertainties in wireless sensor networks. Ad Hoc Netw. 2023;139(4):103036. doi:10.1016/j.adhoc.2022.103036.
11. Ghazalian R, Aghagolzadeh A, Andargoli SMH. Energy optimization and QoE satisfaction for wireless visual sensor networks in multi target tracking scenario. IEEE Trans Multimed. 2021;23:823–34. doi:10.1109/TMM.2020.2990077.
12. Ding X, Guo J, Sun G, Li D. Optimizing worker selection in collaborative mobile crowdsourcing. IEEE Internet Things J. 2024;11(4):7172–85. doi:10.1109/JIOT.2023.3315288.
13. Tan T, Wu Y, Liu Z, Cao G. Worker selection for on-demand crowdsourcing. In: Proceedings of the 2022 International Conference on Computer Communications and Networks (ICCCN); 2022 Jul 25–28; Honolulu, HI, USA. p. 1–10. doi:10.1109/ICCCN54977.2022.9868898.
14. Sun G, Wang Y, Ding X, Hu R. Cost-fair task allocation in mobile crowd sensing with probabilistic users. IEEE Trans Mob Comput. 2021;20(2):403–15. doi:10.1109/TMC.2019.2950921.
15. Zhu Q, Sarwar Uddin MY, Venkatasubramanian N, Hsu CH. Spatiotemporal scheduling for crowd augmented urban sensing. In: Proceedings of the IEEE INFOCOM 2018–IEEE Conference on Computer Communications; 2018 Apr 16–19; Honolulu, HI, USA. p. 1997–2005. doi:10.1109/INFOCOM.2018.8485869.
16. Zhang X, Yang Z, Gong YJ, Liu Y, Tang S. SpatialRecruiter: Maximizing sensing coverage in selecting workers for spatial crowdsourcing. IEEE Trans Vehicular Technol. 2017;66(6):5229–40. doi:10.1109/TVT.2016.2614312.
17. Toledo K, Kaschel H, Torres J. A stochastic approach for spectrum sensing and sensor selection in dynamic cognitive radio sensor networks. Phys Commun. 2019;37(5):100879. doi:10.1016/j.phycom.2019.100879.
18. Zhu X, Luo Y, Liu A, Tang W, Bhuiyan MdZA. A deep learning-based mobile crowdsensing scheme by predicting vehicle mobility. IEEE Trans Intell Transp Syst. 2021;22(7):4648–59. doi:10.1109/TITS.2020.3023446.
19. Hu W, Winter S, Khoshelham K. Forecasting fine-grained sensing coverage in opportunistic vehicular sensing. Comput Environ Urban Syst. 2023;100:101939. doi:10.1016/j.compenvurbsys.2023.101939.
20. Hadjkouider AM, Abdelaziz Kerrache C, Sahraoui Y, Adnane A, Calafate CT. A machine learning-assisted game theoretical service selection strategy in UAV networks. In: Proceedings of the Global Congress on Emerging Technologies (GCET-2024); 2024 Dec 9–11; Gran Canaria, Spain. p. 58–64. doi:10.1109/GCET64327.2024.10934408.
21. Bithas PS, Moustakas AL. Generalized UAV selection with distributed transmission policies. IEEE Trans Commun. 2023;71(2):741–56. doi:10.1109/TCOMM.2022.3229665.
22. Zheng K, Fu J, Liu X. Relay selection and deployment for NOMA-enabled multi-AAV-assisted WSN. IEEE Sens J. 2025;25(9):16235–49. doi:10.1109/JSEN.2025.3551916.
23. Bithas PS, Nikolaidis V, Kanatas AG, Karagiannidis GK. UAV-to-ground communications: channel modeling and UAV selection. IEEE Trans Commun. 2020;68(8):5135–44. doi:10.1109/TCOMM.2020.2992040.
24. Raju I, Roy A. FU-serve: fog-enabled UAV-as-a-service for IoT applications. In: Proceedings of the GLOBECOM 2023–2023 IEEE Global Communications Conference; 2023 Dec 4–8; Kuala Lumpur, Malaysia. p. 6401–6. doi:10.1109/GLOBECOM54140.2023.10437547.
25. Li C, Zhang Y, Yu L, Yang M. Efficient vehicle selection and resource allocation for knowledge distillation-based federated learning in UAV-assisted VEC. IEEE Trans Intell Transp Syst. 2025;26(5):6321–31. doi:10.1109/TITS.2025.3525735.

26. Zhu S, Gui L, Cheng N, Sun F, Zhang Q. Joint design of access point selection and path planning for UAV-assisted cellular networks. IEEE Internet Things J. 2020;7(1):220–33. doi:10.1109/JIOT.2019.2947718.
27. Bousbaa FZ, Lakas A, Rezigat AE, Benguettache HS, Lagraa N, Kerrache CA, et al. GTSS-UC a game theoretic approach for services' selection in UAV clouds. In: Proceedings of the 2021 IEEE/ACM 25th International Symposium on Distributed Simulation and Real Time Applications (DS-RT); 2021 Sep 27–29; Valencia, Spain. p. 1–8. doi:10.1109/DS-RT52167.2021.9576124.
28. Hadjkouider AM, Kerrache CA, Korichi A, Sahraoui Y, Calafate CT. Stackelberg game approach for service selection in UAV networks. Sensors. 2023;23(9):9. doi:10.3390/s23094220.
29. Mufalli F, Batta R, Nagi R. Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans. Comput Oper Res. 2012;39(11):2787–99. doi:10.1016/j.cor.2012.02.010.
30. Gao F. An integrated multi criteria decision making method using dual hesitant fuzzy sets with application for unmanned aerial vehicle selection. Sci Rep. 2025;15(1):12637. doi:10.1038/s41598-025-95981-0.
31. Asaamoning G, Mendes P, Rosário D, Cerqueira E. Drone swarms as networked control systems by integration of networking and computing. Sensors. 2021;21(8):8. doi:10.3390/s21082642.
32. Sritharan L, Nita L, Kerrigan E. An efficient method for maximal area coverage in the context of a hierarchical controller for multiple unmanned aerial vehicles. Eur J Control. 2023;74(9):100850. doi:10.1016/j.ejcon.2023.100850.
33. Yue W, Zhang X, Liu Z. Distributed cooperative task allocation for heterogeneous UAV swarms under complex constraints. Comput Commun. 2025;231(11):108043. doi:10.1016/j.comcom.2024.108043.
34. Zhao Z, Zhu B, Zhou Y, Yao P, Yu J. Cooperative path planning of multiple unmanned surface vehicles for search and coverage task. Drones. 2023;7(1):1. doi:10.3390/drones7010021.
35. Pintado A, Santos M. A first approach to path planning coverage with multi-UAVs. In: Herrero Á, Cambra C, Urda D, Sedano J, Quintián H, Corchado E, editors. 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020). Cham, Switzerland: Springer International Publishing; 2021. p. 667–77. doi:10.1007/978-3-030-57802-2_64.
36. Caillouet C, Giroire F, Razafindralambo T. Optimization of mobile sensor coverage with UAVs. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS); 2018 Apr 15–19; Honolulu, HI, USA. p. 622–7. doi:10.1109/INFCOMW.2018.8406980.
37. Xing S, Wang R, Huang G. Area decomposition algorithm for large region maritime search. IEEE Access. 2020;8:205788–97. doi:10.1109/ACCESS.2020.3037679.
38. Alqefari S, Menai MEB. Multi-UAV task assignment in dynamic environments: current trends and future directions. Drones. 2025;9(1):1. doi:10.3390/drones9010075.
39. Lee M. AMOVFLY-dataset. Xi'an, China: Northwestern Polytechnical University; 2025.
40. Abbes S, Rekhis S. Reinforcement learning for intelligent sensor virtualization and provisioning in internet of vehicles (IoV). IEEE Access. 2024;12:54352–70. doi:10.1109/ACCESS.2024.3387295.
41. Pathak N, Misra S, Mukherjee A, Roy A, Zomaya AY. UAV virtualization for enabling heterogeneous and persistent UAV-as-a-service. IEEE Trans Vehicular Technol. 2020;69(6):6731–8. doi:10.1109/TVT.2020.2985913.
42. Sanchez-Aguero V, Gonzalez LF, Valera F, Vidal I, López Da Silva RA. Cellular and virtualization technologies for UAVs: an experimental perspective. Sensors. 2021;21(9):3093. doi:10.3390/s21093093.
43. Li X, Zhang Y, Zhu Z, Yao Y, Zhou X. UbiCap: a capability-based run-time model for heterogeneous sensors management in ubiquitous operating system. In: Proceedings of the 14th Asia-Pacific Symposium on Internetware; 2023 Aug 4–6; Hangzhou, China. p. 134–43. doi:10.1145/3609437.3609448.
44. Saha D, Pattanayak D, Mandal PS. Surveillance of uneven surface with self-organizing unmanned aerial vehicles. IEEE Trans Mob Comput. 2022;21(4):1449–62. doi:10.1109/TMC.2020.3022075.
45. Zorbas D, Di Puglia Pugliese L, Razafindralambo T, Guerriero F. Optimal drone placement and cost-efficient target coverage. J Netw Comput Appl. 2016;75(3):16–31. doi:10.1016/j.jnca.2016.08.009.
46. Yanmaz E. Joint or decoupled optimization: multi-UAV path planning for search and rescue. Ad Hoc Netw. 2023;138(3):103018. doi:10.1016/j.adhoc.2022.103018.
47. Choi Y, Choi Y, Briceno S, Mavris DN. Energy-constrained multi-UAV coverage path planning for an aerial imagery mission using column generation. J Intell Robot Syst. 2020;97(1):125–39. doi:10.1007/s10846-019-01010-4.

48. Chen M, Liang W, Li J. Energy-efficient data collection maximization for UAV-assisted wireless sensor networks. In: Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC); 2021 Mar 29–Apr 1; Nanjing, China. p. 1–7. doi:10.1109/WCNC49053.2021.9417258.
49. Thibbotuwawa A, Nielsen P, Zbigniew B, Bocewicz G. Energy consumption in unmanned aerial vehicles: a review of energy consumption models and their relation to the UAV routing. In: Świątek J, Borzemski L, Wilimowska Z, editors. Information Systems Architecture and Technology: Proceedings of 39th International Conference on Information Systems Architecture and Technology-ISAT 2018. Cham, Switzerland: Springer International Publishing; 2019. p. 173–84. doi:10.1007/978-3-319-99996-8_16.
50. Cabreira TM, Ferreira PR, Franco CD, Buttazzo GC. Grid-based coverage path planning with minimum energy over irregular-shaped areas with UAVs. In: Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS); 2019 Jun 11–14; Atlanta, GA, USA. p. 758–67. doi:10.1109/ICUAS.2019.8797937.
51. Araújo JF, Sujit PB, Sousa JB. Multiple UAV area decomposition and coverage. In: Proceedings of the 2013 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA); 2013 Apr 16–19; Singapore. p. 30–7. doi:10.1109/CISDA.2013.6595424.
52. Vasquez-Gomez JI, Marciano-Melchor M, Valentin L, Herrera-Lozada JC. Coverage path planning for 2D convex regions. J Intell Robot Syst. 2020;97(1):81–94. doi:10.1007/s10846-019-01024-y.
53. Torres M, Pelta DA, Verdegay JL, Torres JC. Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. Expert Syst Appl. 2016;55(2):441–51. doi:10.1016/j.eswa.2016.02.007.
54. Zeng Y, Zhang R. Energy-efficient UAV communication with trajectory optimization. IEEE Trans Wirel Commun. 2017;16(6):3747–60. doi:10.1109/TWC.2017.2688328.