electronics

MDPI

Article

AGCo-MATA: Air-Ground Collaborative Multi-Agent Task Allocation in Mobile Crowdsensing

Lixin Yang [1], Kaixing Zhao [2,*], Tianhao Shao [2], Bohan Feng [2], Jian Di [3] and Zuheng Ming [4,5]

[1] Energy Development Research Institute, China Southern Power Grid, Guangzhou 510663, China; yanglx8@csg.cn
[2] School of Software, Northwestern Polytechnical University, Xi'an 310072, China; tianhaoshao@mail.nwpu.edu.cn (T.S.); 2022302946@mail.nwpu.edu.cn (B.F.)
[3] Institute of Advanced Technology, University of Science and Technology of China, Hefei 230026, China
[4] L2TI, Universite Sorbonne Paris Nord, 93430 Paris, France; zuheng.ming@univ-paris13.fr
[5] EmboMind Research, Tongxiang 314503, China
* Correspondence: kaixing.zhao@nwpu.edu.cn

Abstract

The rapid advancement of intelligent unmanned systems has brought new opportunities to mobile crowd sensing (MCS). Compared with traditional homogeneous frameworks, heterogeneous air-ground collaborative multi-agent frameworks consisting of unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) exhibit superior flexibility and efficiency in complex sensing tasks. Task allocation among agents is crucial for improving overall MCS quality. To achieve efficient task allocation for heterogeneous collaborative agents, this study investigated two typical complex multi-agent task allocation scenarios with dual optimization objectives: (1) For the ***Air-Ground Few-Agents-More-Tasks (AG-FAMT)*** scenario, the objectives are to maximize task completion and minimize total travel distance; (2) For the ***Air-Ground More-Agents-Few-Tasks (AG-MAFT)*** scenario (task allocation based on agent locations), the objectives are to minimize total travel distance and travel time cost. Overall, in this paper, we proposed two algorithms: a multi-task minimum cost maximum flow optimization algorithm called **Multi-Task Minimum-Cost Maximum-Flow (MT-MCMF)** tailored for AG-FAMT, and a multi-objective optimization algorithm called **Weighted Integer Linear Programming (W-ILP)** for AG-MAFT (with a focus on optimizing UAV charging path planning). Experiments on a large-scale real-world dataset demonstrated that both proposed algorithms outperform baseline methods under varying experimental settings (task quantity, difficulty, and distribution), providing a novel approach to enhance the overall quality of air-ground MCS tasks.

Keywords: mobile crowdsensing; unmanned aerial vehicle; unmanned ground vehicle; multi-agent framework; multi-task allocation

Academic Editor: Fernando De la Prieta Pintado

Received: 7 April 2026
Revised: 13 May 2026
Accepted: 18 May 2026
Published: 21 May 2026

1. Introduction

In the era of Internet of Things (IoT) and intelligent robots, the capability of MCS has been greatly enhanced [1], rendering it particularly pivotal for contemporary social governance [2]. Today, MCS has been widely deployed across diverse domains, including real-time traffic monitoring [3], precipitation observation [4], air quality assessment [5], and urban infrastructure development [6]. In recent years, although mobile devices such as smartphones are becoming common in many MCS tasks [7], their large-scale application still faces many challenges, especially when dealing with scenarios where human presence is difficult or impossible [8]. In this case, heterogeneous multi-agent systems, especially air–ground robot systems (including UAVs [9] and UGVs [10]) can collaboratively play a significant role in conducting real-time complex sensing tasks [11].

Although scenarios involving a single agent (for example, a moving camera based on an unmanned aerial vehicle) have been extensively studied before [11], their applications in more challenging scenarios are still limited due to restricted capabilities [12,13]. As unmanned technology advances rapidly, UAVs and UGVs are increasingly utilized collaboratively in numerous fields. This technological convergence has catalyzed the emergence of heterogeneous multi-agent frameworks, particularly the air–ground collaborative framework [14] that constitutes the focal point of this paper. Within such a framework, UAVs with their aerial mobility and rapid coverage capabilities can collaborate with UGVs, which possess higher load capacity and longer operational durations [15,16], to perform complex mobile sensing tasks. Compared to homogeneous robotic systems, heterogeneous multi-agent frameworks offer distinct advantages in terms of flexibility, efficiency of task allocation, and diversity of supported tasks, rendering them a promising and valuable research direction for mobile crowdsensing [17].

Simultaneously, both UAVs and UGVs encounter significant challenges when executing multiple missions. While UAVs exhibit high maneuverability and broad spatial coverage, their operational capabilities are constrained by limited battery life, which restricts both flight endurance and effective sensing range [18]. Conversely, UGVs typically offer extended operational endurance and greater autonomy. However, they are generally characterized by slower mobility and more limited sensing coverage [19]. In this context, integrating UAVs and UGVs into an air–ground collaborative framework presents a promising strategy to address these limitations and exploit their complementary capabilities. Such collaboration not only facilitates a more balanced trade-off between sensing coverage and sensing duration but also enhances the overall efficiency of MCS tasks [20,21]. Moreover, by leveraging diverse perspectives afforded through operation at varying angles and altitudes, air–ground collaboration can significantly improve the comprehensiveness of sensory data collection and thereby create new opportunities for the effective execution of sensing missions [22,23].

However, despite the growing body of research on air–ground collaborative frameworks in recent years [24,25], heterogeneous task allocation (i.e., one of the most critical mechanisms) has received relatively limited attention, particularly within the context of UAV–UGV (unmanned aerial vehicle–unmanned ground vehicle) collaboration [26]. Traditional task allocation approaches are predominantly designed for homogeneous systems, where agents are typically selected based on uniform criteria [27]. In contrast, the inherent heterogeneity of air–ground collaborative frameworks introduces significant complexity into the task allocation process, necessitating a more nuanced consideration of the distinct capabilities, constraints, and operational characteristics of each agent type [19]. Indeed, in resource-constrained ubiquitous systems, such as the air–ground collaborative framework under consideration, effective task allocation plays a pivotal role [28]: it not only enhances overall task completion efficiency and optimizes agent trajectories but also substantially reduces redundant movements across the multi-agent system, thereby conserving energy and computational resources.

Despite these efforts, existing studies still exhibit three fundamental limitations. First, most existing task allocation approaches treat task assignment and path planning as two separate problems, where agents are first selected and then routes are determined. This decoupled design fundamentally limits the optimality of task allocation, especially in multi-task scenarios. Second, prior work implicitly assumes a single task-agent configuration, without distinguishing between fundamentally different scenarios such as resource-scarce and resource-abundant settings. Third, current frameworks often consider only partial system components, lacking a holistic integration of sensing, allocation, and energy sustainability.

Importantly, the contribution of this work goes beyond a simple integration of existing optimization tools. Rather than focusing on designing a new optimization primitive, this paper introduces a fundamentally different modeling perspective for heterogeneous task allocation. Specifically, we reformulate the problem as a **task-set-oriented allocation model**, where each agent is assigned a set of tasks rather than independent tasks, thereby unifying multi-task assignment and path planning into a single optimization framework. This formulation departs from conventional one-to-one assignment models by explicitly capturing inter-task dependencies and execution-level coupling.

Furthermore, we propose a **scenario-driven modeling paradigm**, in which two representative but structurally distinct configurations—AG-FAMT and AG-MAFT—are explicitly identified and formulated separately. Finally, we develop an integrated air–ground collaborative framework that jointly considers **sensing, task allocation, and UAV charging coordination**.

Based on the above modeling perspective, we further identify two representative task-agent configurations [29]: the *few-agents-more-tasks* scenario and the *more-agents-few-tasks* scenario, denoted as **AG-FAMT** (Air-Ground Few Agents More Tasks) and **AG-MAFT** (Air-Ground More Agents Few Tasks), respectively (as shown in Figure 1).

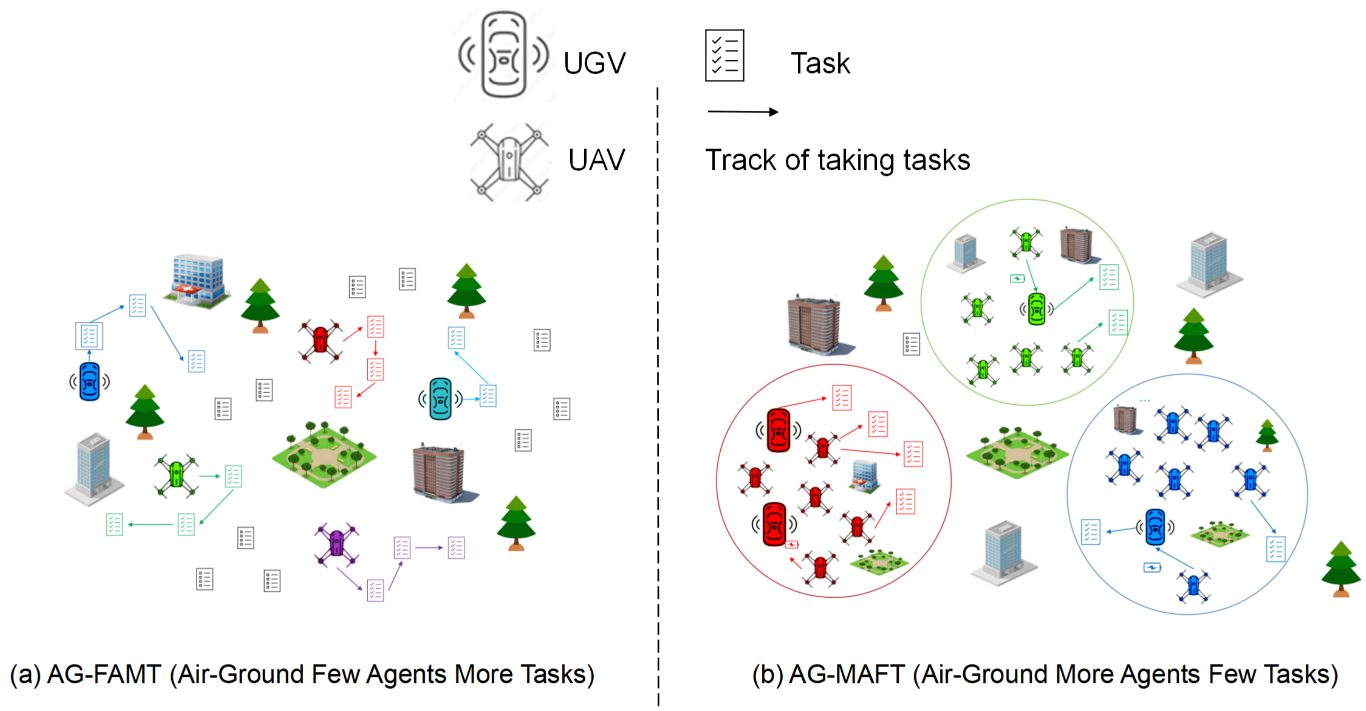

Figure 1. Two scenarios of agents selection.

In the **AG-FAMT** scenario, the number of tasks exceeds the available agents, requiring each agent to execute multiple tasks. Under this setting, the optimization objective is to maximize the number of completed tasks while minimizing the total travel distance, which naturally leads to a coupled combinatorial optimization problem involving **task-set selection** and path planning. In contrast, the **AG-MAFT** scenario considers the case where agent resources are relatively sufficient compared to the number of tasks. In this setting, each agent is typically assigned a single task, and the allocation focuses on selecting suitable agents under spatial proximity and capability constraints. The objective is to minimize both travel distance and execution time while ensuring task quality and preserving location privacy.

The main contributions of this paper are summarized as follows:

1. We develop a **task-set-oriented optimization framework** for heterogeneous UAV–UGV systems, in which each agent is assigned a set of tasks and the execution cost is evaluated jointly through integrated task selection and path planning.
2. We design two tailored optimization models for different task-agent configurations. For the AG-FAMT scenario, we formulate a combinatorial optimization model and solve it using an extended MCMF-based approach. For the AG-MAFT scenario, we construct a structured ILP model incorporating spatial-region constraints and capability matching.
3. We implement an integrated air–ground collaborative system that combines task allocation with UAV charging coordination, including a predictive charging trajectory planning mechanism to support long-term sensing tasks.
4. We conduct extensive experiments under diverse task distributions and system settings, demonstrating the effectiveness and robustness of the proposed framework compared with baseline methods.

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 presents the system model and problem formulation for the AG-FAMT and AG-MAFT scenarios. Section 4 details the performance evaluation. Sections 5 and 6 provide discussion and conclude the paper, respectively.

2. Related Work

2.1. Mobile Crowdsensing

MCS is a paradigm that leverages widely distributed mobile devices to collect spatiotemporal data by harnessing their integrated sensing, computing, and communication capabilities [30,31]. This approach enables large-scale monitoring and data acquisition across diverse domains, including environmental and social contexts. In recent years, MCS has garnered significant scholarly attention and spurred extensive research. For instance, Mohan et al. [32] demonstrated the feasibility of continuous environmental monitoring using low-cost mobile devices. Similarly, Mednis et al. [6] investigated real-time road surface monitoring to enhance urban surveillance capabilities. Furthermore, Pawar et al. [33] proposed a framework that integrates mobile sensors with neural networks, substantially improving the accuracy of real-time urban sensing systems.

Existing MCS platforms primarily focus on three core components: task publishing [34], data collection [35], and participant selection [36], often operating under single-task scenarios. However, as MCS technology advances, the problem of task allocation has emerged as a critical challenge that significantly affects both the efficiency and quality of MCS operations [35]. Effective task allocation must account for diverse participant capabilities, heterogeneous sensing requirements, spatiotemporal constraints, and resource limitations, which are both factors that are especially pronounced in multi-task environments.

Previously, Zhao et al. [37] proposed an approach that enhances data accuracy and integrity by reducing the communication overhead between aggregation nodes. This method demonstrates stronger privacy-preserving capabilities compared to conventional sensing auction models. Nevertheless, like most existing solutions, previous methods often remain limited in their ability to jointly optimize task assignment, resource utilization, and privacy guarantees within dynamic, large-scale, and heterogeneous MCS ecosystems [38]. Consequently, there is a pressing need for novel frameworks that can holistically address these intertwined challenges—particularly in emerging paradigms such as air–ground collaborative crowdsensing, where UAVs and UGVs introduce additional layers of complexity in collaboration, coverage, and energy-aware scheduling [39]. This paper seeks to bridge this gap by proposing an integrated task allocation mechanism tailored for heterogeneous multi-agent MCS systems, with explicit consideration of system efficiency, scalability, and privacy preservation.

2.2. Air-Ground Collaborative Systems

As MCS faces growing challenges in complex environments, the limited adaptability of homogeneous robotic systems—characterized by uniform agent capabilities—hinders their ability to meet diverse task requirements [40]. To address these limitations, recent advances in multi-agent technologies and the proliferation of multi-agent frameworks across domains have spurred interest in air–ground collaborative architectures [41]. Such frameworks synergistically integrate UAVs and UGVs to enhance MCS efficiency, data accuracy, and environmental mapping fidelity: UAVs enable large-scale, rapid data collection from above, while UGVs provide complementary ground-level sensing and support, thereby improving overall perception performance [42,43]. For instance, Sun et al. [22] proposed a UAV-based data collection method for Internet of Things (IoT) networks that optimizes flight speed, hovering position, and bandwidth allocation. Meanwhile, Wei et al. [23] improved UAV navigation and mission efficiency through an incentive mechanism for path planning in complex environments.

Meanwhile, task allocation within multi-agent frameworks continues to face significant challenges. As the scale of such systems grows and the diversity of heterogeneous agents increases, traditional centralized task assignment approaches encounter substantial limitations, particularly in terms of computational complexity and communication overhead [44]. In response, distributed task allocation algorithms have become a prevalent strategy in heterogeneous multi-agent systems, allowing individual agents to make allocation decisions based on local information, including their own state and task requirements [45]. This paradigm not only mitigates the computational burden inherent in centralized scheduling but also improves the overall scalability and robustness of the system [46,47]. Furthermore, game-theoretic approaches to task allocation have attracted considerable attention in the literature due to their ability to model strategic interactions among self-interested or cooperative agents [48,49]. Despite these methodological advances, human interventions remain indispensable for providing real-time feedback, handling anomalous situations, and ensuring the validity and reliability of collected data [50,51].

2.3. Task Allocation

Task allocation, defined as the process of assigning tasks to suitable executors, plays a pivotal role in MCS, as the choice of executors directly affects both the efficiency and quality of task execution [52]. In the context of crowdsourcing, two predominant paradigms for task assignment have emerged: (1) executor-initiated autonomous selection [53] and (2) centralized allocation orchestrated by a coordinating server [54]. Liu et al. [29] examined the influence of multiple factors on multi-task assignment within mobile crowds, with a particular focus on three critical dimensions: the number of participants, the volume of tasks, and the urgency associated with each task. In a related study, Song et al. [55] tackled the heterogeneous Quality of Information (QoI) requirements inherent in spatial crowdsourcing tasks and proposed a multi-task allocation strategy designed to identify a minimal set of workers capable of satisfying the QoI demands of concurrent tasks while remaining within a prescribed total budget. Furthermore, the ActiveCrowd [56] framework investigated the problem of multitask workload scheduling under both intentional and unintentional mobility scenarios, thereby accommodating time-sensitive tasks and delay-tolerant tasks, respectively.

The preceding analysis explicitly highlights the multiple advantages inherent in air–ground collaborative frameworks. In the present study, a multi-task allocation framework integrating UAVs and UGVs is proposed, with a specific focus on the correlation between task quantity and agent number.

Recent studies have emphasized the importance of robustness and generalization under distribution shifts. For example, Chen et al. [57] propose a domain generalization framework that learns category-invariant disentangled features to improve performance across varying operating conditions.

More broadly, domain generalization methods aim to learn representations that remain effective across unseen environments. Muandet et al. [58] introduce invariant feature learning to reduce domain discrepancy, while Ganin et al. [59] propose domain-adversarial training to improve robustness against distribution shifts. These works highlight the importance of designing systems that can generalize across diverse and dynamic environments.

In the context of mobile crowdsensing and air–ground collaborative systems, task distributions and environmental conditions may vary significantly across different scenarios. While our work focuses on a modeling-level reformulation of heterogeneous task allocation, incorporating robustness and generalization into the proposed framework remains an important direction for future research.

3. System Model and Problem Description

3.1. Unified Framework Overview

To address the heterogeneous task allocation problem in air–ground collaborative systems, we propose a unified framework that integrates task allocation and energy sustainability under different task-agent configurations.

Figure 2 provides an overview of the proposed unified framework. As illustrated in Figure 2, the framework consists of three tightly coupled components:

(1) **Scenario-aware task allocation module:** The system distinguishes between AG-FAMT and AG-MAFT based on the task–agent relationship. For AG-FAMT, a task-set-oriented combinatorial optimization model is adopted to jointly optimize task assignment and path planning. For AG-MAFT, a constraint-aware allocation model incorporating spatial-region and capability constraints is employed.

(2) **Heterogeneous agent coordination:** The framework leverages the complementary characteristics of UAVs and UGVs, including mobility, sensing capability, and endurance, to enable efficient cross-platform task execution.

(3) **UAV charging support mechanism:** To support long-term task execution, a predictive charging trajectory planning (PCTP) mechanism is introduced, where UGVs dynamically provide charging services to UAVs.

These components are tightly integrated within a unified optimization framework rather than treated as independent modules. In particular, the charging mechanism is inherently coupled with the AG-MAFT scenario, where long-term sensing tasks impose strict energy sustainability requirements.

Based on the above unified framework, we next present the system model and formulate the corresponding optimization problems.

3.2. System Model

As shown in Figure 1, we consider a heterogeneous multi-agent air-ground collaborative MCS framework involving obstacles (e.g., buildings), UAVs, UGVs, task points, and boundary areas. Let the set of UAVs be denoted as $X = \{x_1, x_2, \ldots, x_s\}$, the set of UGVs as $Y = \{y_1, y_2, \ldots, y_t\}$, and the overall agent set as $U = \{X + Y\} = \{x_1, x_2, \ldots, x_s, y_1, y_2, \ldots, y_t\}$. The task set is defined as $T = \{t_1, t_2, \ldots, t_n\}$, where the number of UAVs is s, the number of UGVs is t, the number of tasks is N and the total number of agents $M = s + t$. Obstacles are represented as $Ob = \{ob_1, ob_2, \ldots, ob_a\}$, with a total of a obstacles. The agents and tasks are distributed within a physically bounded area.

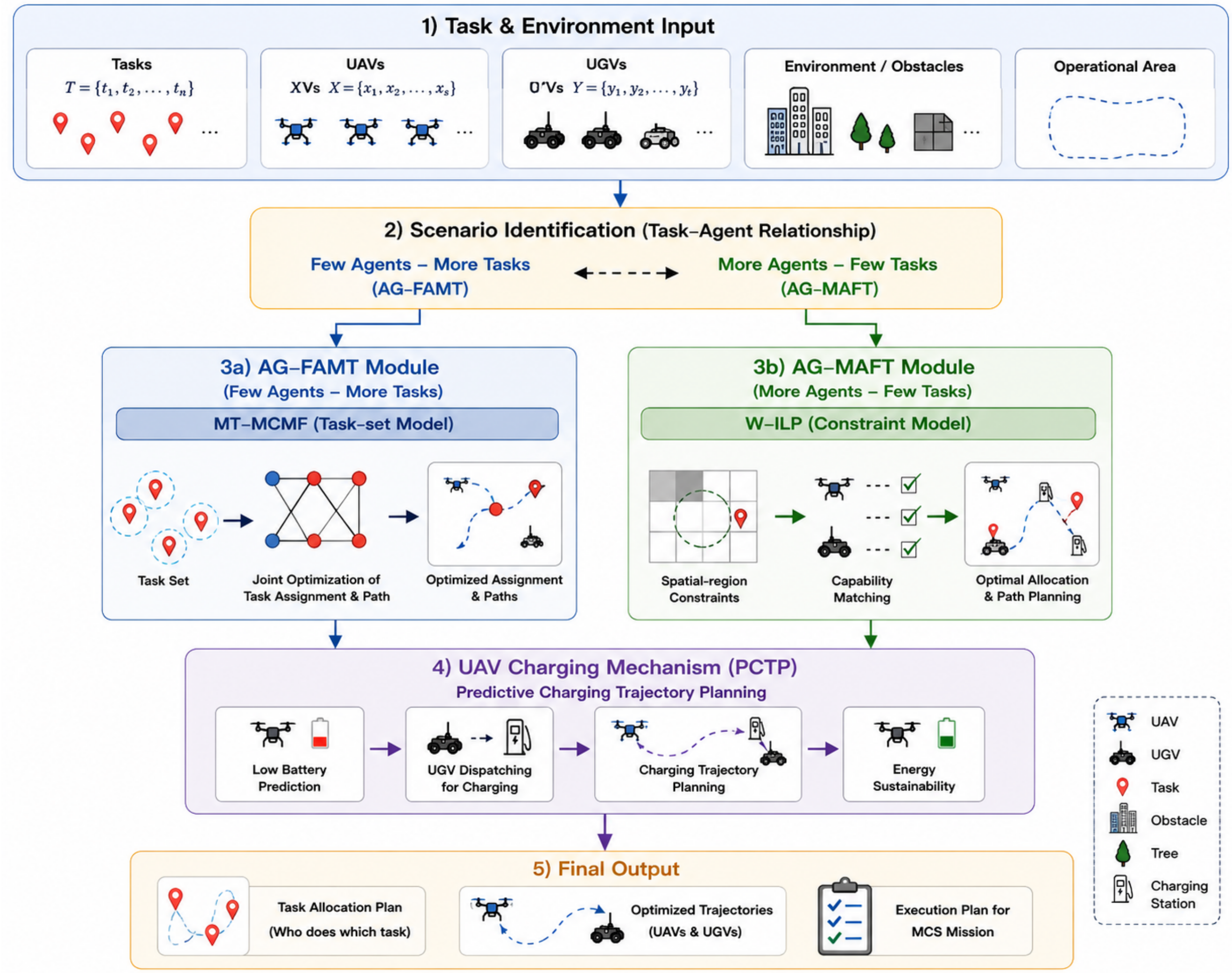

Figure 2. A unified framework for heterogeneous air–ground collaborative task allocation integrating scenario-aware optimization and energy sustainability mechanisms.

The basic information for each task can be expressed as a tuple $< (x_i, y_i, h_i), c_{i1}, \ldots, c_{in} >$, with UAV information given by $< (x_j, y_j, h_j), c_{j1}, \ldots, c_{jn} >$, and UGV information as $< (x_k, y_k, h_k), c_{k1}, \ldots, c_{kn} >$. This information includes location details, represented in a 3D Cartesian coordinate system (x, y, h). The set $\{c_{i1}, c_{i2}, \ldots, c_{in}\}$ represents the required capability levels across types for task t_i $(1, 2, \ldots, n)$, while $\{c_{j1}, c_{j2}, \ldots, c_{jn}\}$ and $\{c_{k1}, c_{k2}, \ldots, c_{kn}\}$, respectively, denote the capability levels of UAVs and UGVs for each capability type $(1, 2, \ldots, n)$.

3.3. Problem Description

3.3.1. AG-FAMT (Air-Ground Few Agents More Tasks) ($N > M$)

In the AG-FAMT (Air-Ground Few Agents More Tasks) scenario, the quantity of tasks is greater than that of available agents, which frequently involves time-critical tasks. A typical example is the urban environmental sensing mission carried out after severe weather events, such as typhoons—specifically, the monitoring of water level fluctuations and the assessment of structural damage to buildings as well as vegetation (e.g., trees) in the aftermath of such disasters. Each agent is required to undertake multiple tasks sequentially or concurrently, and must navigate to predefined target locations to fulfill these assigned tasks. Under such circumstances, travel distance emerges as a crucial influencing factor: a shorter travel distance between task locations can effectively reduce the overall task completion cycle and improve mission efficiency.

In this context, the optimization objectives are defined as maximizing the total number of tasks successfully completed and minimizing the cumulative travel distance of all agents. Considering the urgency of the tasks and the scarcity of agent resources, incentive costs and the potential risk of leakage of agents' precise location information are excluded from the scope of consideration in this scenario. To improve the quality and reliability of sensing tasks, it is necessary for multiple agents to collaborate in the execution of each individual task. Meanwhile, an upper limit is set on the number of agents permitted to participate in a single task, so as to prevent data redundancy caused by excessive agent involvement. Based on the above consideration, for AG-FAMT, each agent is necessary to complete multiple tasks. In order to balance task quality and agent fairness, a task upper limit of $q_i, (i = 1, 2)$ is set for each type of agent, where q_1 is the task upper limit of UAV and q_2 is the task upper limit of UGV. We assume that the travel time is proportional to the travel distance, thus transforming the goal of minimizing the task completion time into the goal of minimizing the travel distance.

3.3.2. AG-MAFT (Air-Ground More Agents Few Tasks) ($N < M$)

In contrast to the AG-FAMT scenario, the AG-MAFT (Air-Ground More Agents Few Tasks) scenario is characterized by an excess of agents relative to tasks. More specifically, in this scenario, some tasks do not demand the extensive participation of a large number of agents, whereas others require uninterrupted execution—typical examples include air quality continuous monitoring and ongoing urban safety patrol missions. For the AG-MAFT scenario, given the sufficiency of agent resources, agents can be allocated in a flexible manner to enhance the execution quality of tasks. In this context, each agent is only required to undertake one single task, and a preliminary inspection is conducted beforehand to verify that each agent meets the minimum competency criteria for performing the assigned task. Owing to the non-urgent nature of the tasks in this scenario, precise location data of the agents is not a prerequisite. Instead, tasks can be selected within a predefined range around each agent's approximate location, thereby safeguarding the agents' location privacy.

This study classifies tasks according to distance and spatial requirements. Given that UAVs can operate at high speeds, they can greatly expand the perception scope of long-distance tasks. In contrast, UGVs are more suitable for close-range tasks, which can reduce the travel distance and improve the response efficiency. To ensure task success, task allocation optimization and selection of agents meeting the minimum execution thresholds are essential. In the scenario of AG-FAMT, we strive to maximize the number of completed tasks while minimizing the total travel distance. In the AG-MAFT scenario, we aim to complete all tasks while minimizing the agents' travel time cost and the total travel distance. Moreover, in the AG-MAFT scenario, where long-term and continuous tasks are prevalent, energy sustainability becomes a critical constraint. Therefore, we integrate a UAV charging support mechanism into the allocation framework to ensure continuous task execution.

3.4. Problem Formulation

Based on the unified framework described above, we next formulate the task allocation problems under two representative configurations, namely AG-FAMT and AG-MAFT.

3.4.1. AG-FAMT Formulation

Within a defined area, given a set of agents $U = X + Y = \{x_1, x_2, \ldots, x_s, y_1, y_2, \ldots, y_t\}$, assume that the task execution limits for heterogeneous robots are $q = q_1, q_2$ (where q_1 is the task limit for UAVs and q_2 is the task limit for UGVs). The set of published tasks is $T = \{t_1, t_2, \ldots, t_n\}$, where each task t_i can be assigned to at most p_i agents (assignment to UAVs is represented by x_{ij}, and assignment to UGVs by y_{ik}). We use $UT_i = \{x_{i1}, x_{i2}, \ldots, y_{i1}, y_{i2}\}$ to denote the set of agents assigned to task t_i, and $TU_j = \{x_{1j}, x_{2j}, \ldots\}$ or $TU_k = \{y_{1k}, y_{2k}, \ldots\}$ to, respectively, represent the set of tasks assigned to UAV x_j and UGV y_k. Let S denote the maximum movable distance for an agent. Considering the endurance limitations of agents, the travel distance required to execute each task must be evaluated before task assignment. Additionally, $D(TU_j)$ and $D(TU_k)$ represent the total travel distance for completing the set of tasks assigned to each UAV and UGV, respectively. Therefore, AG-FAMT problem can be formulated using Equations (1)–(6):

$$\sum_{i=1}^{q} \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (h_j - h_i)^2} = D(TU_{j/k}) \tag{1}$$

$$maximize \sum_{i=1}^{m} |TU_{j/k}| \tag{2}$$

$$minimize \sum_{i=1}^{m} D(TU_{j/k}) \tag{3}$$

$$|TU_{j/k}| = q \; (q = q_1, q_2), (1 \leq j \leq s, 1 \leq k \leq t) \tag{4}$$

$$|UT_i| <= p_i \; (1 \leq i \leq n) \tag{5}$$

$$D(TU_{j/k}) \leq S \tag{6}$$

The AG-FAMT problem presents two primary challenges: first, ensuring that each agent can complete multiple tasks, and second, achieving the dual optimization objectives of maximizing the total number of completed tasks and minimizing the agents' total travel distance. Addressing the first challenge involves selecting multiple tasks for each agent, which is a complex dynamic process. In the AG-FAMT scenario, not only is there a combinatorial problem between tasks and agents, but the execution path for each agent must also be planned to achieve the shortest travel distance. Given the combination of task allocation and path optimization, finding a comprehensive algorithm to solve the AG-FAMT problem is challenging. Therefore, we plan to calculate the set of all executable tasks for each agent and seek an optimal solution to the combinatorial optimization problem, which could be an effective approach. For the second challenge, achieving both optimization objectives simultaneously is infeasible, as they are often contradictory. We thus try to balance the objectives of maximizing task completion and minimizing total travel distance to find a compromise solution.

On the basis of analyzing the AG-FAMT problem, we adopt the Minimum Cost Maximum Flow (MCMF) model for solving it [60]. The MCMF model is a classic method for solving dual objective optimization problems. We transformed the AG-FAMT problem into an MCMF problem and constructed a new MCMF model called MT-MCMF by setting various constraints to achieve effective task allocation and path planning. This method can help ensure the completion of tasks while minimizing the total travel distance of agents, thereby optimizing overall system performance.

3.4.2. Algorithm: MT-MCMF

The MT-MCMF (Multi Task Minimum Cost Maximum Flow) algorithm is based on MCMF theory, with the goal of maximizing the number of completed tasks while minimizing the total travel distance of a given set of agents and tasks. The core of MCMF model is to find the optimal path set that generates the maximum traffic at the minimum cost. We transform the AG-MAFT problem into an MCMF problem, where the total travel distance represents cost and the total number of completed tasks represents flow. We combine the minimum cost and maximum flow problem with the goals of minimizing travel distance and maximizing task completion.

In the MT-MCMF model graph, each edge has a capacity, flow, and cost. The capacity of an edge denotes the maximum flow it can carry, representing the maximum number of tasks. However, several challenges arise with direct application of the MT-MCMF model: (1) the flow network cannot consist solely of agent and task nodes; (2) representing each agent's path with q tasks is complex; (3) the distinct requirements of agents and tasks need clear representation within the flow network. To address these issues, we restructured the MT-MCMF model as illustrated in Figure 3 and made the following three modifications:

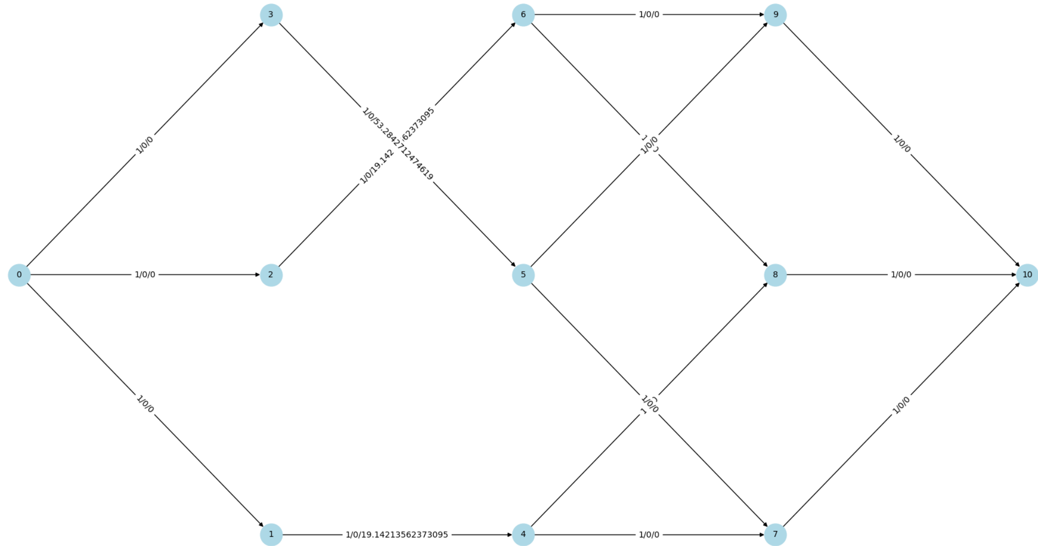

Figure 3. MT-MCMF model of AG-FAMT problem.

(1) Source and sink nodes: We introduced source and sink nodes into the flow network. For edges between the source node and agent nodes, each edge has a capacity of q_i $(i = 1, 2)$, reflecting the q_i tasks that each agent must complete. The cost of these edges is set to zero, as they solely carry incoming flow without involving move distance. Additionally, the flow from agent nodes through task nodes into the sink node reflects task completion. The total maximum flow of the sink node is set to $n \times p$, as each task can be performed by up to p agents. Each edge between the task nodes and the sink node has a capacity of p and a cost of zero. The actual flow to the sink node represents the total number of agents completing each task.

(2) Task sets and secondary nodes: We pre-enumerate all possible task sets, where each set contains q tasks selected from the total N tasks, yielding a total of C_N^q task sets. Each agent can select any task set to perform. We first calculate the shortest path from each agent to every individual task in the task set, and assign the length of each path as the cost of the corresponding edge in the network. Notably, the task sets constructed in the flow network do not predefine an explicit order for task completion. To address this characteristic, we draw on the classical solution paradigm of the Traveling Salesman Problem (TSP), and precompute the shortest paths between each agent and all tasks within a given task set. Consequently, the edge cost connecting agent nodes to task set nodes is defined as the TSP-based travel distance required for a single agent to complete all tasks in the corresponding set.

(3) Tertiary nodes for task completion status: We set different capacities on edges to meet the specific requirements of agents and tasks. Assuming that each task set can only be executed by one agent, we set the capacity of edges between task nodes and the sink node to p, as each task can be performed by up to p agents.

With the aforementioned improvements, the MT-MCMF model can be leveraged more effectively to address the AG-MAFT problem, which enables the completion of the maximum number of tasks while minimizing the total travel distance of all agents, thus achieving an optimized performance of the entire multi-agent system.

Overall, the MT-MCMF algorithm (see Algorithm 1) consists of two main components: constructing the flow network within the MT-MCMF model and finding the optimal solution in the flow network. Given an agent set with M agents and a total of N tasks, each agent is required to complete q tasks. Therefore, these tasks can be grouped into C_N^q task sets, where C_N^q represents the number of combinations of choosing q tasks from N tasks, thus forming the task set TT. Consequently, there are a total of $M \times C_N^q$ possible TSP problems for all task-execution sets. Based on our assumptions, the constructed flow network $G = (V, E, C, W)$ includes nodes V, edges E, capacities C for each edge, and total costs W for each edge. The process of obtaining the optimal solution from the flow network constructed by MT-MCMF is as follows: First, initialize the flow f to zero, then iteratively find an augmenting path until no further paths are available, and return the final flow f. Note that the flow of edges between agent nodes and task set nodes can be zero, indicating that the agent did not choose to execute that task set. At the same time, the flow value q means that the agent has carried out q tasks from that specific task set TT. Finally, the agent set $Agents$ and its corresponding completed task set TT are outputted. Through these steps, the MT-MCMF algorithm effectively optimizes task allocation among agents in the flow network, achieving the goals of maximizing task completion and minimizing total travel distance.

Algorithm 1 MT-MCMF

Input: The agent set $A(M)$ and the task set $T(N)$.
Output: The selected AT set and the corresponding minimum distance d.

Connect the source node to each agent node (from 0 to M), where the edge cost $c = 0$ and the flow upper limit is q.
Create $C(n, q)$ transfer sets TT. Connect each agent node to the task set (from M to $M + C_N^q$)), with edge cost c equal to the minimum distance (calculated using TSP to obtain the minimum distance).
Link each TT set to each task node, adding connections from $M + C_N^q$ to $M + C_N^q + N$, and then connect all these nodes to the sink node.
Create the flow network (V, E, C, W).
Initialize the flow $f = 0$.
while there exists an augmenting path in the residual network G_f **do**
  Select the augmenting path p^* with minimum cost.
  Set $c_f(p^*) = q$.
  Augment the flow f along p^* with $c_f(p^*)$.
end while
Return the agent-task assignment A-T.
Output the assignment A-T and the total minimum distance.

Complexity Analysis and Practical Considerations

The computational complexity of the MT-MCMF formulation is dominated by two components: (i) the enumeration of task sets, and (ii) the evaluation of corresponding TSP-based paths. Specifically, the number of candidate task sets is C_N^q, which leads to a combinatorial growth in complexity with respect to the task number N

and the task set size q. Accordingly, the overall time complexity can be expressed as $O(m \times C_N^q \times q^3) + O(m \times q \times (m+n) + C_N^q)$, where the first term corresponds to the TSP-based path computation and the second term corresponds to solving the flow network.

However, in practical MCS scenarios, the parameter q is typically small, since each agent can only execute a limited number of tasks within a given time window due to physical and operational constraints. Moreover, the number of available agents is usually moderate, which further restricts the effective search space. Under such settings, the proposed MT-MCMF model remains computationally feasible.

To further improve scalability, several optimization strategies can be considered. First, candidate task sets can be pruned based on spatial proximity or feasibility constraints, thereby reducing the number of combinations to be evaluated. Second, heuristic or sampling-based approaches can be employed to approximate the optimal task set selection without exhaustively enumerating all possibilities. Third, parallel computation can be utilized to accelerate the evaluation of task sets and shortest paths.

Importantly, the combinatorial formulation is essential for capturing task interdependencies and path coupling, which cannot be modeled by conventional assignment-based approaches. Therefore, while the worst-case complexity of the MT-MCMF formulation is combinatorial, the proposed method remains practical for typical MCS scenarios with bounded task-set sizes and moderate system scales.

3.4.3. AG-MAFT Formulation

Given a set of predefined sensing range, represented as $A = \{A_1, A_2, \ldots, A_i, \ldots, A_m\}$, each zone includes a group of agents regularly active in that area, denoted as $A_i = \{a_1, a_2, \ldots\}$. Additionally, there are several sensing tasks represented by $T = \{t_1, \ldots, t_j, \ldots t_n\}$, where each task t_j is assigned to p_j agents. Note that D_{ij} represents the distance between agents in zone A_i and task t_j. Furthermore, x_{ij} denotes the number of agents in zone A_i assigned to complete task t_j. Therefore, the AG-MAFT problem can be represented by Equations (7)–(10):

$$minimize \sum_{i=1}^{m} \sum_{j=1}^{n} D_{ij} \times x_{ij} \tag{7}$$

$$\sum_{j=1}^{n} x_{ij} \leq |A_i| \; (i \leq j \leq m) \tag{8}$$

$$\sum_{i=1}^{m} x_{ij} = p_{ij} (i \leq j \leq n) \tag{9}$$

$$x_{ij} \in Z^n \tag{10}$$

Given the sufficient agent resources under AG-MAFT, we set higher standards for task completion quality, allowing agents to execute tasks only if all capability requirements exceed the task requirements. The capability calculation formulas are shown in Equations (11) and (12):

$$eff_{ij} \; = \; (\frac{c_{i1}}{c_{j1}} + \frac{c_{i2}}{c_{j2}} + \ldots + \frac{c_{in}}{c_{jn}}) \times co_{ij} \tag{11}$$

$$co_{ij} \; = \begin{cases} 2 \, , \, if \; \forall \, 1 \leq k \leq n : c_{ik} > c_{jk} \\ \frac{1}{2} \, , \, otherwise \end{cases} \tag{12}$$

The main challenge in the AG-MAFT scenario involves two conflicting optimization goals: minimizing movement time costs and travel distances. In order to solve the AG-MAFT problem, we adopt a multi-objective optimization model. Briefly, the main approach to solve a multi-objective optimization problem is to transform it into a single-objective problem. We use linear weighting to achieve this.

3.4.4. Algorithm: Linear Weighting Method

The core idea of the linear weighting method lies in converting a multi-objective optimization problem into a composite objective function by assigning distinct weights to each individual objective, thereby simplifying the optimization process. A critical consideration is that time cost and distance are measured on disparate scales, rendering their direct summation infeasible. The transformation of the AG-MAFT problem into a single-objective optimization problem via the linear weighting method encompasses two key steps: first, normalizing the original objective functions to eliminate scale discrepancies; second, determining the weight coefficients for each objective to reflect their relative importance in the overall optimization framework. The scaling model we use is shown in Equation (13):

$$f'(x) = \frac{f(x) - f^{min}}{f^{max} - f^{min}} \tag{13}$$

In this model, $f(x)$ represents the value of the objective function, while f^{max} and f^{min}, respectively, represent the maximum and minimum values of the objective function.

More precisely, since the objectives of move time cost and travel distance belong to different dimensions, we use two weights, k_t and k_d, to represent the weights for move time cost and travel distance, respectively. By applying weighted summation to the normalized objective functions, we formulate a single-objective optimization problem. Consequently, the objective function for AG-MAFT is transformed into Equation (14):

$$\begin{aligned} \text{minimize} \quad & k_t \times \frac{\sum_i T_i \cdot \sum_j x_{ij} - T_{min}}{T_{max} - T_{min}} \\ & + k_d \times \frac{\sum_i \sum_j D_{ij} \cdot x_{ij} - D_{min}}{D_{max} - D_{min}} \end{aligned} \tag{14}$$

Clearly, the AG-MAFT problem is an Integer Linear Programming (ILP) problem, and we use the branch-and-bound method for ILP to obtain the optimal solution for the weights. Based on the above analysis, we also apply the K-means clustering algorithm for preliminary task classification.

The W-ILP algorithm (see Algorithm 2) is used to solve the AG-MAFT problem and consists of two parts: the first part calculates the maximum and minimum values of time and distance, and applies the linear weighting method to generalize the new single-objective optimization problem; the second part uses the branch-and-bound method to solve this problem. The W-ILP algorithm greedily selects agents with the shortest travel time, meaning agents with faster movement speeds are chosen to perform tasks to minimize move time, resulting in a potential increase in total distance traveled. Additionally, when considering total travel distance, each task should be performed by the closest agent to minimize overall distance, disregarding time cost.

Through these steps, the AG-MAFT problem is successfully transformed into a single-objective optimization problem, and further into an integer linear programming problem, where the optimal solution is obtained via the branch-and-bound method. W-ILP first generates a rooted tree based on the relaxed solution of the ILP problem, determining upper and lower bounds for each current solution. Subsequently, the original optimal solution is continuously adjusted until the integer constraints for the variables are satisfied, involving three steps: branching, bounding, and pruning. Finally, W-ILP outputs the number of agents per task in each area.

Algorithm 2 W-ILP

Input: The agent set $A(M)$ and the task set $T(N)$.
Output: Agent-task allocation and minimum total distance and time cost.

1: **Identify Task-Intensive Regions:** Define task execution regions and assign existing agents with capabilities and communication abilities.
2: **Agent-Task Matching:** Minimize the weighted sum of time and distance for assignments:

$$\text{minimize} \quad \sum_{i,j} (k_t \cdot \text{time}_{ij} + k_d \cdot \text{distance}_{ij}) x_{ij}$$

subject to:

$$\sum_j x_{ij} = p_i, \quad \sum_i x_{ij} \leq 1, \quad x_{ij} \in \{0,1\}$$

After assignment, UAVs that receive tasks can move toward the task locations.
3: **Establish UGV-UAV Sets:** Group UAVs around UGVs using K-means or nearest-distance methods. Assign each UAV to a UGV for recharging.
4: **Return UAVs for Charging:** After tasks are completed, UAVs return to UGVs for recharging.

To improve task allocation efficiency, this paper also applies the K-means clustering algorithm for preliminary task classification. The K-means clustering algorithm groups tasks into clusters based on the location information of UAVs and UGVs, so that tasks within each cluster are geographically close, thereby reducing the travel distance for agents.

3.4.5. UAV Charging Mechanism (PCTP)

Within the proposed unified framework, the UAV charging mechanism serves as a critical component to support long-term task execution, particularly in the AG-MAFT scenario. In such scenarios, UAVs are frequently deployed for continuous sensing tasks, where uninterrupted operation is essential. However, the limited battery capacity of UAVs poses a significant constraint on long-duration missions. Once the battery is depleted during task execution, task interruption and degraded sensing quality may occur.

To address this issue, we incorporate a UAV charging support mechanism into the unified framework, enabling dynamic energy replenishment during task execution. This mechanism ensures continuous operation of UAVs and enhances the robustness of long-term sensing tasks. Consequently, the charging process is not treated as an independent module but is tightly integrated into the AG-MAFT scenario within the unified framework.

Design Motivation and Rationale

The UAV–UGV charging problem can be formulated as a dynamic rendezvous optimization problem, where UAVs with limited energy must meet mobile UGV charging stations under time and energy constraints. Solving this problem optimally requires jointly considering multiple factors, including UAV trajectories, UGV movement, energy consumption, and time-dependent interactions among multiple UAVs and UGVs.

However, obtaining an exact optimal solution is computationally intractable in practice, as it involves a high-dimensional and dynamically coupled optimization process. Therefore, instead of pursuing a computationally expensive global optimization, we adopt a heuristic trajectory planning strategy that balances solution quality and computational efficiency.

Building upon the above motivation, we design a Predictive Charging Trajectory Planning (PCTP) algorithm, where each UGV is responsible for charging multiple UAVs [42] to improve resource utilization and efficiency. This centralized charging strategy enables adaptive charging decisions and reduces task interruptions, thereby enhancing overall system efficiency.

The PCTP algorithm is designed based on three key factors: (i) spatial distance between UAVs and UGVs, (ii) UAV energy levels, and (iii) relative motion dynamics. By integrating these factors into a unified trajectory adjustment mechanism, the algorithm enables the UGV to adaptively move toward multiple UAVs requiring charging, thereby approximating an efficient rendezvous solution.

Specifically, the direction of UGV movement is determined by the vector aggregation of the relative positions of UAVs approaching for charging. The movement speed is adjusted according to UAV flight speed, remaining energy, and relative distance. The velocity components are designed to be inversely proportional to the UAV flight speed v_{UAV_i} and remaining energy E_{UAV_i}, and directly proportional to the initial distance d_{UAV_i}. This design reflects the intuition that UAVs with lower remaining energy and larger distances should be prioritized in the rendezvous process, thereby reducing the risk of task interruption.

Although the PCTP algorithm does not guarantee global optimality, it provides a computationally efficient and practically effective approximation for dynamic UAV–UGV rendezvous. Such a design is particularly suitable for real-time and large-scale air–ground collaborative systems, where exact optimization is often infeasible.

The PCTP algorithm calculates the energy consumption required for a UAV to move to its task location based on the UAV's current position and task location. This energy consumption is proportional to the travel distance, flight time, and task execution time. Since each UGV in a given area is responsible for charging multiple UAVs, UAVs must autonomously move to the UGV's location for charging. When a UAV's battery reaches a critical level during task execution, it communicates with the UGV to calculate their distance and reserves enough energy to fly back to the UGV for charging.

During the period when the UAV is approaching the UGV for charging, the UGV calculates and adjusts its direction and speed based on the movement speed and trajectory of the UAV it is responsible for. Specifically, the direction of movement of the UGV is determined by the vector sum of coordinates between its current position and the UAVs flying towards it for charging. The movement speed in each direction is determined by the flight speed of the UAV, the remaining battery, and the initial distance. The velocity components of the UGV in each direction are inversely proportional to the flight speed v_{UAV_i} of the UAV and the remaining battery E_{UAV_i}, and directly proportional to the initial distance d_{UAV_i}. The final movement speed of the UGV is the sum of the velocity vectors in these directions. The UGV moves in the calculated direction and at the designated speed, while the UAV approaches the UGV until the distance between them is less than the preset charging distance, at which point charging begins. The trajectory of the UAV charging the UGV is shown in Figure 4.

The initial position of the UGV is (X, Y), and the initial position of each UAV_i is (x_i, y_i). The movement direction $\mathbf{D}_{UGV}$ of the UGV is determined by the sum of the position vectors between it and each UAV_i:

$$\mathbf{D}_{UGV} = \sum_i ((x_i - X), (y_i - Y)) \tag{15}$$

Then, $\mathbf{D}_{UGV}$ is normalized to obtain the unit direction vector $\hat{\mathbf{D}}_{UGV}$:

$$\begin{aligned} \hat{\mathbf{D}}_{UGV} &= \frac{\mathbf{D}_{UGV}}{\|\mathbf{D}_{UGV}\|} \\ &= \frac{\sum_i ((x_i - X), (y_i - Y))}{\sqrt{(\sum_i (x_i - X))^2 + (\sum_i (y_i - Y))^2}} \end{aligned} \tag{16}$$

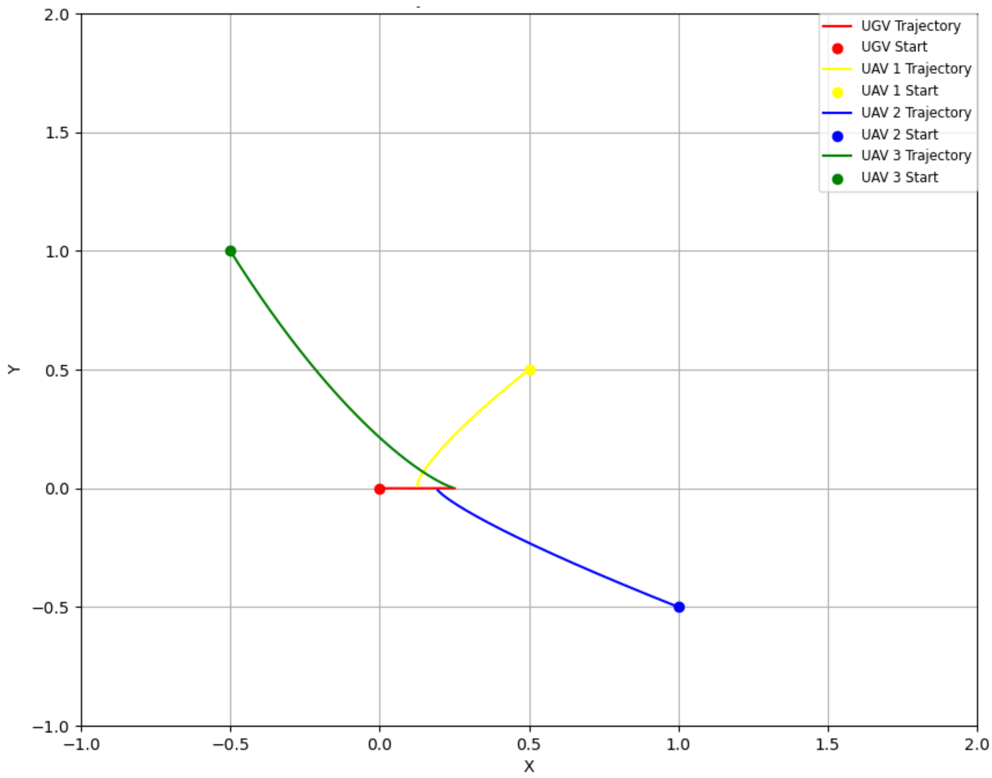

Figure 4. Charging trajectory of UAVs flying towards UGV.

UGV Speed Calculation: Let each UAV_i have a flight speed of v_{UAV_i}, remaining battery level E_{UAV_i}, and initial distance to the UGV denoted by d_{UAV_i}, which can be expressed as:

$$d_{UAV_i} = \sqrt{(x_i - X)^2 + (y_i - Y)^2} \tag{17}$$

The speed component of the UGV in each direction, v_{UGV_i}, is inversely proportional to v_{UAV_i} and E_{UAV_i}, and directly proportional to d_{UAV_i}, which can be expressed as:

$$\begin{aligned} v_{UGV_i} &= k \cdot \frac{d_{UAV_i}}{v_{UAV_i} \cdot E_{UAV_i}} \\ &= k \cdot \frac{\sqrt{(x_i - X)^2 + (y_i - Y)^2}}{v_{UAV_i} \cdot E_{UAV_i}} \end{aligned} \tag{18}$$

Here, k is a proportional constant.

Finally, the UGV's final velocity vector $\mathbf{v}_{UGV}$ is:

$$\mathbf{v}_{UGV} = v_{UGV} \cdot \hat{\mathbf{D}}_{UGV} = \left(\sum i v_{UGV_i}\right) \cdot \hat{\mathbf{D}}_{UGV} \tag{19}$$

The preset charging distance D is:

$$\sqrt{(x_i - X)^2 + (y_i - Y)^2} \ \leq \ D, \tag{20}$$

In comparison, in the AG-FAMT scenario, tasks are generally urgent, so the charging demand for UAVs is relatively lower for time-sensitive tasks. Using the MT-MCMF model for task allocation, each agent receives a task set with a number of tasks up to its maximum executable task limit q_i, and the travel distance required to complete the task set is within its maximum allowable distance S. In this context, agents can effectively manage power consumption while performing urgent tasks through optimized path planning and task allocation, reducing the need for frequent charging.

In summary, by designing a Predictive Charging Trajectory Planning algorithm for UAVs flying to UGVs, we can ensure the continuous and efficient operation of UAVs in AG-MAFT scenarios, enhancing their reliability and efficiency in long-term monitoring tasks. This optimization strategy not only improves task execution stability but also provides valuable insights for future UAV applications.

4. Evaluation

4.1. Dataset and Experiment Setups

4.1.1. Dataset

In this study, we adopt the D4D dataset [61] to construct a realistic simulation environment for task allocation evaluation. The D4D dataset contains large-scale real-world human mobility traces, which have been widely used to model spatial–temporal distributions in MCS and related applications [62].

To simulate task distributions, we sample user trajectory data from the D4D dataset and map the spatial–temporal mobility patterns into sensing task locations. Specifically, the latitude and longitude information in the dataset is converted into a two-dimensional coordinate system, which is then used to generate task points and agent positions. The spatial distribution of tasks is constructed based on the density and movement patterns of mobile users, while base station locations are used to approximate agent deployment positions.

This approach enables us to capture realistic spatial characteristics of task distributions, such as clustering effects and mobility-driven demand variations, which are essential for evaluating task allocation strategies in MCS scenarios. Compared to purely synthetic data, the use of real mobility traces provides a more representative approximation of real-world sensing environments.

However, it should be noted that the D4D dataset does not explicitly model air–ground collaborative systems, and thus cannot fully capture the dynamics of UAV–UGV interactions or aerial mobility patterns. As a result, the constructed simulation environment serves as an indirect approximation rather than a fully realistic representation of air–ground sensing systems.

4.1.2. Experiment Setups

For AG-FAMT, the maximum number of agents required to complete a task is p. Due to the limited resources of agents and the upper limit of their executable tasks, in order to execute as many tasks as possible and avoid excessive repetition of the same task, perception data redundancy may occur. Therefore, considering that the value of p has no impact on the experimental results, we will set it to 6 in the following experiment. In addition, considering the completion status of tasks, the number of tasks that each agent can perform at once is randomly set between 2 and 7. To assess the completion time of an agent with known movement distances, we set the average moving speed of a UGV at 5 m per minute and the average moving speed of a UAV at 20 m per minute. In addition, to evaluate whether our proposed method performs well under different task space distributions, we used three types of task distributions:

- Compact Distribution: Tasks are relatively close together and are mainly concentrated within the region where agents are located.
- Scattered Distribution: Tasks are scattered across a broader target area beyond the region where agents are located.
- Hybrid Distribution: Tasks are randomly distributed across the region where agents are located.

An illustration of the three types of distributions used in our experiments is given in Figure 5.

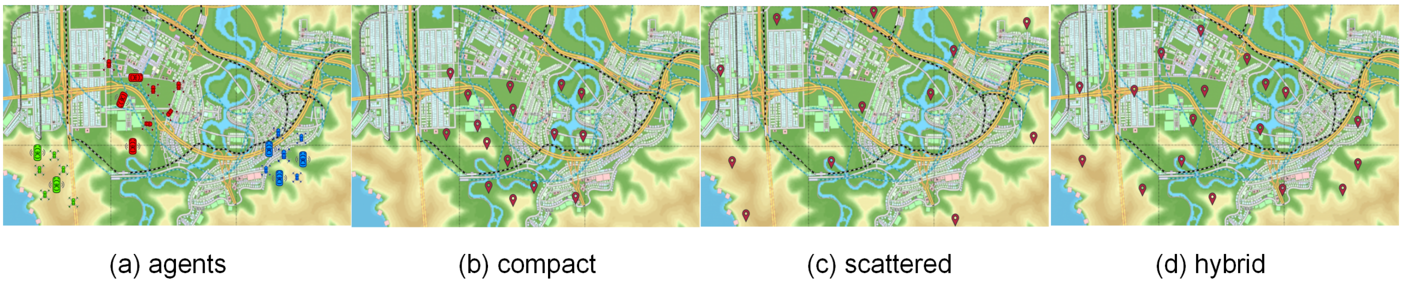

Figure 5. An illustration of the three types of task distribution.

For AG-MAFT, to protect privacy, each agent's precise location is unknown before task execution and is instead represented by the area's location. In our experiment, agents across the entire area are divided into 3 to 5 sections, evenly distributed throughout the city. The number of agents in each area is randomly generated between 5 and 20. Additionally, for the limited-task AG-MAFT scenario, we assume there are 20 randomly distributed tasks throughout the city, each requiring 5 agents to ensure sensing quality. This setup is illustrated in Figure 6.

Figure 6. Distributions of areas and tasks.

4.2. Baseline Methods and Evaluation Metrics

In our evaluation, we selected commonly used baseline methods from previous experiments under similar scenarios. We use MT-GrdPT [29] as a baseline method for comparison with AG-FAMT. Additionally, we introduce a baseline method named W-Grd [29] for AG-MAFT and a static UGV charging algorithm [63,64] as a baseline for comparison with PCTP.

- MT-GrdPT: The goal of the MT-GrdPT algorithm is to minimize travel distance while maximizing the number of completed tasks. It uses a greedy approach where each agent selects tasks based on the shortest distance to a task until the agent completes q tasks. It is important to note that each agent must move from its current position to the assigned task locations.
- W-Grd: The W-Grd algorithm is a greedy algorithm based on the linear weighting method, converting a multi-objective problem into a single-objective problem. Specifically, this method applies the linear weighting approach to transform multi-objective optimization (e.g., minimizing total travel time and total travel distance) into a single-

objective optimization. It assigns weights to each objective function and forms a composite objective function through weighted summation.

- Static-UGV charging: In this model, the UGV remains stationary, and each UGV gradually approaches the UGV at a constant speed until the distance between them reaches a preset charging distance, simulating the charging process upon arrival at the UGV.

4.3. Evaluation Metrics

For AG-FAMT, the total number of tasks completed by the selected agents and the total travel distance are the primary comparison metrics. Additionally, for some urgent tasks in the AG-FAMT problem, the average completion time for q tasks should be as short as possible. Lastly, the algorithm's runtime is also an important factor in solving the problem.

For AG-MAFT, the total travel distance covered by agents during task completion is a key comparison metric, while the cost of time taken to reach the task execution point is another important consideration. Additionally, in the AG-MAFT scenario, the total travel distance of agents during the UAV charging process is also a primary comparison metric.

4.4. Evaluation of AG-FAMT

The optimization objective of AG-FAMT is to minimize the total travel distance while maximizing the total number of completed tasks. It is important to note that, since M agents are required to perform q tasks each, the total number of tasks completed by the agents is $M \times q$. We compare the performance (in terms of total travel distance) of the two algorithms under different total task quantities, different numbers of agents, different task requirements per agent q, and different types of task distributions.

4.4.1. Different Numbers of Tasks

As shown in Figure 7, we present the performance comparison of total travel distance for the two methods under different total task quantities. Additionally, we assume that there are four agents on the MCS platform performing sensing tasks, with each agent required to complete three tasks. As shown in Figure 7, when the four agents execute tasks, the total travel distance remains constant or decreases monotonically as the total number of tasks increases. This is because additional tasks may provide better task options for the agents. Clearly, MT-MCMF outperforms MT-GrdPT across all task quantities.

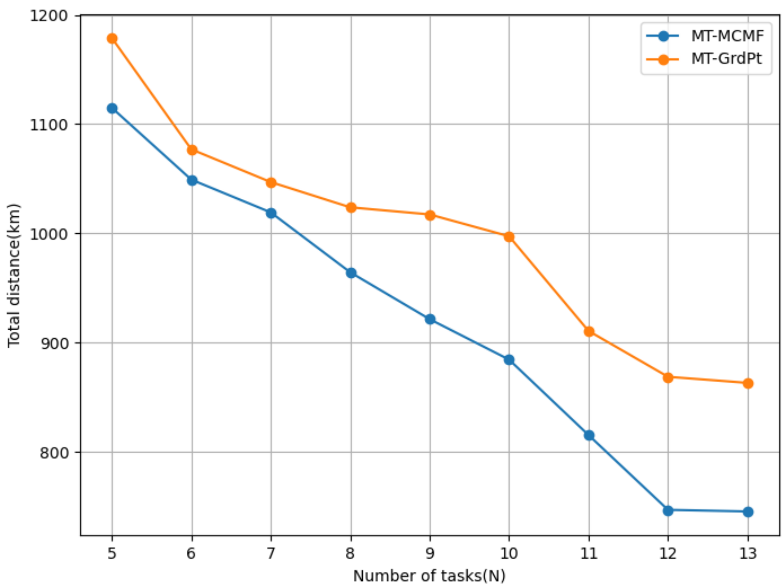

Figure 7. Performance comparison under different number of tasks.

4.4.2. Different Numbers of Agents

As illustrated in Figure 8, we compare the total travel distance of two methods under varying numbers of agents for the same number of tasks. Note that the number of agents M does not exceed the number of tasks N. There are 20 sensing tasks on the MCS platform, with the number of agents ranging from 3 to 11, and each agent is required to complete three tasks. Figure 8a shows that as the number of agents increases, the total number of completed tasks $M \times q$ also increases, leading to an increase in total travel distance. And Figure 8b illustrates that when each agent needs to complete 3 tasks, if the number of selected agents exceeds one-third of the task count, agents can approximately complete all tasks. Clearly, MT-MCMF outperforms MT-GrdPT in terms of travel distance across different numbers of agents.

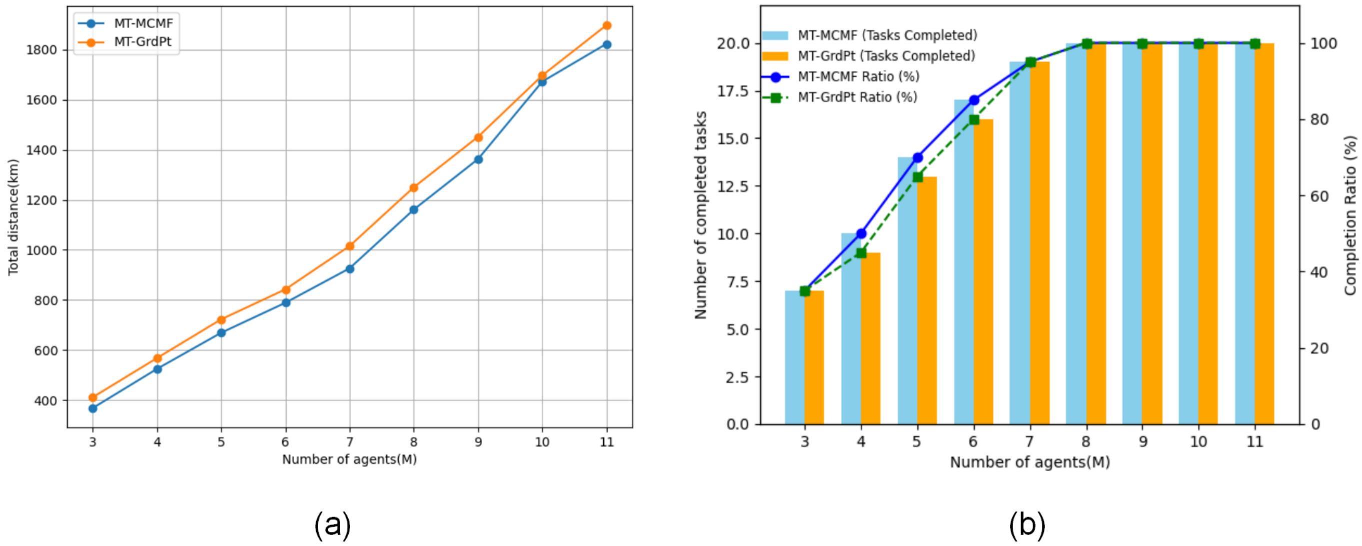

Figure 8. Performance comparison under different number of agents: (**a**) Total distance varies according to number of agents; (**b**) Number of completed tasks varies according to number of agents.

4.4.3. Different Values of q

We assume there are 4 agents available to perform tasks, with each agent required to complete q tasks. We compare the task allocation performance of the two algorithms with different number of tasks for each agent. In Figure 9a, as the number of completed tasks $M \times q$ continuously increases, the total number of possible tasks to be executed, $N \times q$, also increases, resulting in a corresponding increase in total travel distance. It is evident that the MT-GrdPT algorithm performs poorly in terms of travel distance, and this discrepancy becomes more pronounced as q increases. MT-GrdPT is a greedy algorithm that tends to reach a local optimum in a relatively short time, leading to suboptimal results as the number of iterations increases. In addition, Figure 9b illustrated that as the total number of tasks remains constant, the number of completed tasks is proportional to q. When the number of completed tasks $M \times q$ approaches the total number of tasks N, agents are able to complete nearly all tasks.

4.4.4. Different Types

We investigate which type of task distribution is more advantageous for selecting agents to complete tasks: compact, mixed, or dispersed distribution. As shown in Figure 10, we can observe that the dispersed distribution results in the longest total travel distance for agents to complete tasks, while the compact distribution has the shortest distance. This is because tasks in a dispersed distribution are farther apart compared to the other two distributions, requiring agents to travel longer distances and spend more time completing all tasks. Specifically, MT-MCMF outperforms MT-GrdPT across all task distributions.

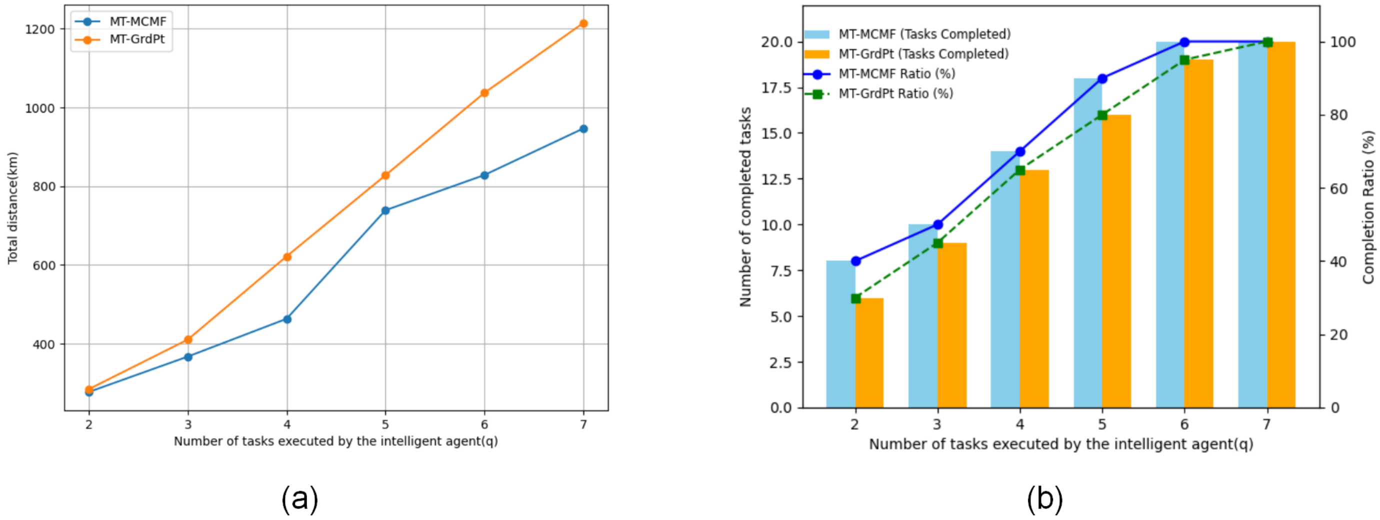

Figure 9. Performance comparison under different number of q: (**a**) Total distance varies according to number of tasks; (**b**) Number of completed tasks varies according to number of tasks executed by the intelligent agent.

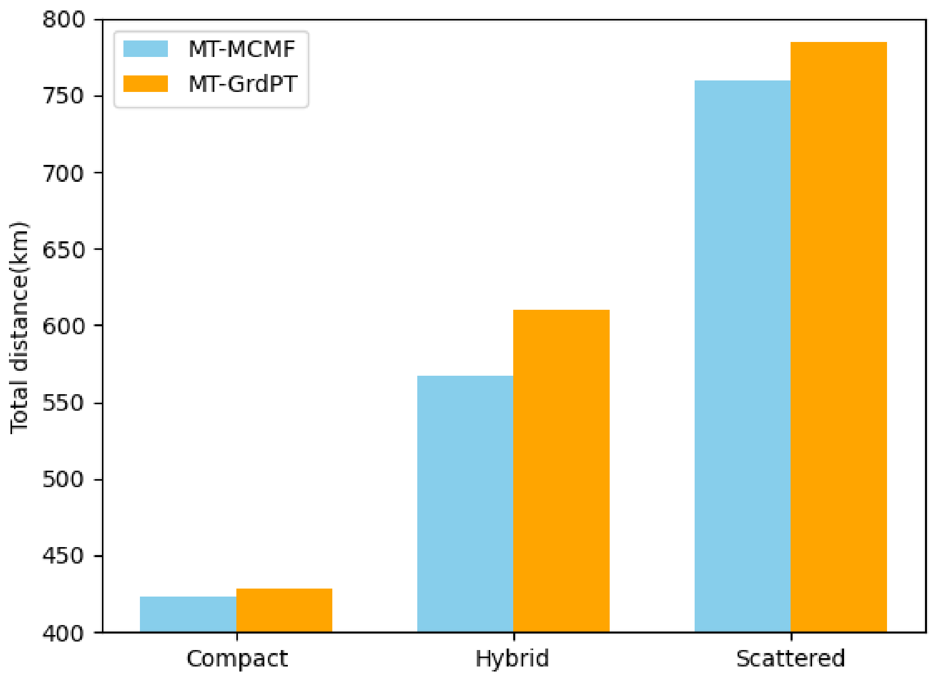

Figure 10. Different types of task distributions.

4.5. Evaluation of AG-MAFT

The optimization objective of AG-MAFT is to minimize the total move time cost and total travel distance while ensuring all tasks are completed. For the bi-objective optimization problem, there are some groups of non-inferior solutions instead of the global optimal solution, known as the Pareto solution. Therefore, we examine the results of the proposed algorithm in terms of both total time cost and total travel distance.

4.5.1. Different Weights

To evaluate the sensitivity of the proposed method to the weighting parameters, we conduct a sensitivity analysis on the weight coefficients K_d and K_t, where $K_d + K_t = 1$.

Figure 11 illustrates the total travel distance and total time cost of the W-ILP and W-Grd algorithms under different weight settings. As shown in Figure 11a, increasing the weight of K_t leads to a gradual reduction in travel time, while increasing K_d results in a decrease in travel distance. This behavior is consistent with the design of the bi-objective formulation, demonstrating that the weighting scheme effectively controls the optimization preference.

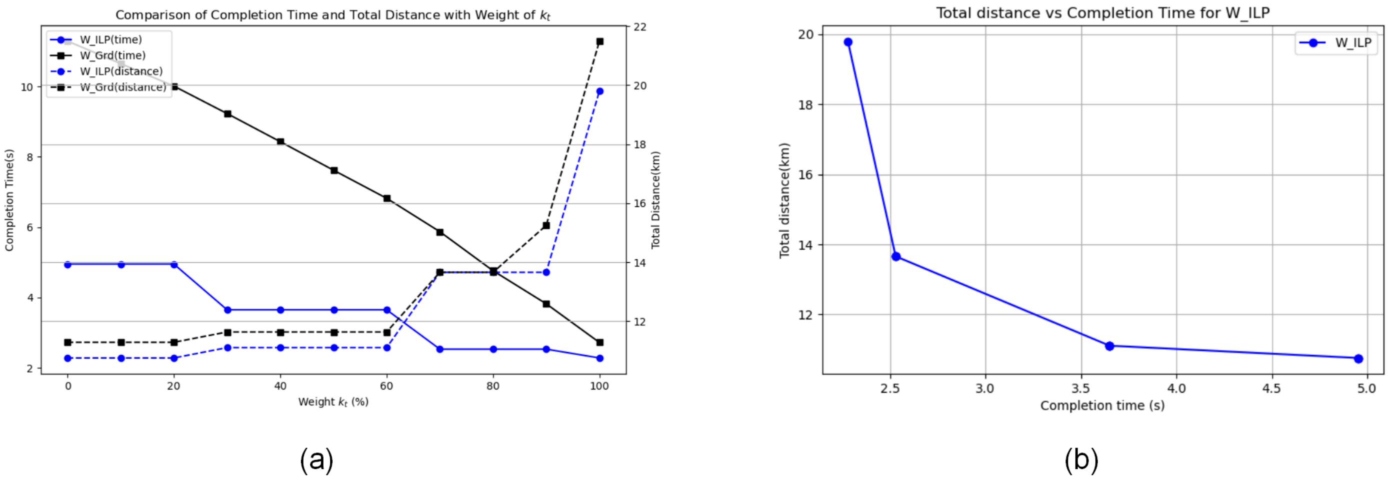

Figure 11. Different weights of AG-MAFT: (**a**) Completion time varies according to the weight; (**b**) Total distance varies according to the completion time.

Moreover, the results indicate that the variation of performance metrics with respect to weight changes is smooth and monotonic, suggesting that the model exhibits stable behavior under different parameter settings. In particular, the W-Grd algorithm achieves performance close to that of W-ILP, indicating that the proposed framework maintains robustness even when approximate solutions are employed.

As shown in Figure 11b, the Pareto frontier of W-ILP provides a set of non-dominated solutions, reflecting the inherent trade-off between travel time and travel distance. No single weight setting can simultaneously minimize both objectives. Among the Pareto solutions, the middle region offers a balanced trade-off, while the extreme regions favor either time or distance optimization.

Overall, the sensitivity analysis demonstrates that although the weighting parameters K_d and K_t influence the optimization preference, the proposed method remains stable and controllable. The W-ILP formulation effectively captures the bi-objective nature of the AG-MAFT problem and provides flexible solution options that can be adjusted according to practical requirements.

4.5.2. Different Charging Path Selections

In the experiment, we divided the setup into four regions and compared charging paths in each region. As illustrated in Figure 12, when a single UGV is responsible for charging different numbers of UAVs, the travel distance of the $PCTP$ algorithm is shorter than that of the Static-UGV charging algorithm. This indicates that $PCTP$ can effectively reduce the total travel distance of agents when multiple UAVs require charging. This result confirms the optimization effectiveness of $PCTP$ in path planning, demonstrating that it can more efficiently respond to the dynamic demands of UAVs through predictive trajectory planning. This leads to effective control of agent travel distance, helping to improve overall system efficiency.

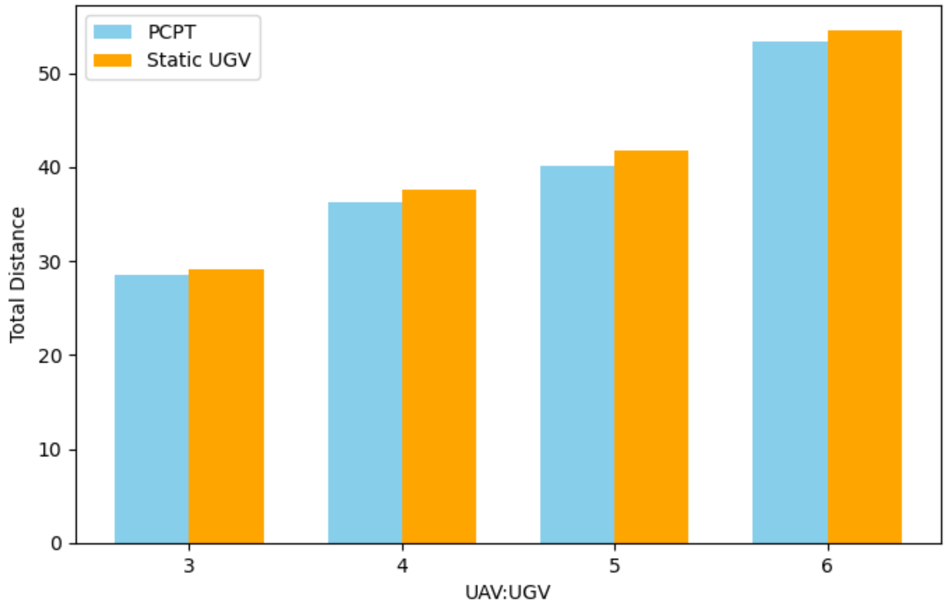

Figure 12. Different total distance of charging.

5. Discussion

5.1. Superiority of Air-Ground Collaborative MCS

This paper proposes a multi-agent collaborative task allocation framework to address the AG-FAMT and AG-MAFT problems. For the AG-FAMT scenario, we introduce a Minimum-Cost Maximum-Flow (MCMF) model to ensure that each homogeneous agent completes an equal number of sensing tasks, thereby maximizing the total number of completed tasks while minimizing movement distances. In the AG-MAFT scenario, we flexibly select suitable agents based on the task demands across different regions and agents' capabilities, optimizing task allocation outcomes. The framework AGCo-MATA, grounded in quantitative considerations, demonstrates high flexibility across various constraints, offering a novel direction for research in multi-agent collaborative task allocation.

5.2. Configuration Optimization

The experimental results demonstrate that the proposed method effectively enhances task completion efficiency in both AG-FAMT and AG-MAFT scenarios. In the AG-FAMT scenario, optimizing the configuration of agent numbers and the number of tasks assigned to each agent maximizes task completion rates while minimizing movement distances. For the AG-MAFT scenario, selecting agents based on time cost and path distance significantly improves task completion rates and system robustness. Additionally, in AG-MAFT, an increase in the weight of K_t leads to a gradual decrease in travel time, while a higher weight for K_d increases travel distance. This suggests that assigning farther tasks to faster-moving UAVs and closer tasks to UGVs can further optimize task allocation. These findings indicate that the proposed allocation strategy has broad applicability across diverse application scenarios.

5.3. Limitations

Although the proposed framework provides valuable insights into heterogeneous task allocation in air–ground collaborative systems, several limitations remain. Firstly, for analytical tractability, we assume homogeneous agents and uniform task difficulty.

While this simplifies the analysis, real-world heterogeneity in agent capabilities and task requirements may affect solution optimality and robustness. Secondly, communication costs and coordination overhead are not explicitly modeled. In practical deployments, communication delays and bandwidth constraints may influence task allocation efficiency. Thirdly, the formulation assumes a relatively static environment and does not account for dynamic task arrivals or environmental uncertainties. In such scenarios, online re-planning or adaptive scheduling would be required. Fourthly, the MT-MCMF formulation incurs combinatorial complexity due to task-set enumeration. Although practical for small task-set sizes and moderate system scales, scalability may be limited in large-scale settings.

Finally, the evaluation is based on task distributions constructed from the D4D dataset, which provides real-world mobility traces. While this enables realistic spatial–temporal approximation, it does not fully capture UAV–UGV dynamics or heterogeneous sensing behaviors. Future work will incorporate real air–ground datasets or field experiments for further validation. Despite these limitations, the proposed framework offers a unified perspective on task allocation and energy-aware coordination, and can be extended to more realistic and dynamic scenarios.

5.4. Future Work

To further enhance task allocation efficiency in the AG-FAMT problem, future research could explore incorporating task prioritization, inter-agent dynamic collaboration mechanisms, and failed task redistribution to refine allocation strategies. In addition, considering heterogeneity in tasks and agents (such as priority of tasks and varying agent capabilities) would help build more precise allocation models. In the AG-MAFT scenario, methods to minimize redundant collaborations among agents are also worth exploring to reduce system resource waste.

6. Conclusions

In this paper, we proposed two bi-objective optimization problems for agent selection: the AG-FAMT problem, where agent resources are insufficient, and the AG-MAFT problem, where agent resources are abundant. We use the MCMF model to address the AG-FAMT problem and propose the MT-MCMF algorithm to select agents with the maximum number of completed tasks and the minimum total travel distance. Additionally, we employ a dual-objective optimization model to solve the AG-MAFT problem and introduce the W-ILP algorithm to select agents based on travel time and minimum total travel distance. Furthermore, we propose the PCTP algorithm for UAVs charging path planning to optimize the travel distance for agent charging.

Author Contributions: Conceptualization, K.Z. and Z.M.; Software, L.Y., T.S. and B.F.; Validation, T.S. and B.F.; Formal analysis, B.F. and J.D.; Data curation, T.S.; Writing—original draft, L.Y.; Writing—review & editing, K.Z. and Z.M.; Visualization, B.F.; Supervision, K.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by China Postdoctoral Science Foundation: 2024M754226

Data Availability Statement: The raw data supporting the findings of this study are not publicly available due to relevant privacy protection regulations and ethical research constraints. Restrictions apply to the dissemination of the involved research data, and therefore the relevant original data cannot be shared or provided.

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

References

1. Zhang, S.; He, N.; Li, Z.; Fu, Q.; Liang, W.; Xiong, N.N. A Survey on Mobile Crowd Sensing: Concepts, Applications, Technologies, and Future Directions. *IEEE Internet Things J.* **2025**, *12*, 52090–52109. [CrossRef]
2. Newton, D.A.; Dell, A.G. Assistive technology. *J. Spec. Educ. Technol.* **2011**, *26*, 55–60. [CrossRef]
3. Wang, Y.; Tang, Q.; Wei, W.; Yang, C.; Yang, D.; Wang, C.; Xu, L.; Chen, L. CrowdRadar: A mobile crowdsensing framework for urban traffic green travel safety risk assessment. *Front. Big Data* **2025**, *8*, 1440816. [CrossRef]
4. Purba, R.A.; Bedogni, L. Assessing Benefits and Limitations of Multiple Data Sources for Environmental Monitoring. In *Proceedings of the 2025 IEEE 22nd Consumer Communications & Networking Conference (CCNC)*; IEEE: New York, NY, USA, 2025; pp. 1–6.
5. Zhou, N.; Liu, Y.; Wang, H.; Man, F.; Xu, J.; Dang, F.; Hong, C.; Liu, Y.; Zhang, X.P.; Song, Y.; et al. CatUA: Catalyzing urban air quality intelligence through mobile crowd-sensing. *IEEE Trans. Mob. Comput.* **2025**, *24*, 9184–9201. [CrossRef]
6. Mednis, A.; Strazdins, G.; Zviedris, R.; Kanonirs, G.; Selavo, L. Real time pothole detection using android smartphones with accelerometers. In *Proceedings of the 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS)*; IEEE: New York, NY, USA, 2011; pp. 1–6.
7. Ray, A.; Chowdhury, C.; Bhattacharya, S.; Roy, S. A survey of mobile crowdsensing and crowdsourcing strategies for smart mobile device users. *CCF Trans. Pervasive Comput. Interact.* **2023**, *5*, 98–123. [CrossRef]
8. Abdeddine, A.; Iraqi, Y.; Mekouar, L. Mobile crowdsensing model: A survey. *J. Syst. Archit.* **2025**, *162*, 103384. [CrossRef]
9. Sumaidaa, S.; AlMenhali, H.; Alazzani, M.; Han, K. Enhancing security of mobile crowd sensing in unmanned aerial vehicle ecosystems. *Front. Commun. Netw.* **2025**, *6*, 1443592. [CrossRef]
10. Chen, S.; Wei, K.; Pei, T.; Long, S. AoI-guaranteed UAV crowdsensing: A UGV-assisted deep reinforcement learning approach. *Ad Hoc Netw.* **2025**, *173*, 103805. [CrossRef]
11. Wang, S.; Jiang, F.; Zhang, B.; Ma, R.; Hao, Q. Development of UAV-based target tracking and recognition systems. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 3409–3422. [CrossRef]
12. Xu, J.; Zhong, F.; Wang, Y. Learning multi-agent coordination for enhancing target coverage in directional sensor networks. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 10053–10064.
13. Gao, H.; Feng, J.; Xiao, Y.; Zhang, B.; Wang, W. A UAV-assisted multi-task allocation method for mobile crowd sensing. *IEEE Trans. Mob. Comput.* **2022**, *22*, 3790–3804. [CrossRef]
14. He, H.; Peng, J.; Cai, L.; Liu, W.; Huang, Z. Air-Ground Collaborative Mobile Crowdsensing by Predictive Multi-Agent Deep Reinforcement Learning. In *Proceedings of the 2025 IEEE 102nd Vehicular Technology Conference (VTC2025-Fall)*; IEEE: New York, NY, USA, 2025; pp. 1–6.
15. Xi, M.; Dai, H.; He, J.; Li, W.; Wen, J.; Xiao, S.; Yang, J. A lightweight reinforcement learning-based real-time path planning method for unmanned aerial vehicles. *IEEE Internet Things J.* **2024**, *11*, 21061–21071. [CrossRef]
16. Yu, H.; Meier, K.; Argyle, M.; Beard, R.W. Cooperative path planning for target tracking in urban environments using unmanned air and ground vehicles. *IEEE/ASME Trans. Mechatron.* **2014**, *20*, 541–552. [CrossRef]
17. Peng, J.; Chen, W.; Fan, Y.; He, H.; Wei, Z.; Ma, C. Ecological Driving Framework of Hybrid Electric Vehicle Based on Heterogeneous Multi-Agent Deep Reinforcement Learning. *IEEE Trans. Transp. Electrif.* **2023**, *10*, 392–406. [CrossRef]
18. Zeng, Y.; Zhang, R. Energy-efficient UAV communication with trajectory optimization. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 3747–3760. [CrossRef]
19. Munasinghe, I.; Perera, A.; Deo, R.C. A comprehensive review of uav-ugv collaboration: Advancements and challenges. *J. Sens. Actuator Netw.* **2024**, *13*, 81. [CrossRef]
20. Wang, Y.; Wu, J.; Hua, X.; Liu, C.H.; Li, G.; Zhao, J.; Yuan, Y.; Wang, G. Air-ground spatial crowdsourcing with uav carriers by geometric graph convolutional multi-agent deep reinforcement learning. In *Proceedings of the 2023 IEEE 39th International Conference on Data Engineering (ICDE)*; IEEE: New York, NY, USA, 2023; pp. 1790–1802.
21. Ding, Y.; Xin, B.; Chen, J. A review of recent advances in coordination between unmanned aerial and ground vehicles. *Unmanned Syst.* **2021**, *9*, 97–117. [CrossRef]
22. Sun, M.; Xu, X.; Qin, X.; Zhang, P. AoI-energy-aware UAV-assisted data collection for IoT networks: A deep reinforcement learning method. *IEEE Internet Things J.* **2021**, *8*, 17275–17289. [CrossRef]
23. Wei, K.; Huang, K.; Wu, Y.; Li, Z.; He, H.; Zhang, J.; Chen, J.; Guo, S. High-performance UAV crowdsensing: A deep reinforcement learning approach. *IEEE Internet Things J.* **2022**, *9*, 18487–18499. [CrossRef]
24. Dai, Z.; Liu, C.H.; Ye, Y.; Han, R.; Yuan, Y.; Wang, G.; Tang, J. AoI-minimal UAV crowdsensing by model-based graph convolutional reinforcement learning. In *Proceedings of the IEEE INFOCOm 2022-IEEE Conference on Computer Communications*; IEEE: New York, NY, USA, 2022; pp. 1029–1038.
25. Ye, Y.; Wang, H.; Liu, C.H.; Dai, Z.; Li, G.; Wang, G.; Tang, J. QoI-Aware Mobile Crowdsensing for Metaverse by Multi-Agent Deep Reinforcement Learning. *IEEE J. Sel. Areas Commun.* **2023**, *42*, 783–798. [CrossRef]

26. Baek, J.; Kaddoum, G. Heterogeneous task offloading and resource allocations via deep recurrent reinforcement learning in partial observable multifog networks. *IEEE Internet Things J.* **2020**, *8*, 1041–1056. [CrossRef]
27. Chen, P.; Luo, L.; Guo, D.; Luo, X.; Li, X.; Sun, Y. Secure Task Offloading for Rural Area Surveillance Based on UAV-UGV Collaborations. *IEEE Trans. Veh. Technol.* **2023**, *73*, 923–937. [CrossRef]
28. Abdeddine, A.; Iraqi, Y.; Mekouar, L. An efficient task allocation in mobile crowdsensing environments. *IEEE Trans. Netw. Serv. Manag.* **2025**, *22*, 2323–2342. [CrossRef]
29. Liu, Y.; Guo, B.; Wang, Y.; Wu, W.; Yu, Z.; Zhang, D. TaskMe: Multi-task allocation in mobile crowd sensing. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*; Association for Computing Machinery: New York, NY, USA, 2016; pp. 403–414.
30. Liu, C.H.; Zhang, B.; Su, X.; Ma, J.; Wang, W.; Leung, K.K. Energy-aware participant selection for smartphone-enabled mobile crowd sensing. *IEEE Syst. J.* **2015**, *11*, 1435–1446. [CrossRef]
31. Zhang, B.; Liu, C.H.; Tang, J.; Xu, Z.; Ma, J.; Wang, W. Learning-based energy-efficient data collection by unmanned vehicles in smart cities. *IEEE Trans. Ind. Inform.* **2017**, *14*, 1666–1676. [CrossRef]
32. Mohan, P.; Padmanabhan, V.N.; Ramjee, R. Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*; Association for Computing Machinery: New York, NY, USA, 2008; pp. 323–336.
33. Pawar, K.; Jagtap, S.; Bhoir, S. Efficient pothole detection using smartphone sensors. In *Proceedings of the ITM Web of Conferences*; EDP Sciences: Paris, France, 2020; Volume 32, p. 03013.
34. Yin, H.; Yu, Z.; Wang, L.; Wang, J.; Han, L.; Guo, B. ISIATasker: Task Allocation for Instant-Sensing and Instant-Actuation Mobile Crowdsensing. *IEEE Internet Things J.* **2022**, *9*, 3158–3173. [CrossRef]
35. Xu, C.; Song, W. Intelligent Task Allocation for Mobile Crowdsensing with Graph Attention Network and Deep Reinforcement Learning. *IEEE Trans. Netw. Sci. Eng.* **2023**, *10*, 1032–1048. [CrossRef]
36. Li, T.; Jung, T.; Qiu, Z.; Li, H.; Cao, L.; Wang, Y. Scalable Privacy-Preserving Participant Selection for Mobile Crowdsensing Systems: Participant Grouping and Secure Group Bidding. *IEEE Trans. Netw. Sci. Eng.* **2020**, *7*, 855–868. [CrossRef]
37. Zhao, B.; Liu, X.; Chen, W.N.; Deng, R.H. CrowdFL: Privacy-preserving mobile crowdsensing system via federated learning. *IEEE Trans. Mob. Comput.* **2022**, *22*, 4607–4619. [CrossRef]
38. Liu, Y.; Yu, Z.; Cui, H.; Helal, S.; Guo, B. SafeCity: A heterogeneous mobile crowd sensing system for urban public safety. *IEEE Internet Things J.* **2023**, *10*, 18330–18345. [CrossRef]
39. Ye, Y.; Tian, Y.; Liu, C.H.; Dong, L.; Qi, G.; Wu, D. AoI-Aware Air-Ground Mobile Crowdsensing by Multi-Agent Curriculum Learning With Collaborative Observation Augmentation. *IEEE Trans. Mob. Comput.* **2025**, *24*, 11675–11687. [CrossRef]
40. Shao, T.; Zhao, K.; Liu, F.; Yang, L.; Guo, B. AG-MPBS: A Mobility-Aware Prediction and Behavior-Based Scheduling Framework for Air-Ground Unmanned Systems. *arXiv* **2025**, arXiv:2512.16454.
41. Zhao, K.; Zhou, Y.; Xue, H.; Ding, L.; He, L.; Guo, B. AG-HCRL: Air-Ground Collaborative Crowdsensing Based on Deep Reinforcement Learning in City Sensing. In *Proceedings of the 2024 IEEE Smart World Congress (SWC)*; IEEE: New York, NY, USA, 2024; pp. 1074–1082.
42. Wang, Y.; Chen, W.; Luan, T.H.; Su, Z.; Xu, Q.; Li, R.; Chen, N. Task offloading for post-disaster rescue in unmanned aerial vehicles networks. *IEEE/ACM Trans. Netw.* **2022**, *30*, 1525–1539. [CrossRef]
43. He, J.; Zhou, Y.; Huang, L.; Kong, Y.; Cheng, H. Ground and aerial collaborative mapping in urban environments. *IEEE Robot. Autom. Lett.* **2020**, *6*, 95–102. [CrossRef]
44. Wang, L.; Yu, Z.; Wu, K.; Yang, D.; Wang, E.; Wang, T.; Mei, Y.; Guo, B. Towards robust task assignment in mobile crowdsensing systems. *IEEE Trans. Mob. Comput.* **2022**, *22*, 4297–4313. [CrossRef]
45. Xu, C.; Song, W. Decentralized task assignment for mobile crowdsensing with multi-agent deep reinforcement learning. *IEEE Internet Things J.* **2023**, *10*, 16564–16578. [CrossRef]
46. Yu, Q.; Shen, Z.; Pang, Y.; Liu, R. Proficiency constrained multi-agent reinforcement learning for environment-adaptive multi uav-ugv teaming. In *Proceedings of the 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*; IEEE: New York, NY, USA, 2021; pp. 2114–2118.
47. Gao, Y.; Chen, J.; Chen, X.; Wang, C.; Hu, J.; Deng, F.; Lam, T.L. Asymmetric self-play-enabled intelligent heterogeneous multirobot catching system using deep multiagent reinforcement learning. *IEEE Trans. Robot.* **2023**, *39*, 2603–2622. [CrossRef]
48. Zhang, C.; Li, Q.; Zhu, Y.; Zhang, J. Dynamics of task allocation based on game theory in multi-agent systems. *IEEE Trans. Circuits Syst. II Express Briefs* **2018**, *66*, 1068–1072. [CrossRef]
49. Cui, R.; Guo, J.; Gao, B. Game theory-based negotiation for multiple robots task allocation. *Robotica* **2013**, *31*, 923–934. [CrossRef]
50. Wang, Y.; Liu, C.H.; Piao, C.; Yuan, Y.; Han, R.; Wang, G.; Tang, J. Human-drone collaborative spatial crowdsourcing by memory-augmented and distributed multi-agent deep reinforcement learning. In *Proceedings of the 2022 IEEE 38th International Conference on Data Engineering (ICDE)*; IEEE: New York, NY, USA, 2022; pp. 459–471.

51. Ding, L.; Zhao, D.; Cao, M.; Ma, H. When crowdsourcing meets unmanned vehicles: Toward cost-effective collaborative urban sensing via deep reinforcement learning. *IEEE Internet Things J.* **2021**, *8*, 12150–12162. [CrossRef]
52. Guo, B.; Liu, Y.; Wang, L.; Li, V.O.; Lam, J.C.; Yu, Z. Task allocation in spatial crowdsourcing: Current state and future directions. *IEEE Internet Things J.* **2018**, *5*, 1749–1764. [CrossRef]
53. Ra, M.R.; Liu, B.; La Porta, T.F.; Govindan, R. Medusa: A programming framework for crowd-sensing applications. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Servicesro*; Association for Computing Machinery: New York, NY, USA, 2012; pp. 337–350.
54. Kazemi, L.; Shahabi, C. Geocrowd: Enabling query answering with spatial crowdsourcing. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*; Association for Computing Machinery: New York, NY, USA, 2012; pp. 189–198.
55. Song, Z.; Liu, C.H.; Wu, J.; Ma, J.; Wang, W. QoI-aware multitask-oriented dynamic participant selection with budget constraints. *IEEE Trans. Veh. Technol.* **2014**, *63*, 4618–4632. [CrossRef]
56. Guo, B.; Liu, Y.; Wu, W.; Yu, Z.; Han, Q. ActiveCrowd: A framework for optimized multitask allocation in mobile crowdsensing systems. *IEEE Trans. Hum.-Mach. Syst.* **2016**, *47*, 392–403. [CrossRef]
57. Chen, Z.; Wu, J.; Deng, Z.; Huang, H.Z. Learning Category-Invariant Disentangled Features for Domain Generalization in Machine Fault Diagnosis. *IEEE/ASME Trans. Mechatronics* **2026**. [CrossRef]
58. Muandet, K.; Balduzzi, D.; Schölkopf, B. Domain generalization via invariant feature representation. In *Proceedings of the International Conference on Machine Learning*; PMLR: New York, NY, USA, 2013; pp. 10–18.
59. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
60. Hadji, M.; Zeghlache, D. Minimum cost maximum flow algorithm for dynamic resource allocation in clouds. In *Proceedings of the 2012 IEEE Fifth International Conference on Cloud Computing*; IEEE: New York, NY, USA, 2012; pp. 876–882.
61. Blondel, V.D.; Esch, M.; Chan, C.; Clérot, F.; Deville, P.; Huens, E.; Morlot, F.; Smoreda, Z.; Ziemlicki, C. Data for development: The d4d challenge on mobile phone data. *arXiv* **2012**, arXiv:1210.0137.
62. Zhou, Z.; Liu, P.; Feng, J.; Zhang, Y.; Mumtaz, S.; Rodriguez, J. Computation resource allocation and task assignment optimization in vehicular fog computing: A contract-matching approach. *IEEE Trans. Veh. Technol.* **2019**, *68*, 3113–3125. [CrossRef]
63. Choi, C.H.; Jang, H.J.; Lim, S.G.; Lim, H.C.; Cho, S.H.; Gaponov, I. Automatic wireless drone charging station creating essential environment for continuous drone operation. In *Proceedings of the 2016 International Conference on Control, Automation and Information Sciences (ICCAIS)*; IEEE: New York, NY, USA, 2016; pp. 132–136. [CrossRef]
64. Leonard, J.; Aldhaher, S.; Savvaris, A.; Tsourdos, A. Automated Recharging Station for Swarm of Unmanned Aerial Vehicles. In *ASME International Mechanical Engineering Congress and Exposition*; ASME: New York, NY, USA, 2012.