

Article

A Two-Stage Optimization Framework for UAV Fleet Sizing and Task Allocation in Emergency Logistics Using the GWO and CBBA

Yongchao Zhang ¹, Wei Xu ¹, Helin Ye ¹ and Zhuoyong Shi ^{2,*} ¹ School of Electrical and Information Engineering, Xi'an Jiaotong University City College, Xi'an 710018, China² Faculty of Science, National University of Singapore, 21 Lower Kent Ridge Road, Singapore 119077, Singapore

* Correspondence: shizhuoyong@u.nus.edu

Abstract

The joint optimization of fleet size and task allocation presents a critical challenge in deploying Unmanned Aerial Vehicles (UAVs) for time-sensitive missions such as emergency logistics. Conventional approaches often rely on pre-determined fleet sizes or computationally intensive centralized optimizers, which can lead to suboptimal performance. To address this gap, this paper proposes a novel two-stage hierarchical framework that integrates the Grey Wolf Optimizer (GWO) with the Consensus-Based Bundle Algorithm (CBBA). At the strategic level, the GWO determines the optimal number of UAVs by minimizing a comprehensive cost function that balances mission efficiency and operational costs. Subsequently, at the tactical level, the CBBA performs decentralized, real-time task allocation for the optimally sized fleet. We validated our GWO-CBBA framework through extensive simulations against three benchmarks: a standard CBBA with a fixed fleet, a centralized Particle Swarm Optimization (PSO) approach, and a Greedy Heuristic algorithm. The results are compelling: our framework demonstrates superior performance across all key metrics, reducing the overall scheduling cost by 13.2–36.5%, minimizing UAV mileage cost and significantly decreasing total task waiting time. This work provides a robust and efficient solution that effectively balances operational costs with service quality for dynamic multi-UAV scheduling problems.



Academic Editors: Xiangkun He,
Hailong Huang, Caizheng Wang and
Henglai Wei

Received: 6 June 2025

Revised: 8 July 2025

Accepted: 14 July 2025

Published: 16 July 2025

Citation: Zhang, Y.; Xu, W.; Ye, H.; Shi, Z. A Two-Stage Optimization Framework for UAV Fleet Sizing and Task Allocation in Emergency Logistics Using the GWO and CBBA. *Drones* **2025**, *9*, 501. <https://doi.org/10.3390/drones9070501>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: unmanned aerial vehicles; consensus bundle algorithm; integrated UAV scheduling; task allocation

1. Introduction

1.1. Context and Motivation

With the continuous advancement of science and technology, drones are increasingly being used in military [1–3] and civilian fields [4–6], especially in the field of material distribution, where they have great potential for research [7–11]. While beneficial for routine deliveries, their true potential is realized in critical scenarios such as post-disaster relief and emergency material delivery. In the aftermath of events like earthquakes or floods, damaged infrastructure often renders ground-based transportation slow, unreliable, or entirely unfeasible [12]. In these time-sensitive situations, drone-based delivery, characterized by its high efficiency, flexibility, and ability to bypass ground obstacles, emerges as a vital and often indispensable solution [13]. This aligns with the broader application of intelligent algorithms in critical emergency response scenarios, such as for smart firefighting and escape route planning [14].

However, the effective deployment of a UAV fleet for emergency logistics presents a complex, coupled optimization challenge that extends beyond simple path planning [15–19]. Two fundamental and interconnected questions must be addressed to ensure mission success:

Strategic Fleet Sizing [20]: What is the optimal number of UAVs to dispatch for a given mission? Deploying too few UAVs may result in failure to meet urgent delivery deadlines. Conversely, deploying too many increases operational costs, airspace congestion, and coordination complexity, which can lead to diminishing returns or even decreased overall efficiency.

Tactical Task Allocation [21]: Once the fleet size is determined, how should the specific delivery tasks be efficiently and robustly assigned to individual UAVs to minimize completion time and resource consumption?

Addressing these questions independently often leads to suboptimal solutions. A fleet size chosen without considering the intricacies of task allocation may be inadequate, while a sophisticated task allocation algorithm is ineffective if the fleet is improperly sized. This paper addresses this coupled problem by proposing an integrated framework designed specifically for emergency logistics scenarios. Our goal is to first determine the optimal fleet size by holistically balancing cost, mission time, and reliability and then perform a robust, distributed task allocation for that optimally sized fleet to maximize overall mission effectiveness.

1.2. State of the Art and Research Gap

UAV task allocation is a well-studied problem, with solutions broadly categorized into centralized and distributed approaches.

1.2.1. Centralized and Distributed Task Allocation Methods

Centralized approaches rely on a single ground station that possesses global information to compute the optimal allocation for the entire fleet. Metaheuristic algorithms like Ant Colony Optimization (ACO) [22] and Particle Swarm Optimization (PSO) [23] are popular for solving these complex, NP-hard problems. While capable of finding high-quality, globally near-optimal solutions, centralized methods suffer from significant drawbacks. They are computationally expensive, struggle to scale with the size of the fleet and the number of tasks, and represent a single point of failure, limiting their robustness in dynamic emergency scenarios [24].

In contrast, distributed approaches empower individual UAVs to make allocation decisions based on local information and communication with neighbors. Market-based mechanisms, where UAVs “bid” for tasks, are a prominent example. The Consensus-Based Bundle Algorithm (CBBA) [25] is a state-of-the-art distributed method known for its robustness and scalability. In CBBA, agents iteratively build bundles of tasks and resolve conflicts through a consensus mechanism, ensuring a conflict-free solution without a central coordinator. This makes it highly suitable for dynamic and uncertain environments.

1.2.2. The Disconnect Between Fleet Sizing and Task Allocation

A critical review of the literature reveals a significant research gap: a disconnect exists between strategic fleet-sizing and tactical task allocation.

Most distributed algorithms, including the standard CBBA, are highly effective at answering the question, “Given a team of N UAVs, which UAV should do what?” However, they presuppose a fixed, pre-determined fleet size (N). They do not address the strategic, higher-level question of “How many UAVs should be dispatched to begin with?”

On the other hand, centralized metaheuristics that can optimize fleet size as part of their search space often oversimplify the complexities of distributed execution and lack the scalability and robustness required for real-world deployment. An integrated framework

that synergistically combines a high-level, global optimizer for fleet sizing with a robust, distributed algorithm for task allocation is needed to bridge this gap.

1.3. Our Contribution and Paper Organization

This paper confronts the coupled challenge of strategic fleet sizing and tactical task allocation by proposing a novel, hierarchical two-stage framework. Our approach synergistically combines a global metaheuristic optimizer with a distributed consensus algorithm, strategically decoupling the problem into two distinct layers: a global optimization layer for determining the optimal fleet size and a local, distributed consensus layer for subsequent task assignment. This methodology is designed to overcome the limitations of monolithic centralized planners and fixed-fleet distributed methods.

The primary contributions of this work are three-fold, establishing a significant advancement over existing approaches:

1. **A Novel Two-Stage Framework:** We propose an integrated GWO-CBBA framework that systematically solves the coupled problem of fleet sizing and task allocation. Such hierarchical models, which decompose complex problems into strategic and tactical layers, have proven effective for sophisticated evaluation and decision-making tasks [26]. GWO is used to determine the optimal number of UAVs (N^*) by optimizing a comprehensive objective function, while CBBA performs distributed task assignment for the resulting N^* agents. This hierarchical approach is designed to achieve near-optimal solutions with greater computational efficiency than purely centralized methods.

2. **A Comprehensive System-Level Cost Model:** For the strategic stage, we formulate a holistic objective function that captures the critical, real-world trade-offs between operational costs (e.g., UAV deployment and mileage costs) and service quality (e.g., total task waiting time). This ensures that the fleet size (N^*) is optimized for overall mission effectiveness, rather than being determined by a single, isolated metric.

3. **Rigorous Performance Validation and Benchmarking:** We conduct an extensive comparative analysis, benchmarking our framework against three distinct paradigms: a standard fixed-fleet distributed algorithm (CBBA), a powerful centralized optimizer (PSO), and a baseline Greedy Heuristic algorithm. The empirical results provide compelling evidence of our framework's superiority across all key performance indicators, demonstrating substantial reductions in overall scheduling cost, required UAV mileage, and task waiting times.

The remainder of this paper is structured as follows: Section 2 formulates the problem domain and defines the constituent objective functions. Section 3 details the mathematical modeling of the UAV scheduling problem. Section 4 presents the proposed two-stage GWO-CBBA framework in comprehensive detail. Section 5 describes the simulation setup, presents the results, and provides the in-depth comparative analysis. Finally, Section 6 concludes the paper, summarizing the key findings and discussing limitations and avenues for future research.

2. Problem Analysis

2.1. Constraint Establishment

The application of Unmanned Aerial Vehicles (UAVs) in logistics distribution is expanding, with task allocation emerging as a key challenge. This is a complex multi-objective optimization problem that aims to optimally assign a set of distribution tasks to a limited UAV fleet. The goal is to maximize overall system efficiency while adhering to resource constraints (e.g., endurance, payload) and task requirements (e.g., timeliness). The allocation process must comprehensively consider multiple factors, including UAV performance pa-

rameters, geographical locations of delivery points, and task priorities. This paper focuses specifically on the UAV-related factors detailed in Table 1.

Table 1. Table of Considerations for an Integrated UAV Scheduling Model.

Consideration	Specific Factors
UAV resources	Performance parameters such as the number of drones, endurance, payload, etc.
Mission requirements	Location, priority, start and end time, and resource requirements of tasks
Restrictive condition	Includes drone range limitations, communication range limitations, and mission time windows
Optimization goals	Minimize task completion time, minimize total energy consumption, and maximize task completion rate

2.2. Single UAV Objective Function

The objective function is a key part of a mathematical model used for optimization and decision making, which defines the objective that needs to be minimized or maximized. In UAV material distribution, the objective function usually involves the following aspects:

(1) Minimize task completion time

Suppose there are N tasks $\{T_1, T_2, \dots, T_N\}$ each task T_i with a completion time f_i , and the goal is to minimize the sum of the completion times of all of the tasks:

$$\min \sum_{i=1}^N f_i \quad (1)$$

(2) Minimize energy consumption

Assuming that the energy consumption of the UAV U_j while performing a mission T_i can be expressed as E_{ij} , the minimized total energy consumption can be expressed as:

$$\min \sum_{i=1}^N \sum_{j=1}^M x_{ij} \cdot E_{ij} \quad (2)$$

where x_{ij} is a binary decision variable; $x_{ij} = 1$ if task i is assigned to UAV j , and $x_{ij} = 0$ otherwise.

(3) Maximizing coverage

Assuming that each task T_i has a specific area or target point to cover, the coverage rate can be expressed as C_i and then the maximized total coverage rate can be expressed as:

$$\max \sum_{i=1}^N C_i \cdot y_i \quad (3)$$

where y_i is a binary variable indicating whether the task T_i has been overwritten or not.

(4) Comprehensive objectives

If there is a need to balance multiple metrics such as task completion time, energy consumption, and coverage, a weighted sum can be used:

$$\min \sum_{i=1}^N (w_1 \cdot f_i + w_2 \cdot \sum_{j=1}^M x_{ij} \cdot E_{ij} + w_3 \cdot C_i \cdot y_i) \quad (4)$$

where w_1, w_2, w_3 are the weights of each indicator, ensuring that the priorities of the indicators are weighed. The task completion time f_i is typically calculated based on the flight

distance to the task location and the UAV's average speed. The energy consumption E_{ij} is estimated using a model that considers flight time, hover time, and the UAV's power profile.

2.3. UAV Cluster Objective Function

In the three-dimensional multi-UAV transportation and delivery task allocation scenario, the scheduling center needs to formulate a task allocation strategy with the help of a task allocation algorithm to maximize the number of UAVs completing the task and, at the same time, ensure that the rewards of task execution are maximized and the costs are minimized. In this subsection, the objective function of UAVs performing transportation and delivery tasks is divided into two dimensions, reward and cost, considering the attribute characteristics of UAVs and the attribute characteristics of tasks.

The distance equation when the UAV flies from its current position to the target position is shown in (5):

$$L_{UtoT} = \sqrt{(X_T - X_{UAV})^2 + (Y_T - Y_{UAV})^2 + (H_T - H_{UAV})^2} \quad (5)$$

where L_{UtoT} is the distance between the current position of the UAV and the target position.

In addition to this, the UAV flight time is affected by its own weight and load weight as shown in Equation (6):

$$t_{Tfly} = \frac{L_{UtoT}}{V_{UAV} - V_T} * \frac{W_{UAV} + W_T}{W_{UAV}} \quad (6)$$

where t_{Tfly} is the UAV flight time.

The UAV needs to unload and load the cargo after arriving at the target location, and let the length of the mission sequence of this UAV be k . The time cost T_{total} for the UAV to perform the mission is shown in Equation (7):

$$T_{total} = \sum_{i=1}^k (t_{Tfly} + t_T) \quad (7)$$

The cost required for the entire UAV cluster to perform the mission can be converted into the final UAV cluster objective function by accumulating the combined rewards of individual UAVs, as shown in Equation (8):

$$R_{total} = \sum_{i=1}^m \sum_{j=1}^n a_{ij} \left(\frac{r_T}{t_{Tfly} + t_T} \right) \quad (8)$$

where a_{ij} is the execution of task u_{ij} by UAV w_j , with 0 indicating no execution and 1 indicating execution.

3. Integrated UAV Scheduling Modeling

3.1. Model for Optimizing the Number of UAVs

Model Assumptions and Symbolic Definitions

- (1) Task requirements are clear and certain

That is, it is assumed that the needs of the task (e.g., the total amount of tasks, completion time requirements, etc.) are known and do not change during the optimization process. For example, the area to be covered by the mission, the amount of data collection, or the amount of cargo transportation is fixed.

- (2) Consistent performance of UAVs

It is assumed that all UAVs dispatched have the same performance parameters, including flight speed, endurance, load capacity, and energy consumption. This avoids the complexity of dealing with heterogeneous UAVs in the optimization process.

(3) Single-task completion model

It is assumed that all UAVs perform a single task (e.g., reconnaissance, transportation, or surveillance) and that the task can be decomposed into a number of subtasks to be completed in parallel. The task division among UAVs is uniform and reasonable, and there is no task allocation overload.

(4) Return of UAVs to the starting point

It is assumed that all UAVs must return to the starting point after completing the task, and the time and energy consumption of the return path are included in the total time and total energy consumption model.

(5) The number of dispatches has a nonlinear effect on mission efficiency and system reliability

It is assumed that the impact of the number of UAVs dispatched NNN on the mission completion time, total energy consumption, and system reliability, etc., conforms to the nonlinear relationship set in the model, e.g., $T(N) = \frac{T_{max}}{N} + T_{delay}(N)$, etc.

(6) Limited influence of the external environment

It is assumed that the influence of the external environment (e.g., weather, obstacles, terrain, etc.) on the UAV's mission execution during the mission is small or its influence is uniformly distributed and negligible. If environmental factors are considered, the relevant model can be corrected with a specific perturbation function.

(7) Total energy consumption is mainly determined by flight consumption

It is assumed that the total energy consumption of the UAV is mainly determined by the energy consumption during flight, ignoring the difference in energy consumption during ground stop, takeoff, or landing.

(8) Reliability is a decreasing function of quantity

It is assumed that the higher the number of UAVs dispatched, the lower the reliability of the mission system, which may be due to the increase in communication burden, management difficulty, etc., and that this relationship can be described by an exponential or other decreasing function.

(9) Costs grow linearly

It is assumed that the cost of dispatching each additional UAV (e.g., startup cost and maintenance cost) grows linearly, without taking into account the economy of scale effect or other nonlinear factors.

The relevant symbols for UAVs are shown in Table 2.

Table 2. Definition of UAV Scheduling Quantity Optimization Symbols.

Notation	Definition
N	Number of UAVs dispatched
$C(N)$	Dispatch cost functions
$T(N)$	Task time function
$E(N)$	Total energy consumption function
$R(N)$	Reliability function
$S(N)$	Security functions

The decision variable is the number of UAVs to be dispatched, N , and the objective is to minimize a composite objective function while satisfying constraints such as mission time and energy consumption.

3.2. Objective Function

Our goal is to find a suitable number N of UAVs that results in an optimal balance between total cost, total mission execution time, mission reliability, energy consumption, and security. This can be achieved by constructing a multi-objective optimization problem, but to facilitate the application of the GWO algorithm, we can transform these objectives into a single-objective optimization problem by means of weighted summation.

Let the objective function be $F(N)$, then it can be described as:

$$F(N) = w_1 \cdot C(N) + w_2 \cdot T(N) + w_3 \cdot \frac{1}{R(N)} + w_4 \cdot E(N) + w_5 \cdot \frac{1}{S(N)} \quad (9)$$

where, w_1, w_2, w_3, w_4, w_5 are weight coefficients, which are used to balance the importance between different objectives. These weights are set according to the actual application scenarios.

3.3. Define Each Subfunction

(1) Dispatch Cost Functions $C(N)$

The dispatch cost function is used to calculate the total cost of dispatching N UAVs. This typically includes the acquisition cost of the UAV (if long-term planning is considered), maintenance costs, operational costs (e.g., fuel, power, personnel operations, etc.), and possibly leasing costs.

$$C(N) = c_{\text{fixed}} \cdot N + c_{\text{variable}} \cdot N \cdot t \quad (10)$$

where c_{fixed} is the fixed cost per UAV, c_{variable} is the variable cost per UAV per hour, and t is the mission execution time.

(2) Task Time Function $T(N)$

The task time function is used to calculate the total time required to complete a particular task. When more drones are dispatched, the task may be able to be completed faster because the drones can work in parallel. However, coordination, communication and possible waiting times between UAVs also need to be considered.

$$T(N) = \max\left(\frac{T_{\text{total}}}{N}, T_{\text{min}}\right) \quad (11)$$

where T_{total} is the total workload of the task (expressed in time), N is the number of drones, and T_{min} is the minimum possible time to complete the task (taking into account coordination between UAVs, etc.). This function assumes that the task can be fully parallelized, but in practice, more complex models may be needed to accurately describe the task time.

(3) Reliability Function $R(N)$

The reliability function is used to assess the probability of success of sending N UAVs to complete the mission. Increasing the number of UAVs may improve the reliability of the mission because even if one or more UAVs fail, the other UAVs can still complete the mission.

$$R(N) = 1 - (1 - r)^N \quad (12)$$

where r is the reliability of a single UAV in accomplishing a mission. This function assumes that failures are independent between UAVs.

(4) Total Energy Consumption Function

The total energy function is used to calculate the total energy consumption required to send N UAVs to accomplish the mission. This includes the energy consumption of all activities such as UAV flight, hovering, communication, etc. [27].

$$E(N) = e_{\text{flight}} \cdot N \cdot t + e_{\text{hover}} \cdot N \cdot t_{\text{hover}} + e_{\text{comm}} \cdot N \cdot t \quad (13)$$

where e_{flight} is the energy consumption rate per UAV in flight, t is the flight time, e_{hover} is the energy consumption rate per UAV in hover, t_{hover} is the hover time, and e_{comm} is the energy consumption rate per UAV in communication.

(5) Security Functions

The safety function is used to evaluate how safe it is to send N drones on a mission. Developing robust safety and security evaluation models is a critical aspect of autonomous UAV operations, encompassing both situational monitoring and assessments based on airborne data [28,29]. Increasing the number of drones may increase the safety risks, such as airborne collisions and ground accidents.

$$S(N) = 1 - \frac{N \cdot p}{N_{\max}} \quad (14)$$

where p is the probability of an accident involving a single drone and $N_{\max}$ is the maximum number of drones allowed. This function assumes that the safety risk increases exponentially with the number of drones.

3.4. Constraints

When optimizing the number of UAVs dispatched, the constraints of the UAVs need to be considered, and the constraints in the UAV scheduling process, which are mainly considered in this section, include the following:

(1) UAV number constraints

The drone number constraint ensures that the number of drones dispatched is within acceptable limits. $N_{\min}$ is the minimum number of drones, which may be determined by mission requirements or technical constraints and $N_{\max}$ is the maximum number of drones, which may be limited by resources (e.g., number of available drones, budget, etc.) or regulations (e.g., airspace management, flight restrictions, etc.).

$$N_{\min} \leq N \leq N_{\max} \quad (15)$$

(2) Cost budget constraints

The cost budget constraint ensures that the total cost of fielding the UAV does not exceed the budget B . The cost function $C(N)$ includes the acquisition, maintenance, operation, and possible rental costs of the UAV. This constraint is critical to ensure the economic feasibility of the project.

$$C(N) \leq B \quad (16)$$

(3) Task time constraints

Task time constraints ensure that tasks are completed within a specified time. The task time function $T(N)$ represents the total time required to complete a particular task, while $T_{\max}$ is the maximum completion time allowed for the task. This constraint is important for meeting the demands of time-sensitive tasks.

$$T(N) \leq T_{\max} \quad (17)$$

(4) Energy consumption constraints

The energy consumption constraint ensures that the total energy consumption of the dispatched UAV does not exceed the limit $E_{\max}$. The energy consumption function $E(N)$ includes the energy consumption for activities such as UAV flight, hovering, and communication. This constraint is important to ensure the sustainable use of energy and to minimize the environmental impact.

(5) Reliability requirement constraints

The reliability requirement constraint ensures that the reliability of the mission meets or exceeds the minimum requirement R_{min} . The reliability function $R(N)$ represents the probability of success of sending N UAVs to accomplish the mission. This constraint is important to ensure a high success rate of the mission and to minimize the risk of failure.

$$R(N) \geq R_{min} \quad (18)$$

(6) Safety requirement constraints

The safety requirement constraint ensures that the safety of the dispatched UAVs meets or exceeds the minimum requirement S_{min} . The safety function $S(N)$ represents the level of safety when N UAVs are dispatched on a mission. This constraint is important to ensure the safety of UAV operations, reduce the risk of accidents, and protect people and property.

$$S(N) \geq S_{min} \quad (19)$$

3.5. Optimization Problems

Combined with the other models in Section 2.3, the optimization model for the number of drones can be described as follows:

$$\min_{N \in \mathbb{Z}^+} F(N) \quad (20)$$

The constraints are as follows:

$$\begin{cases} N_{min} \leq N \leq N_{max} \\ C(N) \leq B \\ T(N) \leq T_{max} \\ E(N) \leq E_{max} \\ R(N) \geq R_{min} \\ S(N) \geq S_{min} \end{cases} \quad (21)$$

Based on this optimization problem, in subsequent simulation experiments, the number of UAVs is optimized in terms of dispatch cost, mission time, reliability, total energy consumption, and safety.

4. The Proposed Two-Stage GWO-CBBA Framework

To address the coupled problem of fleet sizing and task allocation, we propose a two-stage optimization framework that leverages the strengths of both a global metaheuristic (GWO) and a distributed consensus algorithm (CBBA). The framework, illustrated in Figure 1, decouples the problem into a strategic optimization stage and a tactical allocation stage.

4.1. Stage 1: GWO for Optimal Fleet Sizing

The first stage addresses the strategic question of how many UAVs to dispatch. We formulate this as a single-variable optimization problem where the goal is to find the optimal number of UAVs, N , that minimizes a comprehensive system-level objective function, $F(N)$.

The objective function $F(N)$ (previously defined in Equation (9)) is a weighted sum of key performance indicators:

$$F(N) = w_1 \cdot C(N) + w_2 \cdot T(N) + w_3 \cdot R(N) \quad (22)$$

where $C(N)$, $T(N)$, $E(N)$, $R(N)$, and $S(N)$ represent the dispatch cost, mission time, energy consumption, reliability, and security as functions of the fleet size N .

The Grey Wolf Optimizer (GWO), a metaheuristic inspired by the hunting behavior of grey wolves, is used to solve this problem. GWO is well-suited for this task due to its strong global search capabilities and fast convergence. It explores the search space of possible N values (e.g., from $N_{\min}$ to $N_{\max}$) and identifies the optimal fleet size, N^* , that minimizes $F(N)$. This N^* represents the best trade-off between resource cost and mission performance.

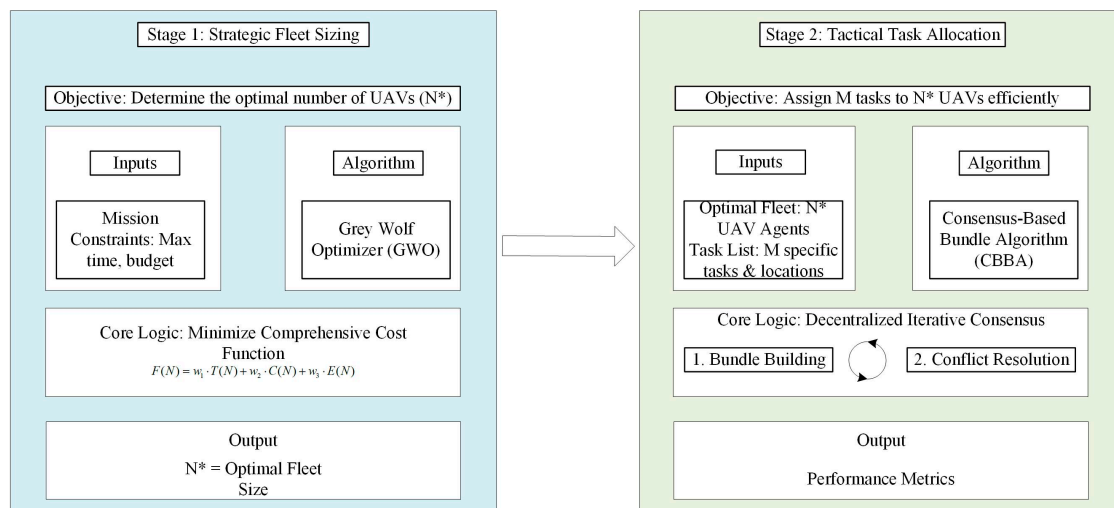


Figure 1. The proposed two-stage optimization framework. Stage 1 employs the Grey Wolf Optimizer (GWO) at a strategic level to determine the optimal fleet size (N^*) by minimizing a comprehensive cost function that balances operational costs and mission efficiency. Stage 2 then utilizes the Consensus-Based Bundle Algorithm (CBBA) at a tactical level for decentralized, iterative task allocation among the N^* deployed UAVs, resulting in a conflict-free final schedule.

4.2. Stage 2: CBBA for Distributed Task Allocation

Once the optimal fleet size N^* is determined in Stage 1, it is used to initialize the second stage. A fleet of N^* UAV agents is created, and the tactical problem of assigning the M tasks to these agents is solved using the Consensus-Based Bundle Algorithm (CBBA).

4.2.1. Communication Model

For the CBBA stage, we assume the N^* UAVs form a communication network represented by an undirected graph $G = (V, E)$, where V is the set of UAVs. An edge (i, j) exists in E if UAV i and UAV j are within a predefined communication range. In our simulations, we assume ideal communication (no delays or packet loss) within this range.

4.2.2. CBBA Execution

CBBA is an iterative, distributed algorithm. Each of the N^* agents maintains a bundle of tasks it plans to execute and a vector of scores for those tasks. The algorithm proceeds in two phases:

Bundle Building: Each agent locally and greedily adds tasks to its bundle based on a score function U_{ij} (Equation (22)), which calculates the utility of agent i performing task j . This score typically considers factors like distance, priority, and energy cost.

Conflict Resolution: Agents communicate their task bundles to their neighbors. If a task is claimed by multiple agents, a consensus rule is applied: the agent with the highest score for that task wins it, and others remove it from their bundles.

This process repeats until no more changes occur in the task assignments, at which point the allocation has converged to a conflict-free solution. The output of this stage is a final, decentralized task assignment for the N^* UAVs.

4.3. Algorithmic Implementation

The overall process of the GWO-CBBA framework is summarized in Algorithm 1.

Algorithm 1: GWO-CBBA Framework for UAV Scheduling

```
// Stage 1: GWO Fleet Size Optimization
1: Initialize GWO parameters (population size, max iterations) and search space for N.
2: Initialize a population of wolves with random N values.
3: Repeat until max iterations or convergence:
4:   For each wolf (candidate N):
5:     Calculate fitness F(N) using Equation (9).
6:   End For
7:   Update Alpha, Beta, and Delta wolf positions (best N values).
8:   Update all other wolf positions based on Alpha, Beta, and Delta.
9: End Repeat
10: Output optimal fleet size N* (position of the Alpha wolf).
// Stage 2: CBBA Task Allocation
11: Initialize N* UAV agents and the set of M tasks.
12: Repeat until task allocation is stable:
13:   // Bundle Building Phase (local to each agent)
14:   For each agent i from 1 to N*:
15:     Agent i selects the best task to add to its bundle based on score Uij.
16:   End For
17:   // Conflict Resolution Phase (communication)
18:   Agents broadcast their bundles to neighbors.
19:   Agents update their bundles to resolve conflicts according to consensus rules.
20: End Repeat
21: Output final, conflict-free task assignments for all N* agents.
```

The algorithm combines the Grey Wolf Optimizer (GWO) algorithm and the CBBA algorithm for solving the UAV mission scheduling problem.

The functions of each part of the GWO-CBBA algorithm proposed in this paper are as follows: the GWO algorithm is used to optimize the number of UAVs N to find the optimal number in a mission scenario to balance the factors of cost, efficiency, and safety. The CBBA algorithm is used to distribute the tasks based on the optimized number of UAVs to ensure that the results of the task allocation satisfy the task constraints and achieve the global optimum.

5. Simulation Verification

5.1. Experimental Setup

5.1.1. Scenario Definition

To validate the proposed GWO-CBBA framework, we designed a simulation based on a representative emergency logistics scenario. In this scenario, a set of 20 critical supply packages must be delivered from a central depot, located at (0,0), to 20 distinct task locations randomly distributed within a 10 km $\times$ 10 km operational area. The locations of the tasks are illustrated in Figure 2. Each UAV can carry one package at a time and must return to the depot after each delivery to pick up the next package.

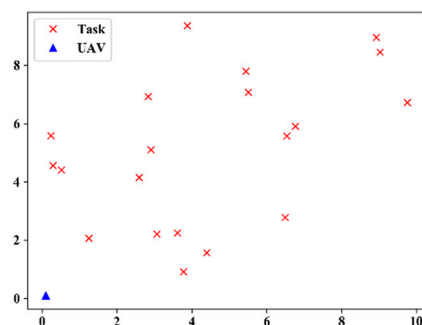


Figure 2. UAV scheduling simulation scenario.

The position information in the UAV scene is shown in Figure 2. The position information of the simulated scene is shown in Table 3.

Table 3. UAV Simulation Parameters Table.

Serial Number	UAV	1	2	3	4	5	6
Position/km	(0,0)	(5.5,7.1)	(2.9,5.1)	(8.9,9.0)	(1.3,2.1)	(0.5,4.4)	(0.3,4.6)
Serial number	7	8	9	10	11	12	13
Position/km	(6.5,2.8)	(6.8,5.9)	(0.2,5.6)	(2.6,4.2)	(2.8,6.9)	(4.4,1.6)	(5.4,7.8)
Serial number	14	15	16	17	18	19	20
Position/km	(3.1,2.2)	(3.9,9.4)	(9.8,6.7)	(9.0,8.5)	(3.8,0.9)	(6.5,5.6)	(3.6,2.3)

5.1.2. Model and Algorithm Parameters

The key simulation parameters are summarized in Table 4. For the GWO algorithm, we configured a population size of 30 wolves and a maximum of 50 iterations to ensure convergence. The search space for the optimal number of UAVs (N) was set from a minimum of 4 (required to complete 20 tasks with a maximum of 5 tasks per UAV) to a maximum of 15.

Table 4. UAV Simulation Parameters Table.

Parameter Name	Size
Flight speed	30 km/h
Maximum number of missions	5
Drone dispatch cost	10
Drone energy cost	0.1 /min
Maximum mission execution time of the drone	120 min
Number of wolves	100
Maximum number of exploration steps	100

5.1.3. Baseline Methods for Comparison

To rigorously evaluate the performance of our framework, we benchmark it against three widely recognized baseline methods:

1. **Standard CBBA (Fixed Fleet):** We execute the standard CBBA algorithm with a fixed, non-optimized fleet size of $N = 5$. This baseline serves to demonstrate the significant performance gains achieved by our GWO-based fleet sizing stage.

2. **Centralized PSO:** A powerful, centralized Particle Swarm Optimization (PSO) algorithm that simultaneously optimizes both fleet size and task allocation. This represents a high-performance centralized competitor, allowing us to compare solution quality and computational cost.

3. **Greedy Heuristic Algorithm:** A simple and fast centralized algorithm. It sequentially assigns each unassigned task to the UAV that can complete it the soonest. This baseline provides a measure of how much our sophisticated optimization improves upon a basic, intuitive approach.

5.2. Stage 1: GWO-Based Fleet Sizing Analysis

For the UAV scheduling problem, a UAV scheduling model based on the GWO-CBBA algorithm is proposed. The simulation in this section mainly includes a UAV scheduling quantity optimization simulation as well as a UAV task allocation strategy simulation. Since the upper limit of a single mission set by a UAV is 5, 20 missions need at least 4 UAVs to execute. Based on this, this simulation calculates the execution of 4–11 UAVs for 20 tasks respectively.

The results of scheduling for the UAVs in the simulation scenario created in this paper are as follows:

The results of UAV scheduling for different numbers of aircraft are depicted in Figure 3. The statistics of the task assignment process for each UAV are shown in Figure 4

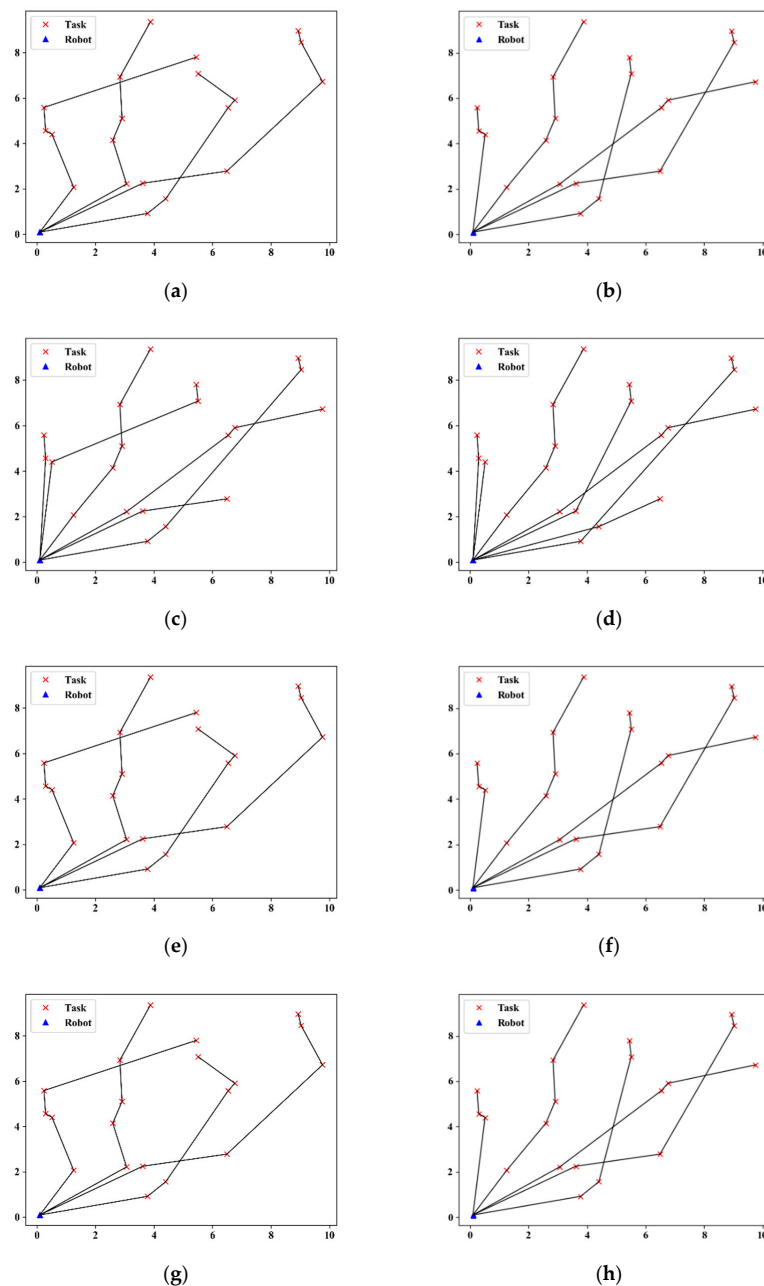


Figure 3. (a) Case of 4 drones dispatched; (b) 5 drones dispatched; (c) 6 drones dispatched; (d) 7 drones dispatched; (e) 8 drones dispatched; (f) 9 drones dispatched; (g) 10 drones dispatched; (h) 11 drones dispatched.

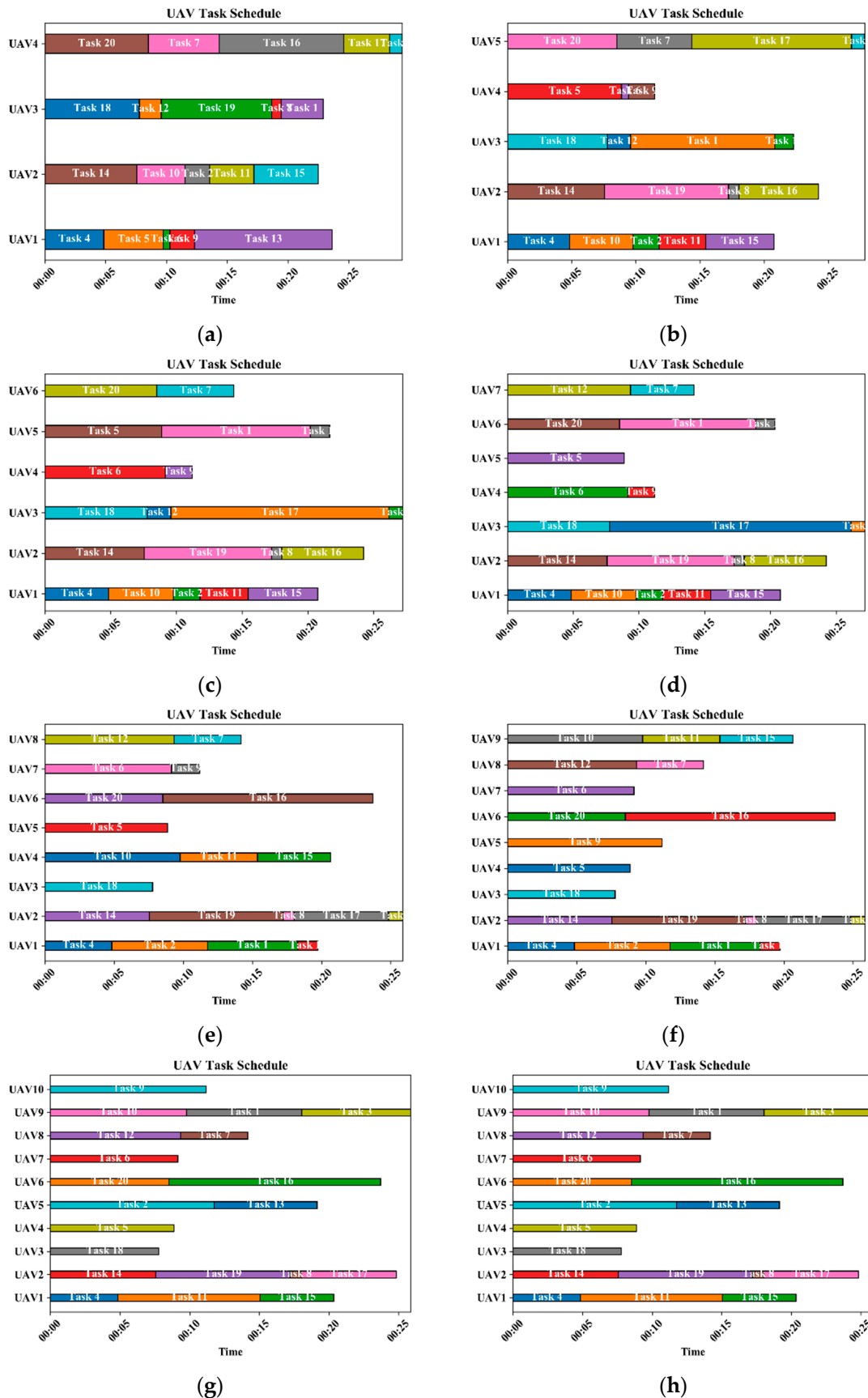


Figure 4. (a) Case of 4 drones dispatched; (b) 5 drones dispatched; (c) 6 drones dispatched; (d) 7 drones dispatched; (e) 8 drones dispatched; (f) 9 drones dispatched; (g) 10 drones dispatched; (h) 11 drones dispatched.

The statistical analysis of the number of miles flown by 4–11 UAVs is shown in Figure 5.

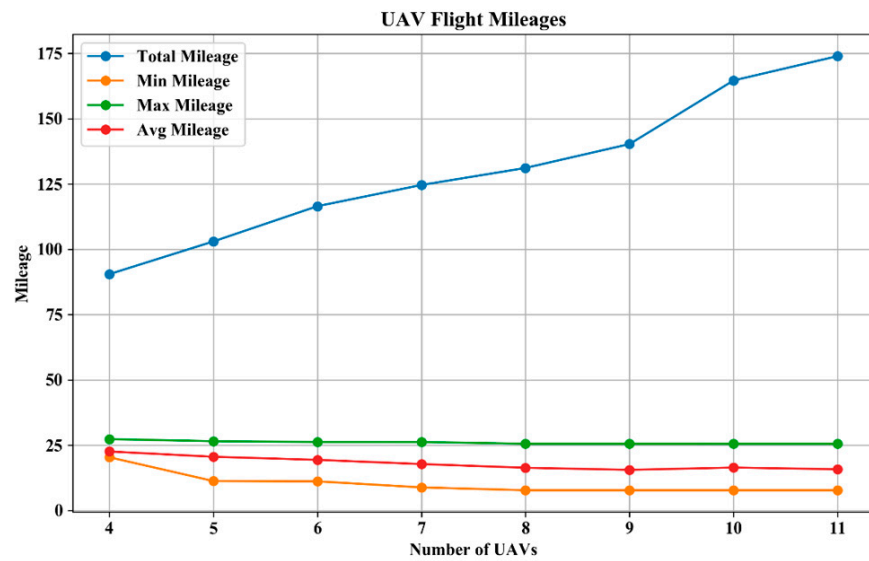


Figure 5. Analysis of the number of drones.

The average execution time cost per task can be expressed as shown in Table 5.

Table 5. UAV Simulation Results.

Serial Number	4	5	6	7	8	9	10	11
Time	16.84	14.87	14.72	14.57	14.34	14.34	14.27	14.26

As the number of UAVs increased from 4 to 11, the change in waiting time for individual tasks is shown in Figure 6.

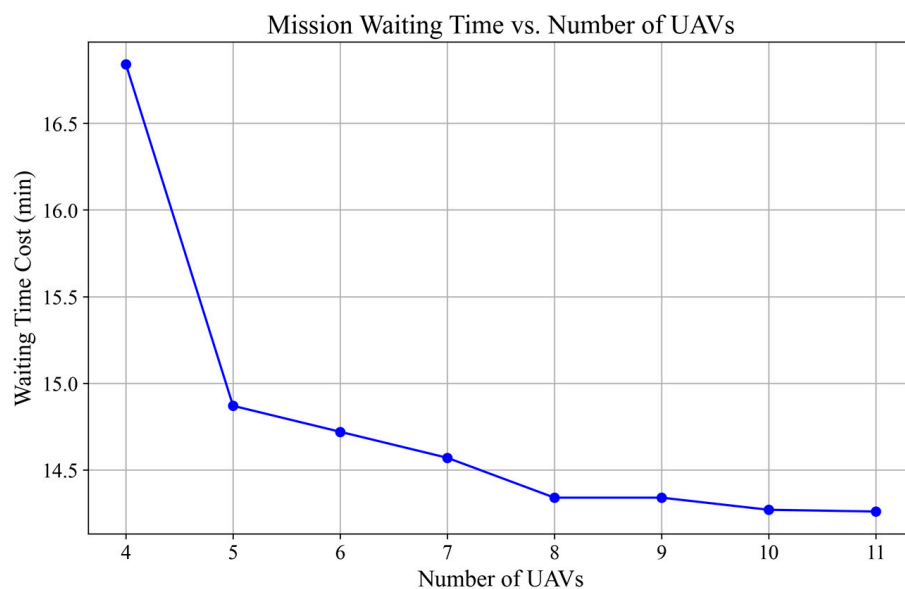


Figure 6. Curve of mission waiting time with the number of UAVs.

Establishment of UAVs' synthesized scheduling costs $F(N)$:

$$F(N) = w_1 \cdot T(N) + w_2 \cdot C(N) + w_3 \cdot E(N) \quad (23)$$

where $C(N)$ is the total cost of the UAV (path, communication overhead, etc.), $T(N)$ is the total time for mission completion, and $R(N)$ is the reliability of mission execution (higher is better).

The first stage of our framework leverages the GWO to determine the optimal fleet size, N^* . This is achieved by minimizing the comprehensive objective function $F(N)$ (defined in Equation (9)), which encapsulates the trade-offs between mission time, operational costs, reliability, and safety.

Figure 7 plots the value of the objective function $F(N)$ for different fleet sizes, representing the cost landscape that GWO explores.

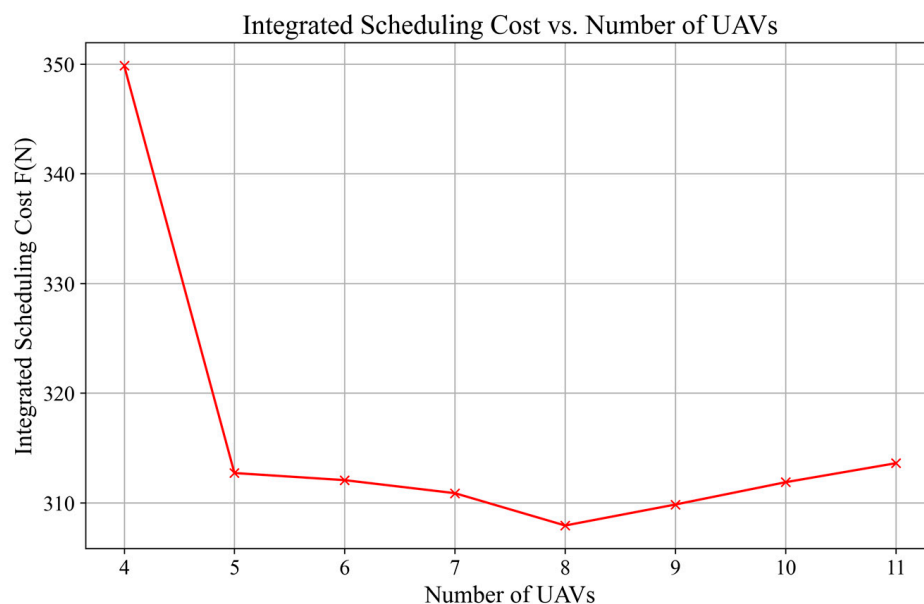


Figure 7. The integrated scheduling cost $F(N)$ as a function of the number of UAVs (N). The plot illustrates the trade-off between mission efficiency (lower cost for more UAVs initially) and system costs (higher cost as more UAVs are added). The GWO algorithm systematically searches this landscape and identifies the minimum cost at $N = 8$, which is determined to be the optimal fleet size (N^*).

As illustrated in Figure 7, the total system cost initially decreases as the fleet size grows from 4 to 8. This is because adding more UAVs drastically reduces the overall mission completion time (makespan), which is a major component of the cost function. However, beyond $N = 8$, the total cost begins to rise again. This ‘point of diminishing returns’ occurs because the marginal improvement in mission time becomes less significant, while the linear increase in dispatch and operational costs starts to dominate.

The GWO algorithm effectively identifies this inflection point. By finding the global minimum of the cost function, it is determined that $N = 8^*$ is the optimal fleet size for this specific mission scenario. This explicitly demonstrates that the optimal number is a direct output of our optimization algorithm, not a manual selection. This optimal value, N^* , is then passed to the second stage of our framework.

5.3. Stage 2: Task Allocation for the Optimal Fleet

With the optimal fleet size of $N^* = 8$ determined by GWO, the second stage employs the CBBA algorithm to perform a distributed and conflict-free task allocation. The 8 UAV agents collaboratively assign the 20 delivery tasks among themselves.

Figure 8 visualizes the final flight paths for the 8 UAVs, while Figure 9 presents the corresponding Gantt chart, illustrating the detailed schedule for each UAV.

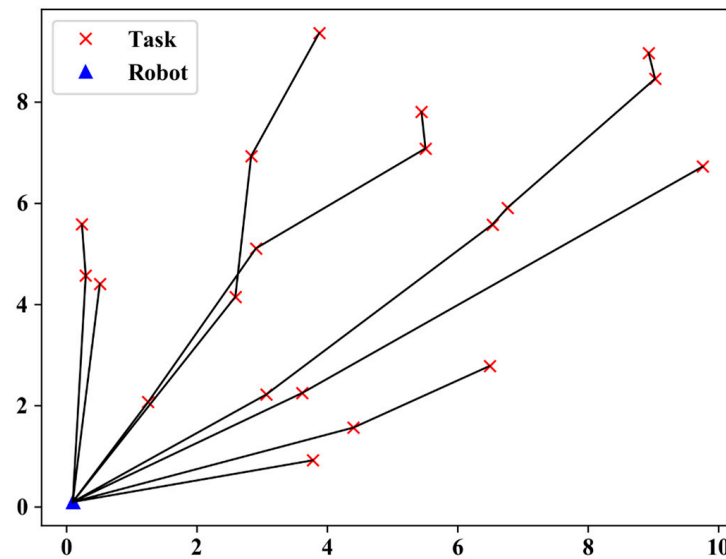


Figure 8. Final flight paths for the optimal fleet of 8 UAVs as determined by the GWO-CBBA framework.

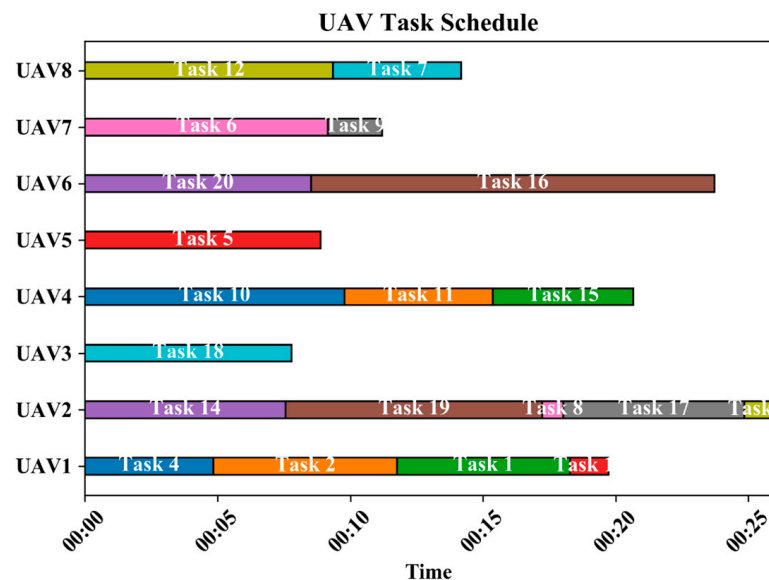


Figure 9. Gantt chart showing the task schedule for the optimal fleet of 8 UAVs.

The results in Figures 8 and 9 demonstrate an efficient and well-balanced workload distribution among the 8 UAVs, ensuring that all 20 tasks are completed in a coordinated and timely manner. The total time to complete all tasks (makespan) is recorded for comparison in the next section.

5.4. Comparative Analysis and Discussion

To evaluate the overall effectiveness of the proposed GWO-CBBA framework, a comparative analysis was conducted against three benchmark methods: (1) the standard Consensus-Based Bundle Algorithm (CBBA) with a fixed fleet size ($N = 5$), (2) a Centralized Particle Swarm Optimization (PSO) approach, and (3) a Greedy Heuristic algorithm. The comparison was based on three key performance indicators (KPIs): Overall Scheduling Cost, total UAV Mileage Cost, and Total Task Waiting Cost. The aggregated results of this comparison are presented in Figure 10.

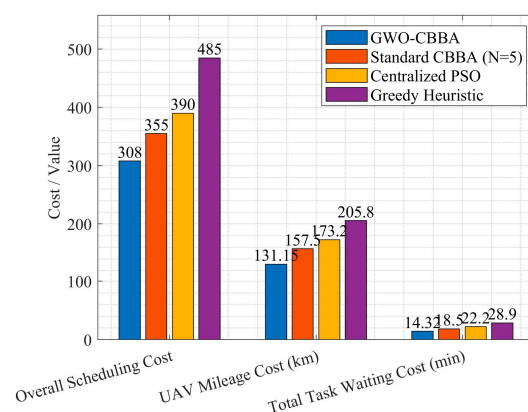


Figure 10. Performance Comparison of Different Algorithms.

As illustrated in Figure 10, the proposed GWO-CBBA framework demonstrates superior performance across all evaluated metrics, validating its effectiveness in solving the complex multi-UAV task allocation and scheduling problem.

Overall Scheduling Cost: The GWO-CBBA framework achieves the lowest overall scheduling cost of 308. This value is significantly lower than its counterparts, representing a reduction of 13.2% compared to the standard CBBA (355), 21.0% compared to the centralized PSO (390), and a substantial 36.5% compared to the Greedy Heuristic algorithm (485). This indicates that the dynamic fleet size optimization by GWO, combined with the efficient distributed task allocation by CBBA, produces a globally more cost-effective solution.

UAV Mileage Cost: In terms of resource consumption, the GWO-CBBA also yields the most efficient flight plans, with a total mileage cost of only 131.15 km. This is markedly better than the 157.5 km required by the standard CBBA and the 173.2 km by the centralized PSO. The Greedy Heuristic algorithm results in the highest mileage (205.8 km), which is expected due to its myopic decision-making process that fails to optimize routes from a global perspective. The efficiency of the GWO-CBBA stems from its ability to deploy the optimal number of UAVs, preventing the unnecessary travel that occurs when a fleet is oversized or the inefficient routes that result from an undersized fleet.

Total Task Waiting Cost: The GWO-CBBA framework minimizes the total task waiting cost to 14.32 min, showcasing its capability to provide timely service. This is a critical advantage over the standard CBBA (18.5 min), the centralized PSO (22.2 min), and particularly the Greedy Heuristic algorithm (28.9 min). The low waiting time is a direct consequence of the synergy between the GWO and CBBA. GWO identifies the appropriate fleet size to handle the task load effectively, while the CBBA ensures that tasks are assigned in a distributed and conflict-free manner, minimizing delays in execution. The fixed-fleet CBBA and centralized PSO struggle to strike this balance, leading to longer queues and increased waiting periods.

In summary, the results presented in Figure 10 provide compelling evidence that the proposed GWO-CBBA framework outperforms established and heuristic methods. Its primary advantage lies in its hierarchical approach, where the GWO metaheuristic intelligently optimizes the fleet size at a global level, while the CBBA efficiently manages decentralized task allocation at the local level. This synergy effectively balances operational costs and service quality, leading to a more robust and efficient solution for dynamic multi-UAV scheduling.

6. Conclusions

This paper has introduced a two-stage GWO-CBBA framework to solve the coupled problem of UAV fleet sizing and task allocation for emergency logistics missions. Our

approach first uses the Grey Wolf Optimizer (GWO) to determine the optimal number of UAVs at a global, strategic level, and then employs the Consensus-Based Bundle Algorithm (CBBA) for efficient, decentralized task assignment at a local, tactical level.

The primary contribution of this work is the demonstrated synergy between these two stages. Through rigorous simulations, we benchmarked our framework against standard fixed-fleet CBBA, a centralized PSO algorithm, and a Greedy Heuristic algorithm. The comparative analysis provided definitive evidence of our framework's superiority. Specifically, the GWO-CBBA approach reduced the overall scheduling cost by up to 36.5% compared to the Greedy Heuristic algorithm and by 13.2% against the fixed-fleet CBBA. Furthermore, it yielded the lowest UAV mileage cost and task waiting times, confirming its ability to produce solutions that are not only cost-effective but also timely.

In summary, the GWO-CBBA framework provides a robust and scalable solution that outperforms established methods by intelligently optimizing fleet size before allocating tasks. This hierarchical strategy effectively balances operational costs and service quality, offering a significant advancement for dynamic multi-UAV scheduling. Future work will focus on extending the framework to handle heterogeneous UAVs and incorporating more complex real-world constraints, such as no-fly zones and dynamic task arrivals.

7. Limitations and Future Work

While the GWO-CBBA framework shows promising results, certain limitations warrant further investigation. The current model assumes a homogeneous UAV fleet, ideal communication, a static environment with pre-defined tasks, and a simplified environmental model, which do not fully represent real-world complexities. Future research will focus on adapting the framework for heterogeneous UAVs, integrating realistic communication models, enabling real-time task injection in dynamic environments, incorporating geospatial and weather data, and validating the framework on a physical multi-UAV testbed to address hardware and communication challenges. Overcoming these limitations will improve the GWO-CBBA framework's applicability to a broader range of real-world challenges.

Author Contributions: Conceptualization, Y.Z. and Z.S.; methodology, Z.S.; writing—original draft preparation, Y.Z., H.Y. and W.X.; writing—review and editing, Z.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article material. Further inquiries can be directed to the corresponding author.

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

References

1. Linker, C.; Mena-Arciniega, C.; Xing, S. military applications, and opportunities of unmanned aerial vehicles. *Aviation* **2024**, *28*, 115–127. [\[CrossRef\]](#)
2. Alexan, W.; Aly, L.; Korayem, Y.; Gabr, M.; El-Damak, D.; Fathy, A.; Mansour, H.A.A. Secure communication of military reconnaissance images over UAV-assisted relay networks. *IEEE Access* **2024**, *12*, 78589–78610. [\[CrossRef\]](#)
3. Guo, J.; Zhang, J.; Wang, Z.; Liu, X.; Zhou, S.; Shi, G.; Shi, Z. Formation Cooperative Intelligent Tactical Decision Making Based on Bayesian Network Model. *Drones* **2024**, *8*, 427. [\[CrossRef\]](#)
4. Fang, Z.; Savkin, A.V. Strategies for optimized uav surveillance in various tasks and scenarios: A review. *Drones* **2024**, *8*, 193. [\[CrossRef\]](#)
5. Javed, S.; Hassan, A.; Ahmad, R.; Ahmed, W.; Ahmed, R.; Saadat, A.; Guizani, M. State-of-the-art and future research challenges in uav swarms. *IEEE Internet Things J.* **2024**, *11*, 19023–19045. [\[CrossRef\]](#)
6. Hawashin, D.; Nemer, M.; Gebreab, S.A.; Salah, K.; Jayaraman, R.; Khan, M.K.; Damiani, E. Blockchain applications in UAV industry: Review, opportunities, and challenges. *J. Netw. Comput. Appl.* **2024**, *230*, 103932. [\[CrossRef\]](#)

7. Wang, X.; Yin, S.; Luo, L.; Qiao, X. Research on Multi-UAV Task Assignment Based on a Multi-Objective, Improved Brainstorming Optimization Algorithm. *Appl. Sci.* **2024**, *14*, 2365. [\[CrossRef\]](#)
8. Ghaffar, M.A.; Peng, L.; Aslam, M.U.; Adeel, M.; Dassari, S. Vehicle-UAV Integrated Routing Optimization Problem for Emergency Delivery of Medical Supplies. *Electronics* **2024**, *13*, 3650. [\[CrossRef\]](#)
9. Zhang, Y.; Zhao, Q.; Mao, P.; Bai, Q.; Li, F.; Pavlova, S. Design and control of an ultra-low-cost logistic delivery fixed-wing UAV. *Appl. Sci.* **2024**, *14*, 4358. [\[CrossRef\]](#)
10. Zhang, Y.; Zhao, Q.; Mao, P.; Bai, Q.; Li, F.; Pavlova, S. Risk Assessment and Distribution Estimation for UAV Operations with Accurate Ground Feature Extraction Based on a Multi-Layer Method in Urban Areas. *Drones* **2024**, *8*, 399. [\[CrossRef\]](#)
11. Meng, Z.; Yu, K.; Qiu, R. Location-routing optimization of UAV collaborative blood delivery vehicle distribution on complex roads. *Complex Intell. Syst.* **2024**, *10*, 8127–8141. [\[CrossRef\]](#)
12. Shi, Y.; Yang, J.; Han, Q.; Song, H.; Guo, H. Optimal decision-making of post-disaster emergency material scheduling based on helicopter–truck–drone collaboration. *Omega* **2024**, *127*, 103104. [\[CrossRef\]](#)
13. Julio, J.S.; Silva, D.F.; Smith, A.E. Drone-enabled material handling in smart manufacturing. *Int. Trans. Oper. Res.* **2025**, *32*, 3296–3315.
14. Li, N.; Shi, Z.; Jin, J.; Feng, J.; Zhang, A.; Xie, M.; Min, L.; Zhao, Y.; Lei, Y. Design of Intelligent Firefighting and Smart Escape Route Planning System Based on Improved Ant Colony Algorithm. *Sensors* **2024**, *24*, 6438. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Debnath, D.; Vanegas, F.; Sandino, J.; Hawary, A.F.; Gonzalez, F. A Review of UAV Path-Planning Algorithms and Obstacle Avoidance Methods for Remote Sensing Applications. *Remote Sens.* **2024**, *16*, 4019. [\[CrossRef\]](#)
16. Zhang, J.-D.; Shi, Z.-Y.; Zhang, A.-L.; Yang, Q.-M.; Shi, G.-Q.; Wu, Y. UAV trajectory prediction based on flight state recognition. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *60*, 2629–2641. [\[CrossRef\]](#)
17. Luo, J.; Tian, Y.; Wang, Z. Research on unmanned aerial vehicle path planning. *Drones* **2024**, *8*, 51. [\[CrossRef\]](#)
18. Shi, Z.; Zhang, J.; Shi, G.; Ji, L.; Wang, D.; Wu, Y. Design of a UAV Trajectory Prediction System Based on Multi-Flight Modes. *Drones* **2024**, *8*, 255. [\[CrossRef\]](#)
19. Shi, Z.; Shi, G.; Zhang, J.; Wang, D.; Xu, T.; Ji, L.; Wu, Y. Design of UAV flight state recognition system for multisensor data fusion. *IEEE Sens. J.* **2024**, *24*, 21386–21394. [\[CrossRef\]](#)
20. Guo, H.; Miao, Z.; Ji, J.; Pan, Q. An effective collaboration evolutionary algorithm for multi-robot task allocation and scheduling in a smart farm. *Knowl.-Based Syst.* **2024**, *289*, 111474. [\[CrossRef\]](#)
21. Yasser, M.; Shalash, O.; Ismail, O. Optimized decentralized swarm communication algorithms for efficient task allocation and power consumption in swarm robotics. *Robotics* **2024**, *13*, 66. [\[CrossRef\]](#)
22. Lu, Y.; Da, C. Global and local path planning of robots combining ACO and dynamic window algorithm. *Sci. Rep.* **2025**, *15*, 9452. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Daviran, M.; Maghsoudi, A.; Ghezelbash, R. Optimized AI-MPM: Application of PSO for tuning the hyperparameters of SVM and RF algorithms. *Comput. Geosci.* **2025**, *195*, 105785. [\[CrossRef\]](#)
24. Ma, B.; Pan, Y.; Xu, Y.; Gao, Z.; Zhang, Z.; Chen, C.; Li, C. UAV-Assisted Computing Power Network Task Allocation and 3D Urban Trajectory Optimization. *IEEE Internet Things J.* **2025**, *12*, 19294–19307. [\[CrossRef\]](#)
25. Yan, S.; Pan, F.; Shi, Y.; Xu, J.; Zhang, D. Research on Task Allocation of Heterogeneous UAVs Based on DLR-CBBA. In Proceedings of the 2024 4th International Conference on Computer, Control and Robotics (ICCCR), Shanghai, China, 19–21 April 2024; IEEE: New York, NY, USA, 2024.
26. Shi, Z.; Jia, Y.; Shi, G.; Zhang, K.; Ji, L.; Wang, D.; Wu, Y. Design of motor skill recognition and hierarchical evaluation system for table tennis players. *IEEE Sens. J.* **2024**, *24*, 5303–5315. [\[CrossRef\]](#)
27. Liu, C.H.; Chen, Z.; Tang, J.; Xu, J.; Piao, C. Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2059–2070. [\[CrossRef\]](#)
28. Shi, Z.; Zhang, J.; Shi, G.; Zhu, M.; Ji, L.; Wu, Y. Autonomous UAV Safety Oriented Situation Monitoring and Evaluation System. *Drones* **2024**, *8*, 308. [\[CrossRef\]](#)
29. Shi, Z.; Liu, M.; Yang, Q.; Liu, J.; Wang, D.; Zhang, J. Autonomous security evaluation model for UAV based on airborne information. In Proceedings of the 2023 6th International Conference on Information Communication and Signal Processing (ICICSP), Xi'an, China, 23–25 September 2023; IEEE: New York, NY, USA, 2023; pp. 785–789.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.