ORIGINAL RESEARCH **Open access**

Optimization of Multi-Target Detection and Path Planning in Air–Ground Collaborative UAV–UGV Systems

Nora Schneider[1*], Tobias Frank[1], Andreas Müller[2], Christoph Meier[3]

Abstract

Joint exploration by multiple unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) constitutes a vital research area within multi-agent systems. However, investigations into the assignment of multi-target detection responsibilities and the design of collective paths remain insufficient, despite their critical importance for improving exploration efficiency and reducing energy expenditure. This study presents a formulation for an air–ground cooperative multi-target detection task based on Mixed Integer Linear Programming (MILP). To enhance the formulation's alignment with actual operational settings, constraints representing UAV and UGV kinematics, dynamic collision avoidance, task-to-agent allocation, and obstacle circumvention are embedded. An objective function that jointly evaluates time outlay, energy draw, and path smoothness is constructed to improve the model's fidelity and achieve a more operationally relevant goal. Concurrently, a Branch-and-Bound scheme merged with an Improved Genetic Algorithm (IGA-B&B) is devised to optimize the objective function, thereby deriving optimal task assignments and optimal paths for air–ground cooperative multi-target detection. A multi-agent, multi-obstacle, multi-task-point simulation arena is set up. Findings indicate that the IGA-B&B strategy reduces computational time by 30% compared to the classical Branch-and-Bound (B&B) approach. Furthermore, an outdoor physical trial is conducted, offering additional proof of the proposed scheme's validity and practicality.

Keywords Air–ground cooperative system, Task assignment, Swarm path design, MILP

*Correspondence:
Nora Schneider
nora.schneider@natfak.uni-freiburg.de
[1] Department of Artificial Intelligence for Organic and Regenerative Farming Practices, Faculty of Agriculture, University of Freiburg, Freiburg, Germany
[2] Department of Soil Carbon Sequestration Data Science, Faculty of Agriculture, Heidelberg University, Heidelberg, Germany
[3] Department of Smart Cover Crop Selection, Faculty of Agriculture, University of Hohenheim, Stuttgart, Germany

Introduction

Lately, interest in heterogeneous multi-agent cooperative systems has surged, owing to their varied functionality, broad applicability, and notable resilience [1]. As a distinct subclass of heterogeneous frameworks, an air–ground cooperative system merges the expansive observation reach and rapid maneuverability of multi-UAVs with the precise inspection capacity and prolonged staying power of multi-UGVs. Fields of application span patrolling [2], rescue operations [3], reconnaissance and attack missions [4], parcel distribution [5], and further military or civilian domains.

Within practical deployment contexts, UAVs and UGVs are frequently directed to reach pre-specified locations to carry out tasks, a requirement that inherently involves both task-to-agent allocation and trajectory computation. UAVs and UGVs are autonomous entities with dissimilar motion parameters and significant functional disparities. To formulate solutions for

This is an open access article distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0), which permits non-commercial use, distribution, reproduction, and adaptation in any medium or format, provided that appropriate credit is given to the original author(s) and source, a link to the license is provided, and any changes made are indicated. Adapted material must be distributed under the same license.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

such optimization problems that more authentically reflect reality, model construction must take into account both their shared properties and distinguishing characteristics in full measure.

The Traveling Salesman Problem (TSP), among the most emblematic task allocation formulations, was first employed in settings where solely UAVs or solely UGVs conducted operations. Within the TSP, urban locations correspond to task-execution target points, while inter-city links correspond to movement paths. Murray and Chu [5] proposed the Flying Partner Traveling Salesman Model (FSTSP) alongside the Parallel Drone Dispatching Traveling Salesman Model (PDSTSP), which provide theoretical underpinnings for addressing the last-mile problem in air–ground collaborative parcel delivery. Mara [6] formulated the Traveling Salesman Problem with Drone (TSP-D), akin to the FSTSP, and added operational-range constraints for UAVs. In Refs. [7-9], adaptations of the aforementioned models and diverse solution techniques are introduced.

The Vehicle Routing Problem (VRP), which resembles the TSP, has likewise been applied and extended within the air–ground coordination domain. Wang [10] introduced the vehicle routing problem with drones (VRPD), analogous to the FSTSP, to model the situation in which UAVs are transported alongside trucks. The work in [11] introduced multiple UAVs, generalized the formulation to VRPD-MD, and enabled cooperative path computation for UAVs and UGVs. Moreover, refs. [12, 13] utilized valid inequalities (VIEQ) and iterative local search with variable neighborhood descent (ILS-VND) to quicken the solving process for such problem classes, strengthening the formulation's usefulness in practice.

Still, the TSP, VRP, and their derivatives remain constrained by their foundational structures, making it difficult to expand into highly constrained formulations. That largely explains why most TSP and VRP formulations remain concentrated on delivery use cases. Diverging from these two model categories, Mixed Integer Linear Programming (MILP) does not assume a pre-defined base structure, and any constraint that can be expressed in linear or integer form may be incorporated into the MILP construct. Consequently, the robust extensibility of MILP formulations permits their broad adoption in tackling multi-objective optimization tasks.

Beyond that, the works cited above can be recast as a MILP representation [14-16] that integrates UAV collision-prevention requirements and mission timing restrictions into the MILP framework, bringing this family of mathematical models closer to reality. Later, in Ref. [5], various heterogeneous UAV types suitable for anti-radar missions were incorporated into the MILP formulation, while task completion deadlines were enforced via time-window constraints. In [17], UGVs were introduced as mobile energy resuppliers for UAVs, thereby extending the operational window of the aerial units.

From the foregoing survey, it is clear that MILP is exceptionally well-suited to capture air–ground coordination scenarios. Yet the range of scenarios covered by existing models stays fairly limited—a relatively comprehensive model exists essentially for the case of ground vehicles transporting drones for delivery—and a broadly applicable formulation remains urgently needed. Therefore, this paper delivers the following contributions:

(1) A novel MILP-based air–ground cooperative multi-target detection task model is formulated, embedding kinematic restrictions, dynamic collision avoidance constraints, task allocation rules, and obstacle avoidance conditions, to portray the air–ground cooperative environment coverage challenge in a more lifelike and reasoned fashion.

(2) A richer objective function that jointly captures mission completion time, energy outlay during operations, and path trajectory evenness is put forward, aimed at yielding the optimal outcome for air–ground cooperative multi-target detection task allocation and path design.

(3) A Branch-and-Bound method fused with an Improved Genetic Algorithm (IGA-B&B) is designed to tackle the objective function, enabling the derivation of the optimal task distribution and optimal paths for air–ground cooperative multi-target detection. Within this approach, the chromosome characteristics and genetic operators of the Genetic Algorithm are modified to better adapt the Improved Genetic Algorithm to the model and to speed up computation.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

(4) A simulation testbed incorporating multiple agents, multiple obstacles, and multiple task points, alongside additional features, is built to substantiate the effectiveness and superiority of the proposed methodology. Simulation results show that the presented IGA-B&B approach can reduce runtime by 30% compared to the conventional B&B approach. In addition, practical value and feasibility are further attested through outdoor experiments.

The remainder of this paper is arranged as follows. Related work is covered in Section 2. The proposed approach is expounded in Section 3 and Section 4, covering the MILP-based task allocation and path optimization model for air–ground cooperative multi-target detection, the integrative construction of the objective function, and the optimization function solving methodology based on IGA-B&B. Section 5 describes the simulation and experimental setup, and the evaluation criteria and experimental results are dissected and contrasted, confirming the soundness and feasibility of the presented approach. Finally, the paper concludes in Section 6.

Related Work

This section examines earlier studies on task assignment and route design for heterogeneous multi-agent collectives.

Task assignment

For both UAVs and UGVs, a wide range of task assignment formulations is available, including the Traveling Salesman Model, Assignment Problem, Transportation Problem, and Vehicle Routing Problem. To be more precise, the situation of a lone UAV executing multiple jobs in time-ordered sequence can be formalized as a Traveling Salesman Model or Vehicle Routing Problem model, whereas the situation of multiple UAVs undertaking several jobs can be formalized as an Assignment Problem model. Frequently adopted solving strategies are split into two branches. One consists of global optimization methods, such as the Branch-and-Bound technique and set-covering approaches. The other comprises local optimum methods, including Tabu Search (TS), Simulated Annealing (SAA), Genetic Algorithm (GA), Ant Colony Optimization (ACO), and neural networks.

Under centralized coordination, Wu *et al.* [18] proposed an Improved Genetic Algorithm based on a multi-type gene scheme to address the challenge of task distribution in heterogeneous UAV collectives. By structuring distinct job categories into parallel gene segments and crafting selection, crossover, and mutation operators tailored to multiple gene sets, the heterogeneous multi-UAV cooperative task distribution problem was resolved via GA. Kim *et al.* [19] proposed a distributed scheme based on a probabilistic choice mechanism inspired by the response threshold model for area search formation and job distribution among heterogeneous UAVs, achieving swift, adaptable area coverage and task assignment. Zhang *et al.* [20] designed a concurrent contract network protocol (CNP). By refining the auctioneer's bidding approach, using the auctioneer as a reference for filtering high-caliber bidders, and incorporating a concurrency feature that permits simultaneous tendering of multiple jobs, this protocol improves negotiation effectiveness, reduces communication burden and frequency, and can be practically deployed in settings with elevated real-time constraints. Yan *et al.* [21] introduced a cooperative particle swarm method anchored in the contract net protocol, factoring in the resource demands of target locations and UAV asset pools, which can effectively address the challenge of real-time job allocation in heterogeneous multi-UAV setups.

Path planning

Present-day path planning approaches can also be sorted into two branches: conventional techniques and heuristic techniques. Well-known conventional techniques include the Dijkstra algorithm, A algorithm, probabilistic roadmap approach, and sampling-driven rapidly exploring random tree approach. The use of heuristic algorithms in path planning parallels their use in task assignment, with representative algorithms including the Genetic Algorithm (GA), Ant Colony Optimization (ACO), and others.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Chen *et al.* [22] proposed a blended ACO-GA approach to determine the optimal route for the air–ground cooperative delivery scenario. By disentangling the UAV and UGV trajectories through an ACO-first, GA-second procedure, workable routes are assigned while honoring the delivery ordering. To overcome the drawbacks of substantial memory demand and lengthy computation time of the A algorithm when applied to larger-scale settings, Wang *et al.* [23] put forward an A algorithm augmented with jump point search, which can pick out salient jump points; greater leaps are realized by disregarding points throughout the process that are unnecessary to examine. Moreover, with the growing adoption of reinforcement learning in path planning, a deep reinforcement learning scheme based on echo state network (ESN) units was introduced in Ref. [24]. This scheme empowers UAVs to resolve unseen network states by leveraging rewards collected from earlier states, exhibiting predictive properties, and thus lending itself to a broad array of deployment scenarios in dynamic environments. Liu *et al.* [25] applied the K-means clustering algorithm to decompose the multi-TSP formulation into multiple stand-alone TSP formulations. Then they improved the efficiency of solving these formulations by refining the Genetic Algorithm.

Mathematical Model Description

Air–ground cooperative multi-target detection task model

Given that MILP can only address linearly describable problems, the conditions of UAVs and unmanned ground vehicles must be represented in a linear, discrete manner. To keep the problem model simple and to accelerate computational performance, the assumptions enumerated below are followed in this paper.

- Every agent is modeled as a point mass, and agents of the same class possess identical dynamic parameters;
- Every UAV maintains a uniform flight altitude, and the ground underlying all UGVs is even;
- Every agent retains adequate energy while carrying out operations;
- The impact of environmental factors such as slick roadways, aerodynamic drag, and sudden wind gusts is not accounted for.

In actuality, these assumptions only modestly curtail the motion agility of UAVs and UGVs. Nonetheless, all assumptions stay within the confines of practical settings, and the eventual outcomes remain transferable to real-world situations.

Under such conditions, the air–ground cooperative multi-target detection challenge can be outlined as follows: Nd UAVs and Nv UGVs proceed to Nt target positions to execute detection tasks. The target positions fall into four classes: $Nt1$ target positions tailored for UAVs, $Nt2$ target positions tailored for UGVs, $Nt3$ target positions suitable for either class of agent, and $Nt4$ target positions requiring both classes of agent. The tally of target positions conforms to:

$$N_t = N_{t1} + N_{t2} + N_{t3} + N_{t4} \quad (1)$$

Temporal progression is captured through the time step Δt and the count of time steps t.

Every variable listed above is in **Table 1**. The problem formulation captures the situation where these UAVs and UGVs cover all target positions subject to the constraint that the maximum number of time steps equals T. Beyond that, we embedded diverse constraints into the MILP to improve the proposed formulation's ability to represent the true physical process of multi-target detection in an air–ground cooperative system.

Table 1. Variables of the air–ground cooperative system.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Symbol	Description
Nd	Total number of UAVs
Nv	Total number of UGVs
Nt	Total number of target points
$Nt1$	Count of target points that are suitable for UAVs
$Nt2$	Count of target points that are suitable for UGVs
$Nt3$	Count of target points that can be handled by both UAVs and UGVs
$Nt4$	Count of target points that must be handled by both UAVs and UGVs
Δt	Duration of a single time step
t	Total number of time steps
T	Maximum allowed number of time steps

Kinematics model constraint

When vertical displacement is excluded, UAVs and UGVs exhibit analogous kinematic behavior. In this work, we lay out only the derivation path for the kinematics model of UAVs, which exhibit more involved kinematic behavior, and it can be written as:

$$\begin{bmatrix} x(t+1) \\ y(t+1) \\ v_x(t+1) \\ v_y(t+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \\ v_x(t) \\ v_y(t) \end{bmatrix} + \begin{bmatrix} \frac{(\Delta t)^2}{2} & 0 \\ 0 & \frac{(\Delta t)^2}{2} \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a_x(t) \\ a_y(t) \end{bmatrix} \quad (2)$$

Where $x(t)$ stands for lateral displacement, $y(t)$ stands for longitudinal displacement, $vx(t)$ stands for lateral velocity, $vy(t)$ stands for longitudinal velocity, $ax(t)$ stands for lateral acceleration, and $ay(t)$ stands for longitudinal acceleration. **Eq. 2** can be rewritten succinctly as:

$$S_p(t+1) = AS_p(t) + Ba_p(t) \quad (3)$$

Where (t) is the state matrix of the UAV at time step t, covering displacement and velocity, and (t) is the acceleration matrix of the pth UAV at time step t. A and B denote the coefficient matrices that depend solely on the time step size Δt. The acceleration parameter serves as the pivotal variable in the model, and the state depends exclusively on the starting position, starting velocity, and the acceleration applied at each time step.

Velocity and acceleration constraints

Under realistic conditions, the velocity and acceleration of UAVs and UGVs are capped by their own dynamic performance limits. In this work, we take the maximum speed to be $vmax$ and the maximum acceleration to be $amax$.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Within a Cartesian coordinate framework, the velocity and acceleration bounds are delineated by circles whose radii correspond to the maximum velocity and maximum acceleration, as shown in **Figure 1**. The equation defining the circle is nonlinear and must be linearized before it can be embedded into the MILP.

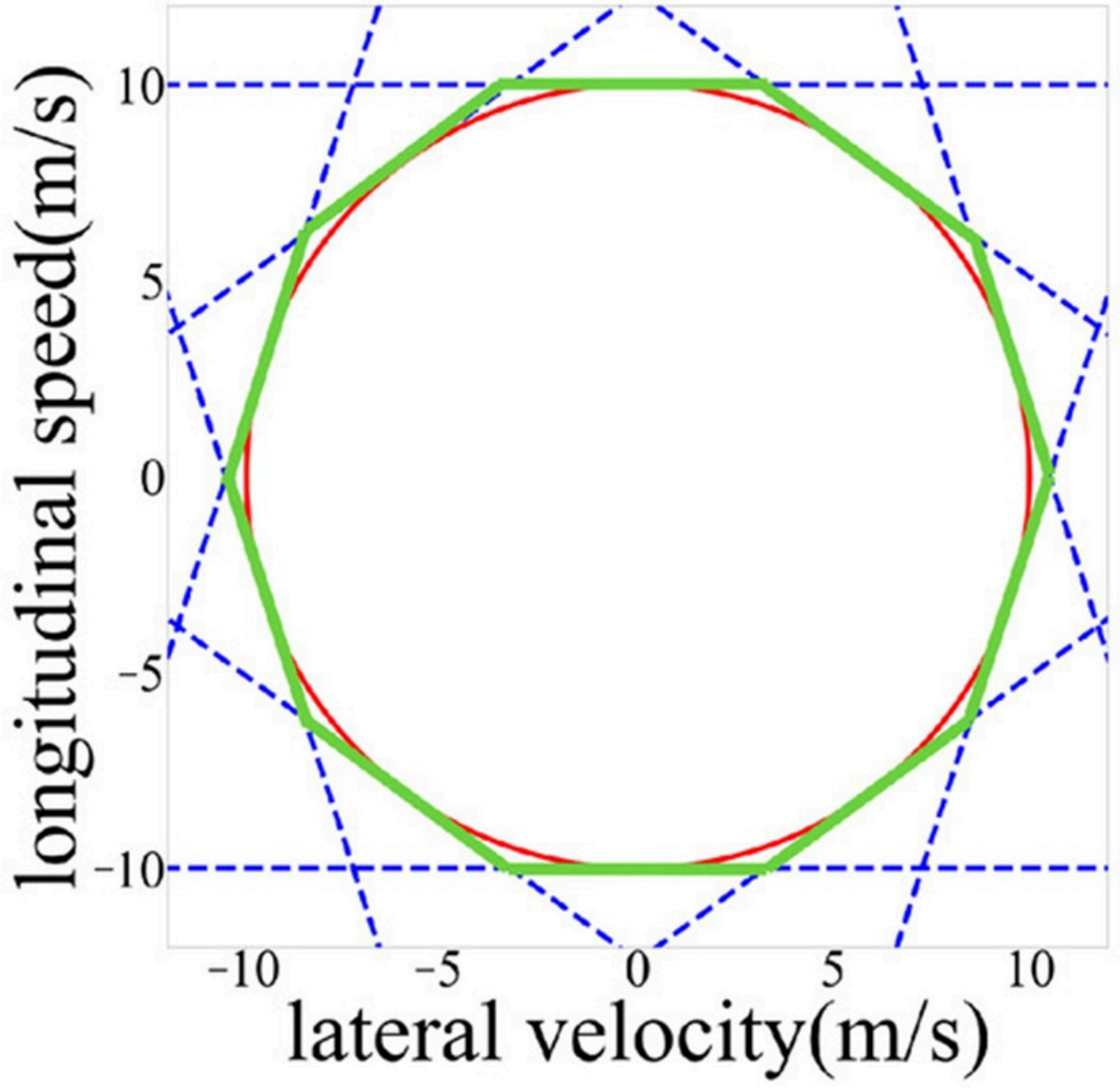


Figure 1. Maximum speed limit circle and linearization.

In this paper, the circular shape is approximated through a circumscribed regular polygon. Because the polygon circumscribes the inscribed circle, some zones inside the polygon still fall outside the inscribed circle. These zones do not respect the constraint. Consequently, it is necessary to push the number of polygon sides as high as possible to ensure the inscribed circle is entirely enveloped. Using the velocity constraint as a case in point, a regular M-gon is circumscribed around the maximum-speed-limit circle, as illustrated in **Figure 1**. The red line traces the circle circumscribed by the maximum-velocity vector in the Cartesian coordinate frame, the blue dotted line indicates the linear constraint, and the area bounded by the green solid line forms the final linearized constraint region. The inclination of side m is

© 2024 The Author(s). | Published by Agro Intelligence Research Press

$$k = \tan(2\pi m/M), \quad m \in [1,2,\ldots,M] \tag{4}$$

From this, the mathematical expression for each straight line can be obtained via the point-to-line distance formula and can be formulated as

$$v_x \sin\left(\frac{2\pi m}{M}\right) + v_y \cos\left(\frac{2\pi m}{M}\right) = v_{max} \tag{5}$$

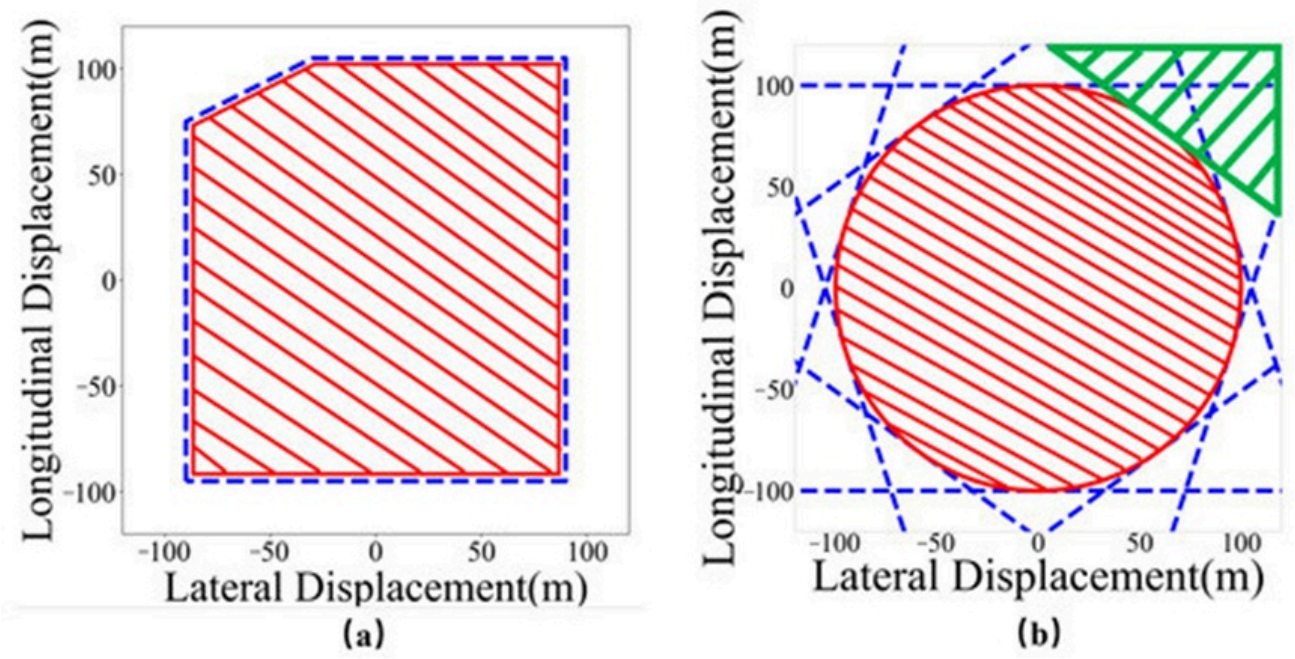
Accordingly, the ultimate restrictions on velocity can be captured as

$$\begin{gathered}\forall p \in [1,2,\ldots,N_d+N_v], \forall t \in [1,2,\ldots,T], \forall m \in [1,2,\ldots,M] \\ v_x \sin\left(\frac{2\pi m}{M}\right) + v_y \cos\left(\frac{2\pi m}{M}\right) \le v_{max}\end{gathered} \tag{6}$$

Velocity restrictions closely mirror acceleration restrictions, which this article does not treat separately.

Obstacle avoidance constraints

While UAVs and UGVs are in motion, they will inevitably encounter impassable terrain. For UAVs, these comprise circular no-fly areas, irregularly shaped woodlands, and so forth. For UGVs, they include orderly road borders, irregularly contoured lakes, and so on. To keep the number of constraints as small as possible and improve computational efficiency, irregular obstacle geometries are approximated as convex polygons or circles. As depicted in **Figure 2**, the red zones represent obstacles, the blue dotted lines indicate the linear constraints, and the green zone marks one of the admissible regions. In particular, obstacles are modeled as convex polygons in **Figure 2a**, which works well for obstacles with crisp boundaries and neat dividing edges. Obstacles are modeled as circles in **Figure 2b**, which works well for obstacles with complex shapes and blurry perimeters. In this work, circles are chosen to portray obstacles.



© 2024 The Author(s). | Published by Agro Intelligence Research Press

Figure 2. Polygonal and circular descriptions of complex obstacles: (a) polygonal description, and (b) circular description.

Taking as an illustration an obstacle whose center lies at (xno,) and whose radius equals Rno, the avoidance constraints are

$$\forall p \in [1,2,\ldots,N_d+N_v], \forall t \in [1,2,\ldots,T], \forall m \in [1,2,\ldots,M]$$
$$[x_p(t)-x_{no}]\sin\left(\frac{2\pi mW}{M}\right)+[y_p(t)-y_{no}]\cos\left(\frac{2\pi m}{M}\right) \geq R_{no}-(1-c_m^p)R_{inf} \quad (7)$$

$$\forall p \in [1,2,\ldots,N_d+N_v], \forall t \in [1,2,\ldots,T]$$
$$\sum_{m=1}^{M} c_m^p \geq 1 \quad (8)$$

Where $Rinf$ stands for a constant whose magnitude is far greater than R, and cpm is the binary decision variable tied to the obstacle avoidance constraint of the pth agent. Should the value of cpm equal 0, **Eq. 8** holds unconditionally and exerts no restrictive effect. Should the value of cpm equal 1, it signals that the constraint condition is met, and the agent's position is kept outside the constrained zone. **Eq. 9** prescribes that no fewer than one of the values of cpm must equal 1, with the resultant behavior shown in **Figure 2b**. So long as the agent resides beyond any edge of the polygon, the obstacle avoidance constraint holds.

Dynamic collision constraints

Since every UAV operates at a common altitude and UGVs likewise traverse the ground at a common altitude, dedicated constraints must be introduced to forestall individual collisions while moving. We illustrate collision avoidance among several drones as a representative case **(Figure 3)**.

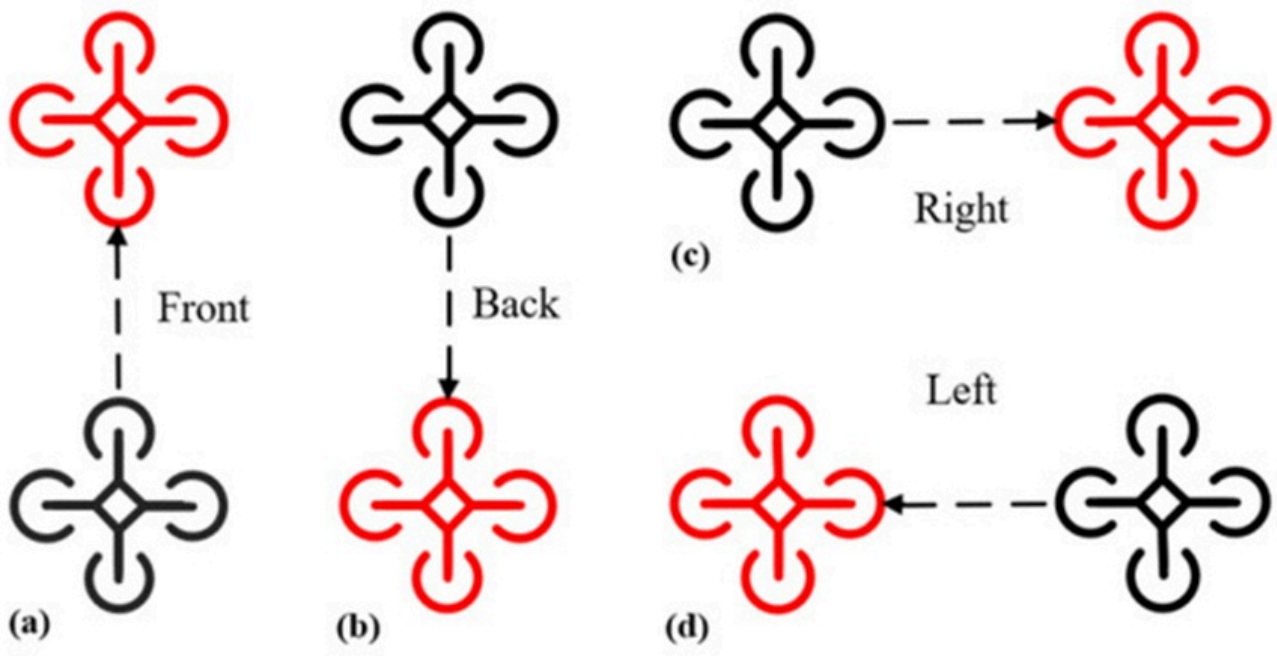


Figure 3. The four orientations of the agent in the anti-collision constraint: (a) front, (b) rear, (c) right, and (d) left.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

For a single UAV, four motion orientations exist, as shown in **Figure 3**: front, rear, right, and left. The dynamic anti-collision requirement for the agent can be formulated as keeping a safe separation threshold from any other UAV along every orientation.

$$\begin{aligned} &\forall p,q \in [1,2,\ldots,N_d+N_v], \forall t \in [1,2,\ldots,T] \\ &x_p(t) - x_q(t) \geq d_{safe} - (1 - C_1^{pq})R_{tnf} \\ &x_q(t) - x_p(t) \geq d_{safe} - (1 - C_2^{pq})R_{tnf} \\ &y_p(t) - y_q(t) \geq d_{safe} - (1 - C_3^{pq})R_{tnf} \\ &y_q(t) - x_p(t) \geq d_{safe} - (1 - C_4^{pq})R_{tnf} \\ &\sum_{dtr=1}^{4} C_{dtr}^{pq} = 1 \end{aligned} \tag{9}$$

Much like the obstacle avoidance restriction, the spacing between two individuals is regulated through the value of the binary decision variable $C_{\mathrm{dir}}^{\mathrm{pq}}$ in **Eq. 9**. When its value becomes 1, the collision avoidance restriction must be honored along that orientation. At that instant, the values of the other $Cpqdir$ are all 0, implying that the collision avoidance restrictions along the remaining three orientations need not be honored.

Task assignment constraints

UAVs and UGVs differ in their specialized capabilities and functions. UAVs, or unmanned aerial vehicles, are engineered for airborne operations and excel at tasks such as aerial scouting, observation, and data acquisition. They capitalize on their airborne mobility to gain access to hard-to-reach locations and to offer a sweeping vista from above.

In contrast, UGVs, or unmanned ground vehicles, are terrestrial platforms built to operate over land. Their functionalities routinely cover tasks such as ground-based monitoring, freight movement, and traversing various terrains. UGVs are well-suited to tasks that require ground-level engagement and movement.

Although UAVs and UGVs can share certain functionalities, such as observation, their dissimilar platforms yield distinct strengths. To illustrate, UAVs are superior at tasks that require overhead views and rapid repositioning, whereas UGVs are more capable at ground-level activities and crossing heterogeneous terrain. In light of that, the task assignment of UAVs and UGVs can be framed as a multi-target detection task assignment challenge for heterogeneous agents.

UAVs benefit from a high vantage point and can inspect ground features from above, making them better suited to probing broad, flat expanses. UGVs can scrutinize ground targets at close range, which makes them better suited to conducting methodical searches of designated zones. Yet, drones are constrained by prohibited airspace and cannot reach every location. Urban flight also places bounds on the lowest and highest allowable altitudes owing to safety imperatives. Hence, in certain situations, drones cannot fulfill all detection tasks. In parallel, UGVs face difficulties, such as crossing lakes and dense forests.

Thus, to ensure assigned agents complete tasks in the proper sequence, it is vital to consider which agent best fits the geographical setting and whether multiple agents are needed during the allocation phase. The task assignment of the air–ground coordination problem can be sorted into four types. As depicted in **Figure 4a**, areas such as lakes, rivers, and dense forests are difficult for UGVs to cross, so it is more appropriate to detect target points with UAVs. As depicted in **Figure 4b**, drones cannot legally enter military exclusion zones and urban no-fly areas, so it is more appropriate to detect those target points with UGVs. As depicted in **Figure 4c**, in settings such as open meadows and thoroughfares offering

© 2024 The Author(s). | Published by Agro Intelligence Research Press

unblocked views, target points can be detected by both UAVs and UGVs. As depicted in **Figure 4d**, in settings such as thin tree cover and tightly packed buildings with intricate surroundings, target points cannot be reliably detected by either UAVs or UGVs by themselves.

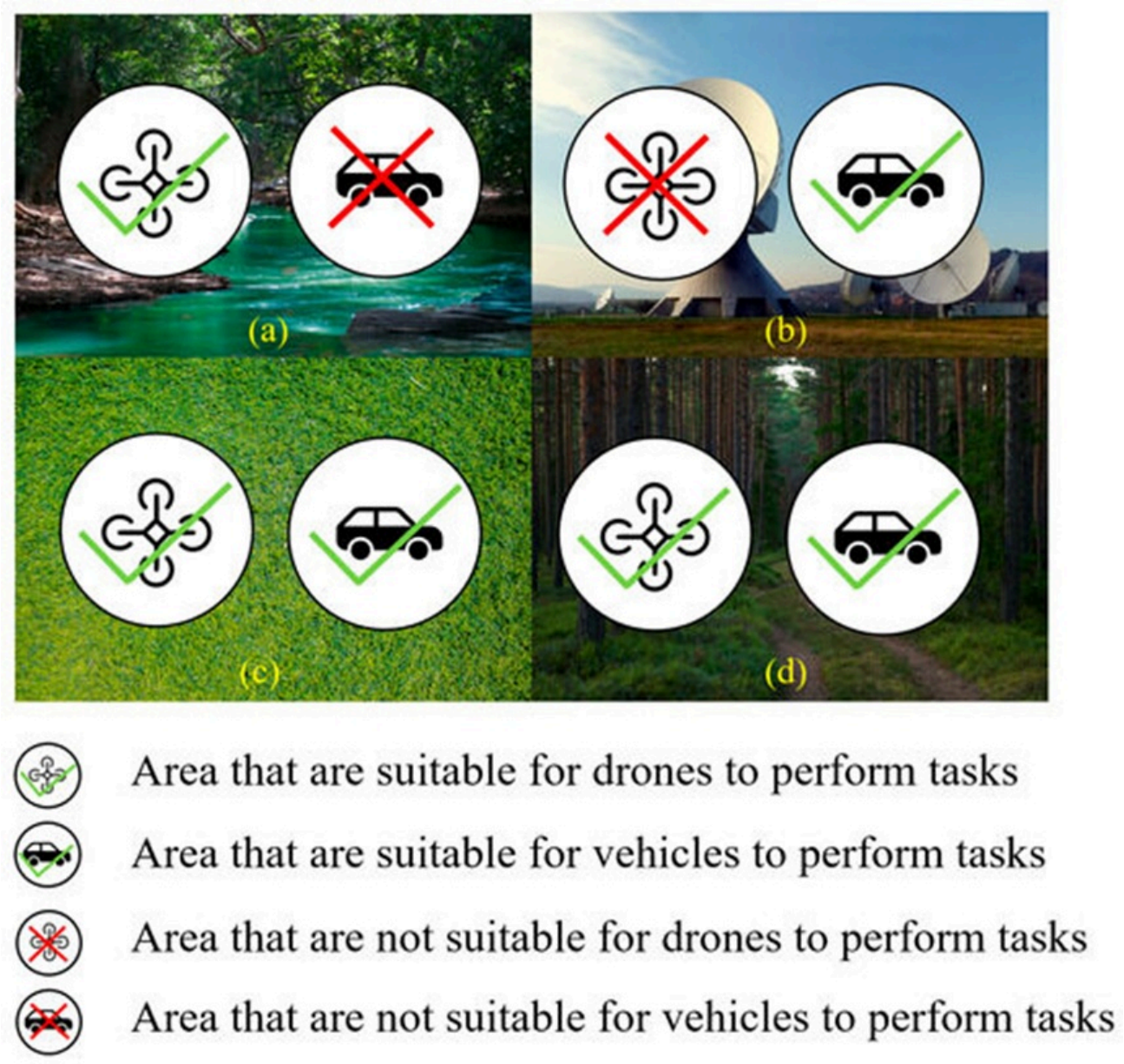


Figure 4. Four different example scenarios: (a) scene suitable for UAVs; (b) scene suitable for UGVs; (c) scene suitable for both UGVs and UAVs; (d) scene that needs to be detected by both UGVs and UAVs.

As a result, the task assignment of the air–ground coordination problem can be sorted into four cases:

Case 1: target points suitable for UAVs.

Case 2: target points suitable for UGVs.

Case 3: target points suitable for both UGVs and UAVs.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Case 4: target points that need to be detected by both UGVs and UAVs.

In this work, the binary decision variable $\mathrm{T}_{\mathrm{p}}^{\mathrm{i}}(t)$ serves to signal whether the pth agent is required to arrive at the target point at time step t. If arrival is required, the value of $\mathrm{T}_{\mathrm{p}}^{\mathrm{i}}(t)$ becomes 1. If not, the value stays at 0. The full equation can be set out as

$$\forall i \in [1,2,\ldots,N_{t1}] \quad \sum_{p=1}^{N_d}\sum_{t=1}^{T} T_p^t(t) = 1 \tag{10}$$

$$\forall i \in [1,2,\ldots,N_{t2}] \quad \sum_{p=1}^{N_v}\sum_{t=1}^{T} T_p^t(t) = 1 \tag{11}$$

$$\forall i \in [1,2,\ldots,N_{t3}] \quad \sum_{p=1}^{N_v+N_d}\sum_{t=1}^{T} T_p^t(t) = 1 \tag{12}$$

$$\forall i \in [1,2,\ldots,N_{t4}] \quad \begin{cases} \sum_{p=1}^{N_d}\sum_{t=1}^{T} T_p^t(t) = 1 \\ \sum_{p=1}^{N_v}\sum_{t=1}^{T} T_p^t(t) = 1 \end{cases} \tag{13}$$

Where **Eqs. 10-13** capture Case 1, Case 2, Case 3, and Case 4, respectively.

Task completion constraints

In the task allocation restrictions, each task location is assigned to an inspection by a matching UAV or UGV, with the arrival time (number of time steps) precisely specified. It is therefore essential to formulate the task-completion restriction to ensure that the UAV or UGV arrives at the target location (or draws sufficiently close to it) within the specified number of time steps. As depicted in **Figure 5**, so long as the agent reaches target location i having coordinates (x_target^i,y_target^i), within a square whose side measures $2dpass$ at the designated time step t, the agent may be deemed to have fulfilled the task.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

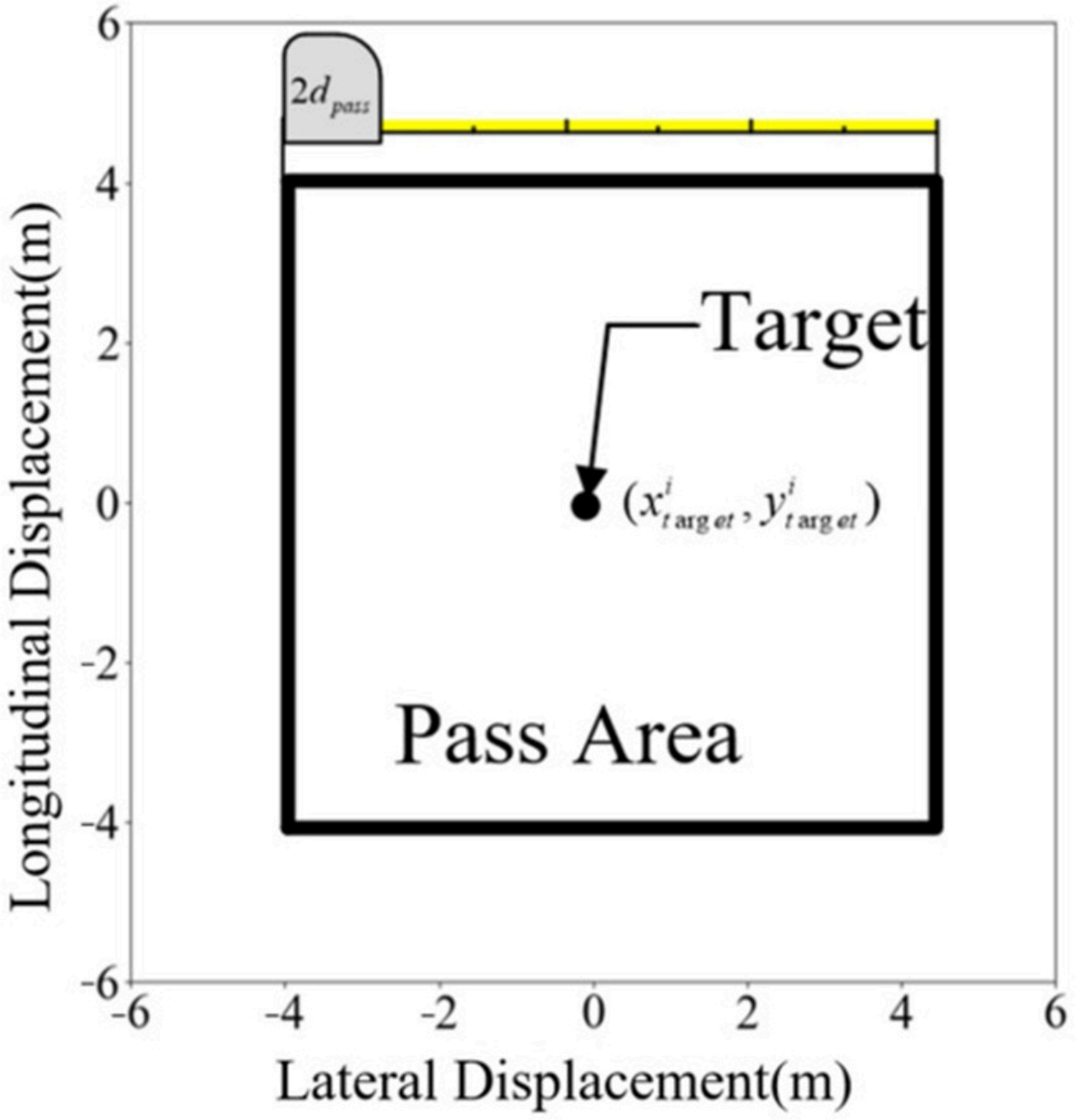


Figure 5. Schematic diagram of the task completion constraints of the target point.

The mathematical description of task completion restrictions can be rendered as:

© 2024 The Author(s). | Published by Agro Intelligence Research Press

$$\begin{aligned}
&\forall p \in [1,2,\ldots,N_d+N_v], \forall i \in [1,2,\ldots,N_t], \forall t \in [1,2,\ldots,T] \\
&x_p(t) - x^i_{target} \le d_{pass} + (1 - T^i_p(t))R_{inf} \\
&x^i_{target} - x_p(t) \le d_{pass} + (1 - T^i_p(t))R_{inf} \\
&y_p(t) - y^i_{target} \le d_{pass} + (1 - T^i_p(t))R_{inf} \\
&y^i_{target} - y_p(t) \le d_{pass} + (1 - T^i_p(t))R_{inf}
\end{aligned} \tag{14}$$

Objective function

The objective function of the air–ground cooperative multi-target detection task formulation established in this paper comprises four components: the shortest time, the overall time, the overall energy outlay, and the trajectory smoothness.

The shortest time

The shortest time captures the entire duration elapsed up to the instant the final target location is inspected, and its mathematical description can be given as

$$\begin{aligned}
&\forall p \in [1,2,\ldots,N_d+N_v], \forall i \in [1,2,\ldots,N_t], \forall t \in [1,2,\ldots,T] \\
&t_{max} \ge t \cdot T^i_p(t) \\
&J_1 = \min t_{max}
\end{aligned} \tag{15}$$

Where $t$max stands for the greatest global time, and $J1$ constitutes the optimization function. Through contrasting the restriction against the minimum objective function, the minimization time $t$max is cast in a linearized manner.

The total time

The overall time matches the aggregate of the time expended by each UAV and UGV to conclude the task, which can be given as:

$$\begin{aligned}
&\forall i \in [1,2,\ldots,N_t], \forall t \in [1,2,\ldots,T] \\
&t^p_{max} \ge t \cdot T^i_p(t) \\
&J_2 = \min \sum_{p=1}^{N_d+N_v} t^p_{max}
\end{aligned} \tag{16}$$

Where $\mathrm{t}^{\mathrm{p}}_{\mathrm{max}}$ signifies the time required by the pth agent to wrap up all assigned tasks. $J2$ constitutes the optimization function.

The total energy consumption

Under steady-speed conditions, the energy outlay of the UGV is tied to its duration of operation, which can be obtained by the following line of reasoning set out in [26] for the UAV.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Presume that the airstream velocity generated by one propeller of the UAV equals v, and the air mass equals $mair$. It can be deduced that the kinetic energy contributed by the propeller, discounting air friction, equals:

$$E = \frac{1}{2} m_{atr} v^2 \quad (17)$$

The air mass $mair$ can be expressed as:

$$m_{atr} = \pi r^v v t \rho \quad (18)$$

Where ϱ denotes the ambient air density at the current altitude and r denotes the radius of the propeller. Hence, for a six-rotor UAV, the power draw P comes to:

$$P = 6\frac{dE}{dt} = 3\pi^2 v^3 \rho \quad (19)$$

Eq. 19 can be alternatively restated as:

$$P = Fv = mgv \quad (20)$$

Where m denotes the mass of the UAV and g denotes the local acceleration due to gravity. Joining **Eqs. 19 and 20**, one obtains:

$$P = \sqrt{\frac{(mg)^3}{2\pi r^2 \rho}} \quad (21)$$

Thus, for UAVs, the energy expenditure during hovering scales linearly with time. Likewise, during uniform movement, **Eq. 20** can be restated as:

$$P = F_1 v = \frac{mgv}{\cos\theta} \quad (22)$$

Where θ denotes the inclination between the drone's main axis and the vertical orientation. It can be seen that during uniform movement, the energy outlay is solely and positively linked to time.

Consequently, at steady speed, the overall energy consumption target $J3$ can be expressed as

© 2024 The Author(s). | Published by Agro Intelligence Research Press

$$J_3 = \min \sum_{p=1}^{N_d+N_v} \sum_{t=1}^{T-1} \alpha \left[\frac{x_p(t+1) - x_p(t)}{v_{px}(t)} + \frac{y_p(t+1) - y_p(t)}{v_{py}(t)} \right] \quad (23)$$

Where α serves as a proportionality factor that handles the dimensional conversion linking energy outlay and time, and may also be utilized to tune the relative weight of total energy consumption; (t) and $vp(t)$ stand for the horizontal and vertical speed components of the pth agent at time step t, respectively.

The smoothness of the trajectory

To ease the tracking of trajectory waypoints by UAVs and UGVs in practical circumstances, the waypoints generated by the designed path-planning algorithm should be as free of abrupt changes as possible. In this work, the constraint for trajectory smoothness is constructed as

$$J_4 = \min \sum_{p=1}^{N_d+N_v} \sum_{t=1}^{T-1} \beta \left[\frac{a_{px}(t+1) - a_{px}(t) + a_{py}(t+1) - a_{py}(t)}{\Delta t} \right] \quad (24)$$

Where β serves as a proportionality factor that handles the dimensional conversion linking energy outlay and time, and may also be utilized to modulate the penalty arising from trajectory unevenness inside the objective function. (t) and $a(t)$ stand for the lateral and vertical components of acceleration of the pth agent at time step t, respectively.

Optimal Solution Method of the Mathematical Model

Following the formulation of the mathematical model, a matching solution strategy must be engineered to resolve the model and deliver a result with suitable accuracy. The model constructed in this paper can be tackled either through established optimization approaches or through heuristic procedures; however, the base algorithm must still be suitably adapted to ensure convergence and to accelerate computation to the greatest extent possible.

Branch-and-bound

The branch-and-bound (B&B) method was originally developed by Land and Doig [27] in the 1960s to address integer programming problems and later broadened by Ref. [28] to handle MILP problems. The essential idea behind this method consists of branching, bounding, and pruning.

In particular, imagine a Mixed Integer Linear Programming problem A for which the minimum solution is sought, and imagine its matching linear programming problem B. Setting out from the solution of problem B, should the optimal solution fail to uphold the integer condition of A, the optimal objective function value z must then act as the lower bound of the optimal objective function value z∗ of A, written as z. In addition, the objective function value for any viable solution of A will function as the upper bound of z∗, written as $\bar{z}$. The Branch-and-Bound method splits the feasible territory into multiple sub-territories; this action is called branching. Subsequently, each sub-territory yields a fresh upper bound $\bar{z}$new and a fresh lower bound znew to function as the territory's limits; this is called bounding. If the fresh lower bound znew surpasses the currently best-known solution znow∗, it signals that the territory has been exhaustively scanned and no additional branching is called for; its subsets can be omitted, an action called pruning. To summarize, the Branch-and-Bound method needs to steadily shrink $\bar{z}$ and lift z throughout the iterative cycle until z∗ is ultimately uncovered. The B&B algorithmic scaffold is presented in **Algorithm 1**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Input: A mixed integer programming problem A, the upper bounds of the current solution $\bar{z}$, the lower bounds of the current solution $\underline{z}$, B is the corresponding linear programming problem of A, B' is a branch of B

Output: The optimal solution z^*

```
while solution of B is not empty do
  if B' is feasible then
      Let z be optimal solution of B'
      if z > z̄ then
          Cut the branch B'
      end if
      if z satisfies integral solution of A then
          if z < z̄ then
              z̄ = z
          end if
      else
          z = z
      end if
  end if
end while
return z̄
```

Algorithm 1. Branch-and-Branch

Improved genetic algorithm and a algorithm

The Genetic Algorithm yields faster solution speed in task allocation and path planning, yet in this paper, task planning and path planning are tightly coupled, which hinders conventional Genetic Algorithms from effectively resolving this class of problem through the mere design of elementary chromosomes and genetic operators. On that account, in this paper, the Genetic Algorithm is used for task distribution, the A algorithm is used to secure the path-planning outcomes, and the objective function value computed for the entirety serves as the Genetic Algorithm's fitness score. The concrete workflow is laid out in **Algorithm 2**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

```
Initialize the population
Coding chromosome
while the number of iterations is not reached do
    Calculate fitness
    Select operation
    Crossover operation
    if chromosomes need to be adjusted then
        Adjust operation
    end if
    Mutate operation
    Generate a new generation
end while
```

Algorithm 2. Genetic Algorithm Process

The population size $Nper$, the selection probability, the crossover probability, the mutation probability, and the maximum number of evolutionary generations need to be fixed before initializing the population.

For chromosome encoding, each chromosome is treated as a two-tier code. Within this scheme, the first tier of the chromosome code specifies the execution order across all target locations, and the second tier specifies the ID of the agent called upon to detect the associated target location. Should the overall count of target locations come to Nt, and the count of target locations that call for detection by both a UAV and a UGV come to $Nt4$, then the chromosome's length equals $Nt+Nt4$. By way of illustration, if three UAVs and two UGVs exist—amounting to a total of 5 target locations, among which 2 target locations require detection by both UAVs and UGVs and 3 target locations require detection by UAVs or UGVs—then the encoded chromosome can be depicted as in **Figure 6**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

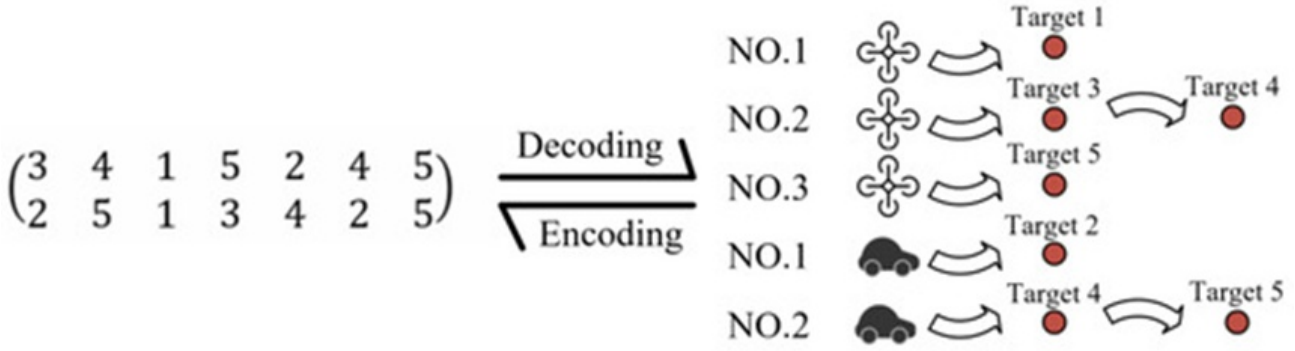


Figure 6. Description after chromosome encoding and meaning after decoding.

From **Figure 6**, one can gather that UAV No. 1 handles the detection job at target location No. 1, UAV No. 2 successively handles the detection jobs at target locations No. 3 and No. 4, UAV No. 3 handles the detection job at target location No. 5, UGV No. 1 handles the detection job at target location No. 2, and UGV No. 2 successively handles the detection jobs at target locations No. 4 and No. 5. Chromosomes encoded following this scheme can satisfy the restrictions governing the assignment of task locations.

At this stage, drawing on the target locations that must be inspected by the assigned agent once the individual $iper$ has been decoded, the route for each agent is derived via the A algorithm. The solution yielded by evaluating the previously described objective function then becomes the initial fitness value, ($iper$). To streamline the subsequent selection of population members, the ultimate fitness measure, $Fi(iper)$, is formulated as:

$$Fitness(i_{per}) = \frac{1}{J(i_{per})} \quad (25)$$

A greater fitness magnitude signifies a shorter path and, as a direct result, lower resource usage. A higher fitness value, therefore, indicates more efficient resource use.

An elitist approach governs the selection step, giving individuals who demonstrate superior fitness a stronger likelihood of persisting through the genetic cycles. Accordingly, the likelihood of an individual being chosen is defined as:

$$p(i_{per}) = Fitness(i_{per}) / \sum_{i_{per}=1}^{N_{per}} Fitness(i_{per}) \quad (26)$$

Following the crossover likelihood, a pair of chromosomes is drawn from the population. A crossover point is randomly selected along the chromosome, and the genetic material is exchanged at that location. To illustrate, should two individuals be chosen to undergo crossover at the fifth gene, the resultant effect is depicted in **Figure 7**. A green line splits the chromosome into two sections, and the parent and mother individuals swap the genetic material on the right side of this dividing line.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

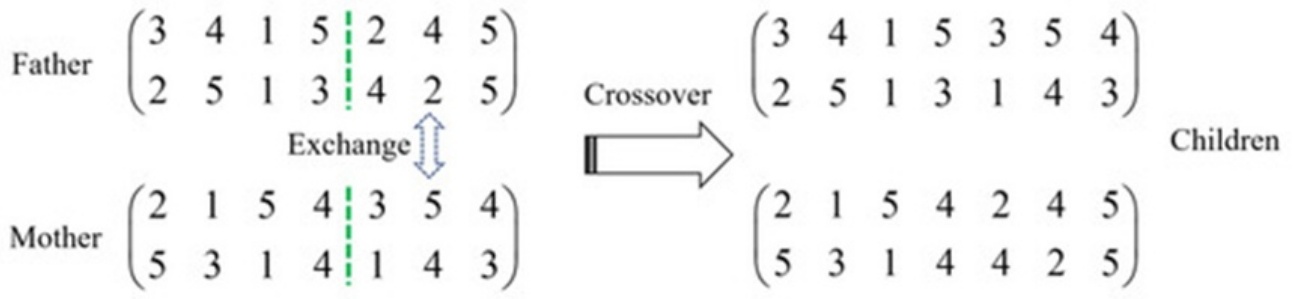


Figure 7. Schematic diagram of chromosome crossover operation.

If the genetic information at the chosen crossover point differs between the two parent individuals, the resulting offspring will concurrently suffer from both omitted and duplicated task locations. The count of omitted locations will perfectly match the count of duplicated ones. To remedy this, a single corrective step is required: replace the superfluous, repeated task points with the absent ones. Following this correction, a verification step confirms whether the agent-to-task allocation in the offspring adheres to the specified requirements. The corrected individual is presented in **Figure 8**, and the two resulting offspring now satisfy the coding criteria and are eligible to continue in the evolutionary process.

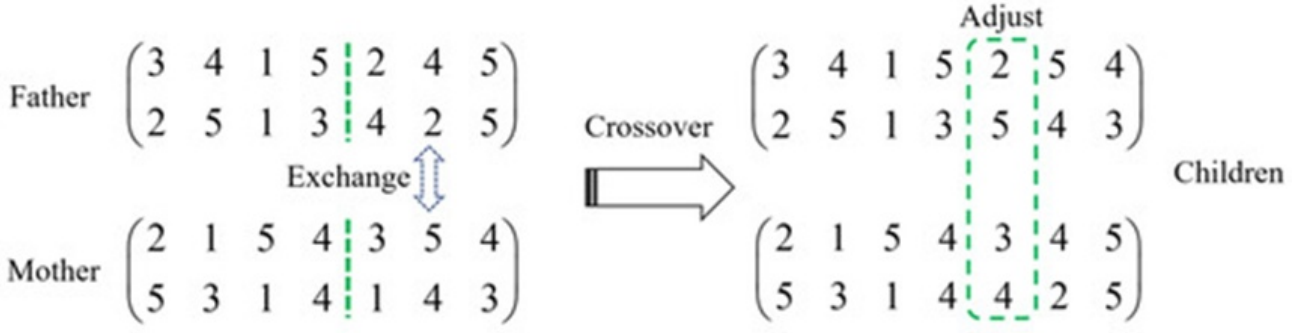


Figure 8. Schematic diagram of chromosome crossover-adjustment operation.

Based on the mutation likelihood, an individual is randomly flagged for mutation within the population. Two distinct mutation sites are then randomly chosen along that individual's chromosome, and the gene codes occupying these two positions are swapped. The practical execution of this operation is illustrated in **Figure 9**, where the genes enclosed in the green and blue boxes are shown to conform to the model's rules following a valid mutation.

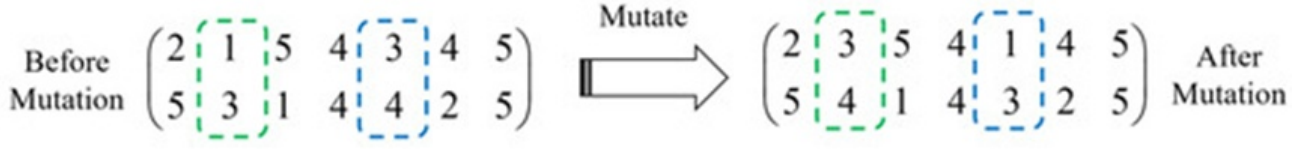


Figure 9. Schematic diagram of chromosome mutation operation.

The chosen parent chromosomes are combined with offspring chromosomes produced through crossover and mutation to assemble the next generation of the population.

The fitness scores for the newly formed population are then computed, and an evaluation is performed to determine whether the current best solution meets the termination criteria for the current iteration. Should this not be the case, the procedure returns to step 4 to choose a new generation and launch another cycle of evolutionary refinement. The Genetic Algorithm terminates once the pre-defined maximum number of iterations is reached.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Branch-and-bound with improved genetic algorithm (IGA-B&B)

Hot-start initialization for the B&B technique is a common practice for solving small-scale MILP formulations. This approach depends on an initial set of viable solutions to establish tighter bounds. In our work, this initial set of viable solutions is provided by our proposed Improved Genetic Algorithm, which serves to bound the problem's upper bound, thereby boosting the pruning process's effectiveness and overall computational throughput.

In a concrete sense, the execution starts by putting aside the kinematic specifics of the UAVs and UGVs, thereby simplifying the original problem into a basic single-agent route-finding problem. An Improved Genetic Algorithm is then run to pinpoint an initial solution. Given the certainty that this initial solution cannot surpass the true optimal solution in quality, it is perfectly suited to act as an upper bound for the B&B algorithm.

Moreover, by launching the B&B algorithm with this newly established upper bound, the search domain is intelligently segmented, dramatically accelerating search velocity and the algorithm's overall performance. The logical flow of this specific combined algorithm is detailed in **Algorithm 3**.

1: Employ Improved Genetic Algorithm to find initial solution z_{iga}
2: Set z_{iga} as the upper bound for B
3: Initialize Branch-and-Bound algorithm with the upper bound z_{iga}
4: while solution of B is not empty do
5: Select a branch B' from B
6: **if** B' is feasible **then**
7: Let z' be optimal solution of B'
8: **if** $z' > z_{iga}$ **then**
9: cut the branch B'
10: **else**
11: Update $z_{iga} \leftarrow z'$
12: **end if**
13: **end if**

14: **if** z_{iga} satisfies integral solution of A **then**
15: $z^* \leftarrow z_{iga}$
16: **Break**
17: **end if**
18: **end while**
19: **return** z^*

Algorithm 3. Branch-and-Branch combined with Improved Genetic Algorithm

© 2024 The Author(s). | Published by Agro Intelligence Research Press

Simulation Comparison and Experimental Verification

To sharpen the comparative analysis and reduce the influence of environmental inconsistencies on the model, our simulation tests use a grid map with a constant environmental layout, with motion limits defined by maximum speed and acceleration. Agent movement is restricted to 8 possible directions. The simulation map includes three UAVs and two UGVs, eight task locations in total, and features two restricted airspace zones for UAVs and two impassable zones for UGVs.

While the Branch-and-Bound algorithm is theoretically capable of finding a global optimum, its prohibitively long runtime makes it impractical. The trend line in **Figure 10** reveals that after a 34-minute mark, the objective function's value becomes virtually stagnant, and the trajectory produced undergoes negligible further refinement. For this reason, a 34-minute run time cap is enforced in all subsequent simulation experiments.

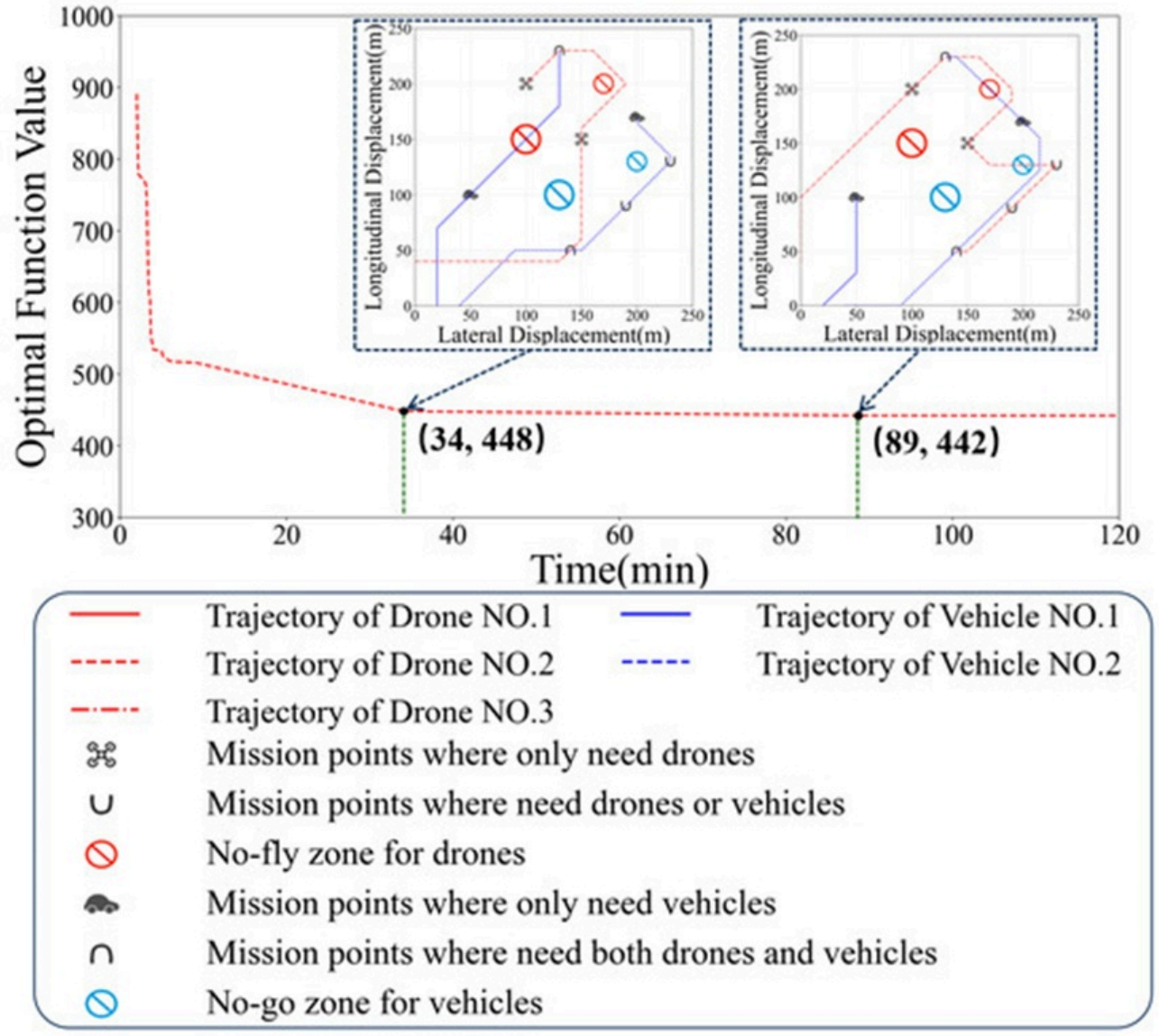


Figure 10. The effect of iteration time on the results of the Branch-and-Bound method.

In a model variant where the objective function only accounts for the shortest completion time, the lack of any constraint on travel distance gives rise to two problematic behaviors: the first is that an agent with a mission continues moving

© 2024 The Author(s). | Published by Agro Intelligence Research Press

aimlessly after its task is finished; the second is that an agent with no assigned mission behaves erratically, as its actions are neither directed nor penalized by the objective function, creating chaotic trajectory patterns. This is visible in **Figure 11a**. These shortcomings are successfully resolved by introducing energy consumption terms, as shown in **Figure 11b**.

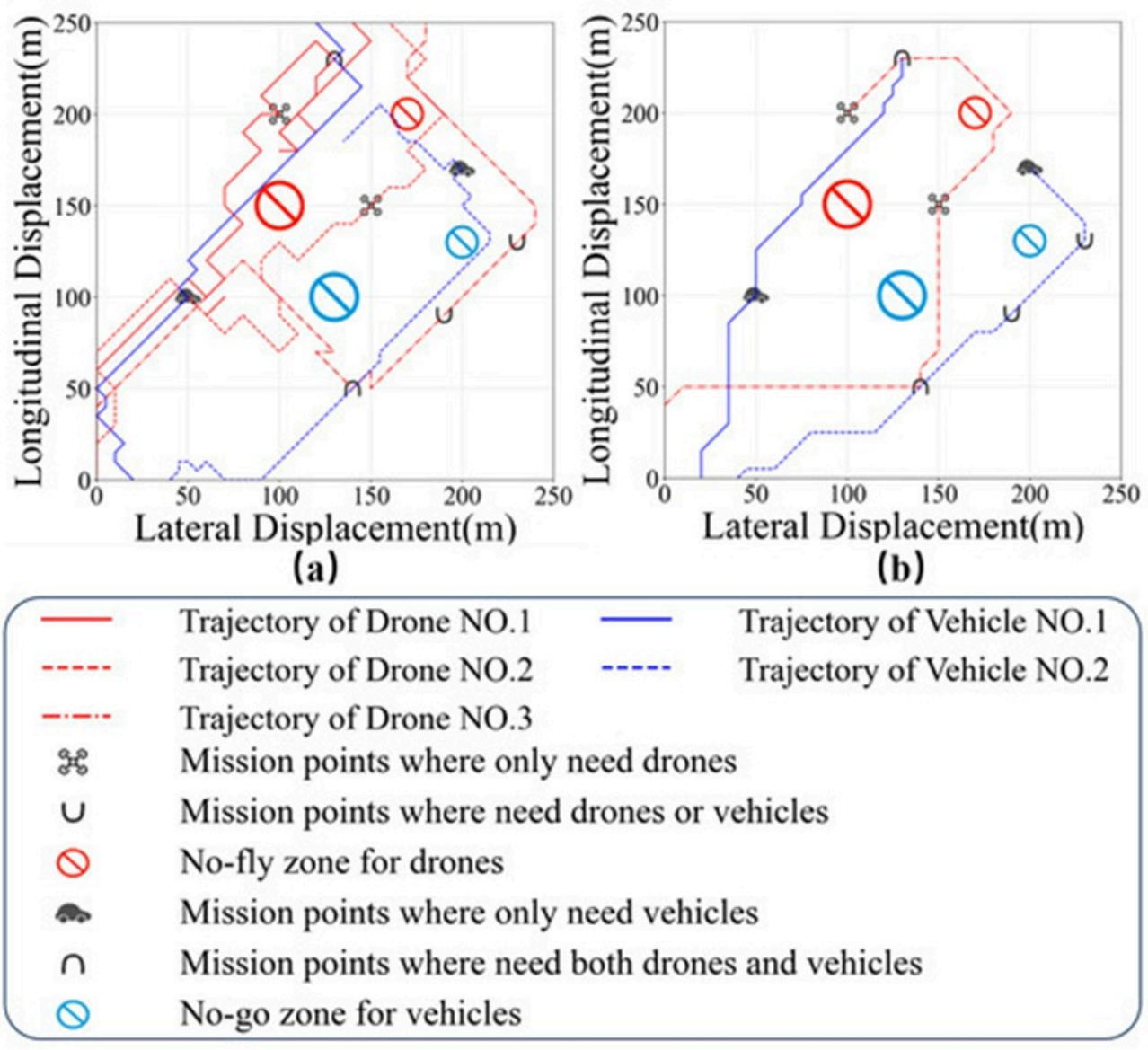


Figure 11. The results of optimization with or without energy consumption constraints: (a) with only a time constraint, and (b) with time and energy consumption constraints.

When a trajectory smoothness term is incorporated into the objective function, a significant improvement is observed over the model that includes only time and energy terms. The frequent, sharp directional shifts by each agent are notably reduced, which, in turn, eases the control challenge for path following and eliminates wasteful energy expenditure, as shown in **Figure 12**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

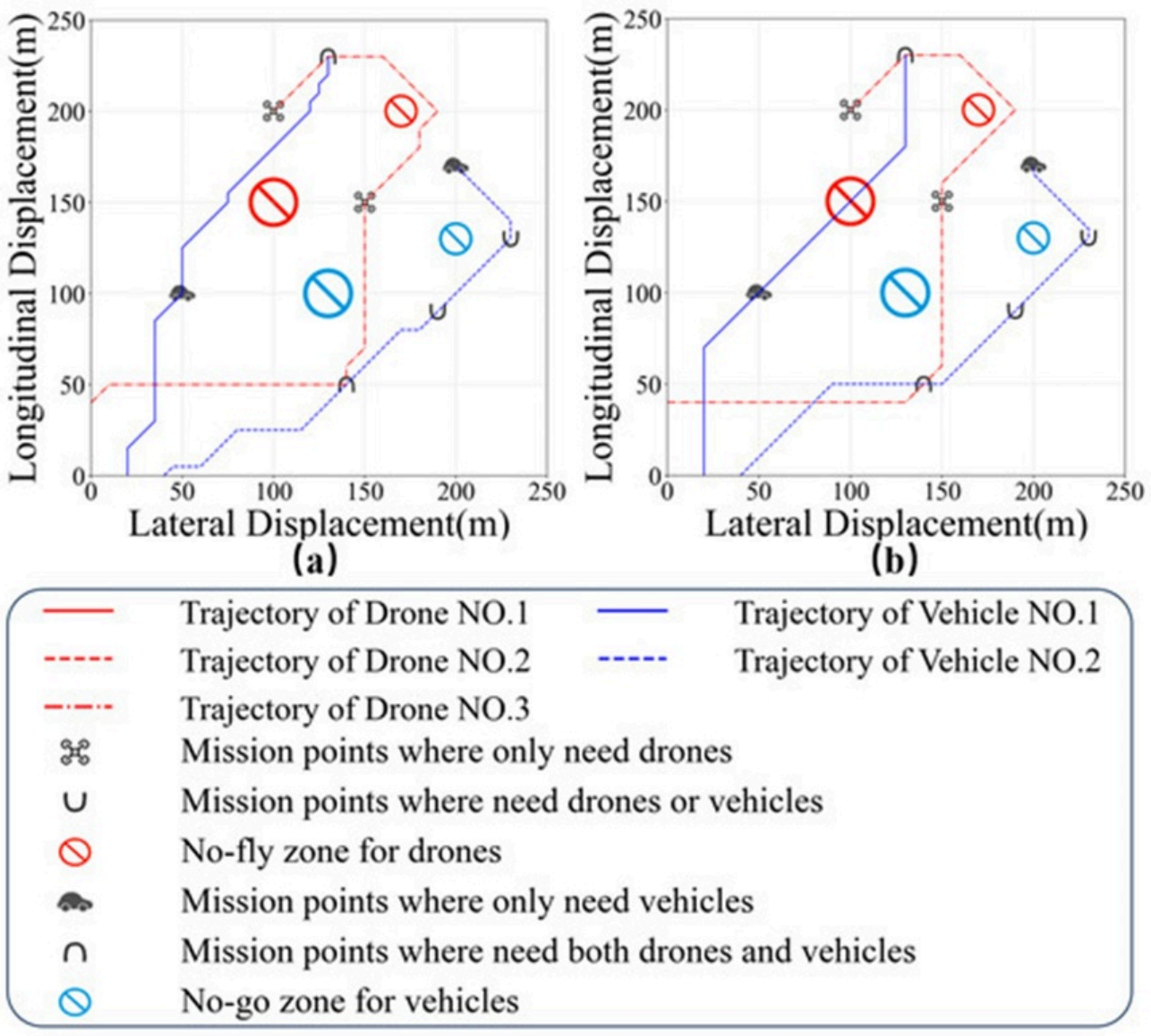


Figure 12. Optimization results with and without trajectory smoothness: (a) with time and energy consumption constraints, and (b) with optimized time, energy consumption, and trajectory smoothness constraints.

To prove the superiority of the IGA-B&B algorithm we propose, we conducted a comparative analysis of simulation outcomes from various objective-function-solving methods, with results presented in **Figure 13**. From **Figure 13a**, it is clear that while the stand-alone Genetic Algorithm operates at high speed, the lack of dynamic constraint integration from the A algorithm results in a relatively jagged final path and a suboptimal final objective function value. The GA method achieves a final value of 633, showing minimal improvement at the cost of additional computation time. The traditional B&B method, while capable of reliably finding the optimal solution, suffers from a slow, exhaustive search process and high memory demands, often leading to sluggish performance and potential memory overflow, as evidenced in **Figure 13b**. The IGA-B&B method we propose successfully synthesizes the strengths of both the Genetic Algorithm and the B&B method. This synergy results in a 30% effective reduction in computation time while still producing a highly favorable allocation result, as shown in **Figure 13c**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

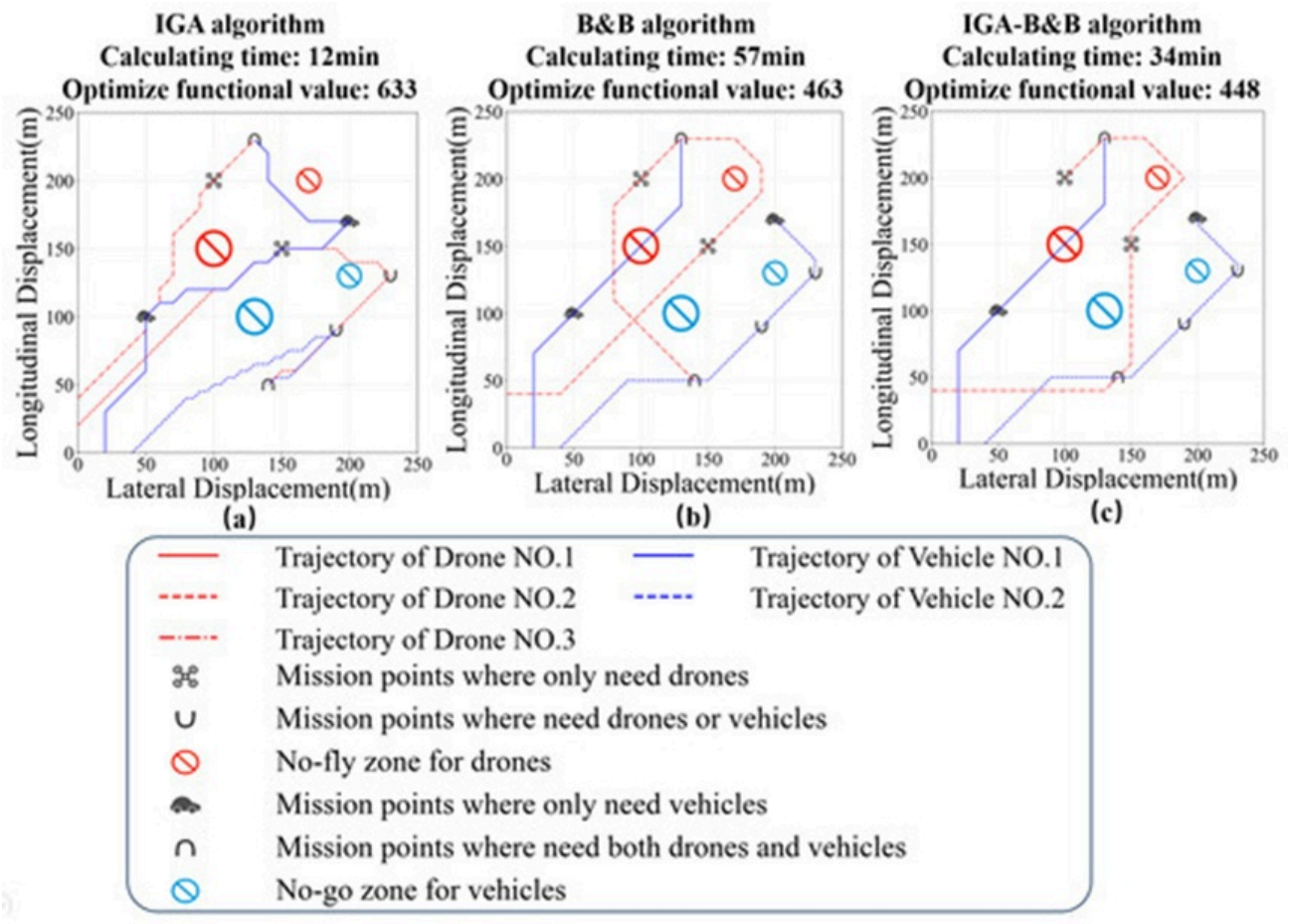


Figure 13. Comparison of several optimization methods: (a) Genetic Algorithm, (b) B&B, and (c) IGA-B&B.

To confirm the practical viability of the proposed model, outdoor field experiments were conducted. The testing apparatus utilized for these trials is displayed in **Figure 14**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press



Figure 14. Equipment used during the experimental test: (a) UGV, and (b) UAV.

Our experimental setup, situated in a 250 m by 250 m square zone on the Southeast University campus, comprises one UAV and one UGV, four mission points in total, along with one simulated no-fly zone for the UAV and one simulated impassable zone for the UGV, as illustrated in **Figure 15**.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

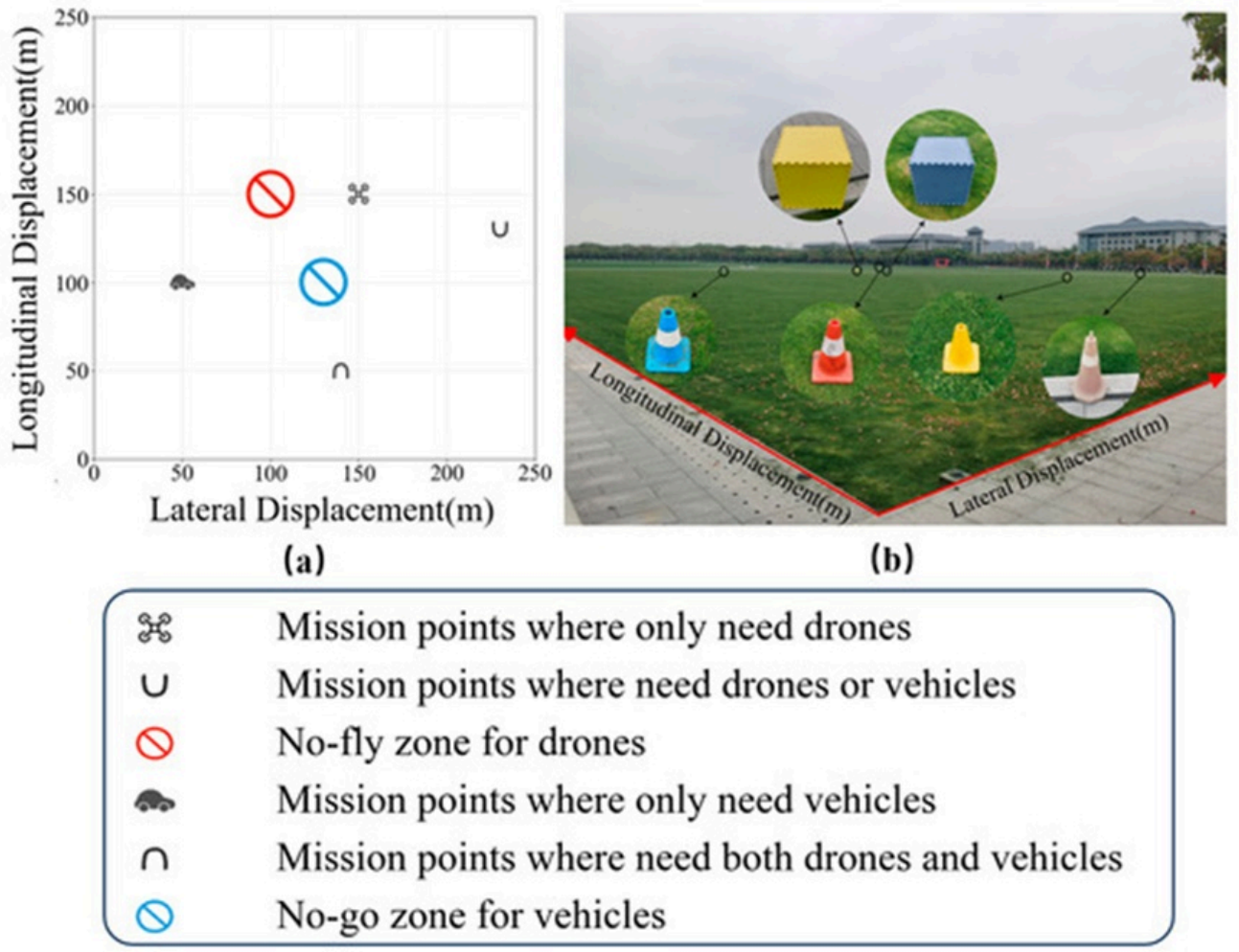


Figure 15. The experimental scene: (a) schematic diagram, and (b) side view of the experimental scene.

By applying our proposed method, the detection tasks at the target points are assigned to the UAV and the UGV, and their optimal paths are generated simultaneously. Guided by their individual path-tracking controllers, the UAV and UGV navigate along these pre-planned routes. They demonstrate a strong ability to precisely track the planned trajectories, successfully visiting all path waypoints within a 3-minute window. Their recorded flight and ground paths are shown in **Figures 16a and 16b**, respectively. The results clearly show that both the UAV and UGV effectively visit their designated target points while reliably circumventing their respective restricted areas, offering further proof of the proposed method's effectiveness and real-world applicability.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

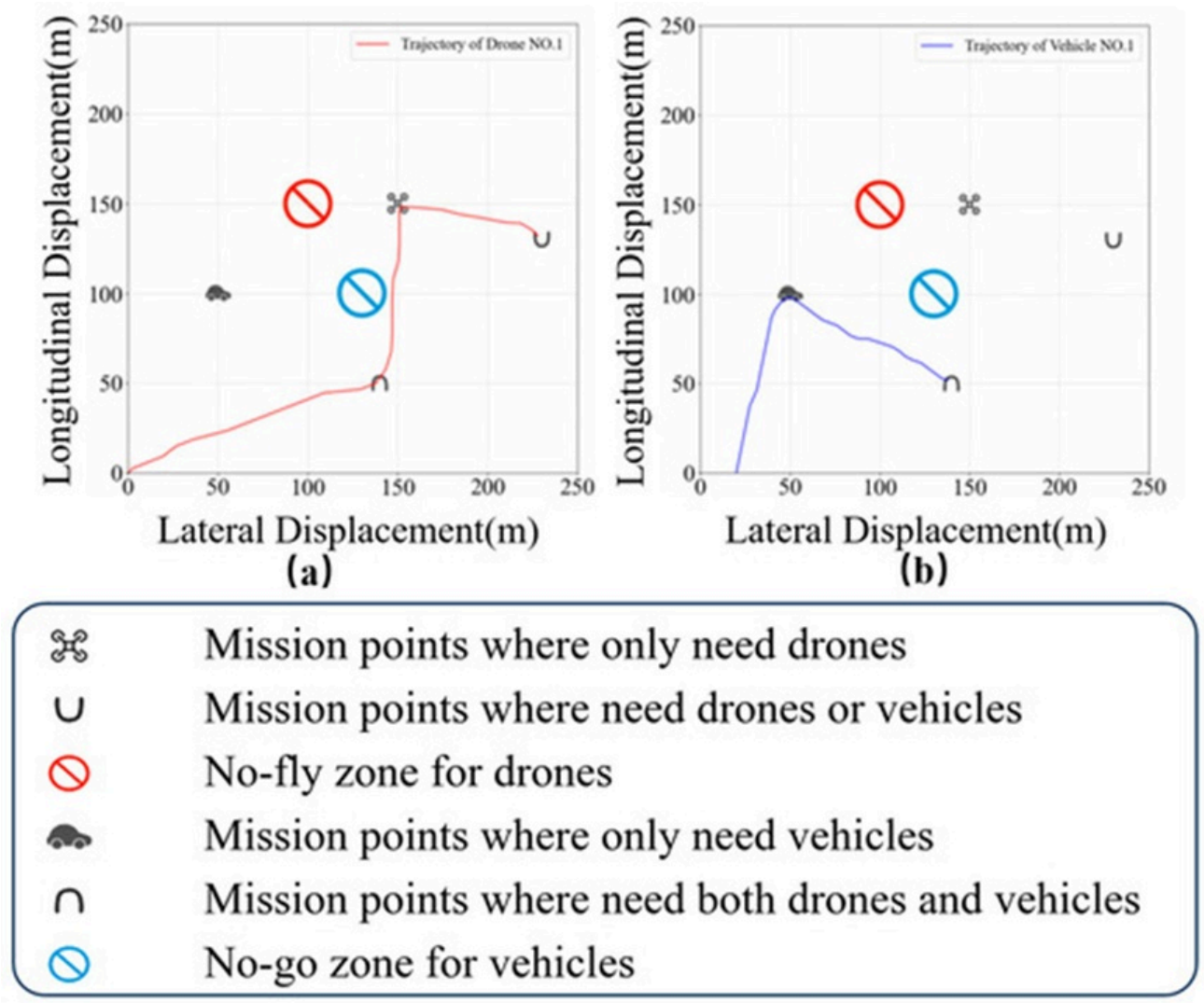


Figure 16. Trajectories of UAVs and UGVs: (a) trajectory of UAVs, and (b) trajectory of UGVs.

Conclusion

This work introduces a model for air–ground collaborative multi-target detection tasks, formulated on the principles of Mixed Integer Linear Programming (MILP). The model integrates key operational constraints, including the kinematic properties of UAVs and UGVs, dynamic collision-avoidance rules, task-allocation protocols, and obstacle-circumvention requirements. A composite objective function is also constructed to guide optimization toward minimal time and energy consumption, and maximal path smoothness. To effectively resolve this function, a hybrid solution strategy, termed Branch-and-Bound with Improved Genetic Algorithm (IGA-B&B), is proposed, successfully deriving the optimal task distribution and flight/ground paths for the collaborative system. A comprehensive simulation environment featuring multiple agents, diverse obstacles, and scattered task points was created. Both the simulation and experimental findings confirm the effectiveness and practical feasibility of the proposed methodology.

To make the problem computationally tractable, the model's construction relied on several simplifications, which inevitably detract from its absolute realism. For example, the energy cost associated with a drone's altitude changes is not

© 2024 The Author(s). | Published by Agro Intelligence Research Press

accounted for. Furthermore, in highly complex and dense scenarios—such as those involving large swarms of UAVs and UGVs, numerous target points, and restricted zones—the current method may fail to produce a feasible solution within an acceptable timeframe. To address these limitations and broaden the model's utility, our future research will focus on developing a novel algorithm for the objective function that delivers fast, high-quality approximate solutions.

Acknowledgements

Thanks to Xiaolong Cheng and Kaichuan Shen for their important technical help and for providing experimental equipment and related materials.

Conflict of interest

None

Financial support

This research was funded by the National Natural Science Foundation of China (grant number 52272414).

Ethics statement

None

Received: 28 Oct 2022 Revised: 02 Jan 2023 Accepted: 19 Jan 2023
Published online: 25 May 2024

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at https://creativecommons.org/licenses/by-nc-sa/4.0/.

References

1. Jeong J, Tian H, Du D. Trajectory-based data forwarding schemes for vehicular networks. ZTE Commun. 2014;12:17–25.

2. Yang J, Ding Z, Wang L. The programming model of air-ground cooperative patrol between multi-UAV and police car. IEEE Access. 2022;9:134503–17.

3. Roberts W, Griendling K, Gray A, Mavris D. Unmanned vehicle collaboration research environment for maritime search and rescue. In: Proceedings of the 30th Congress of the International Council of the Aeronautical Sciences, Daejeon, Republic of Korea; 2016.

4. Sowmya N, Costas A, Guy B, Regina L. Use of UAV-borne spectrometer for land cover classification. Drones. 2018;2:16–30.

5. Murray CC, Chu AG. The flying sidekick traveling salesman problem: optimization of drone-assisted parcel delivery. Transp Res Part C Emerg Technol. 2015;54:86–109.

6. Bouman P, Agatz N, Schmidt M. Dynamic programming approaches for the traveling salesman problem with drone. Networks. 2018;72:528–42.

7. Raj R, Murray C. The multiple flying sidekicks traveling salesman problem with variable drone speeds. Transp Res Part C Emerg Technol. 2020;120:102813.

8. Dell'Amico M, Montemanni R, Novellani S. Modeling the flying sidekick traveling salesman problem with multiple drones. Networks. 2021;78:303–27.

9. El-Adle AM, Ghoniem A, Haouari M. Parcel delivery by vehicle and drone. J Oper Res Soc. 2019;72:398–416.

10. Wang Z, Sheu JB. Vehicle routing problem with drones. Transp Res Part B Methodol. 2019;122:350–64.

11. Huawei M, Kai M, Jun G. Research on vehicle routing problem with drones considering multi-delivery. Comput Eng. 2022;48:299–305.

12. Schermer D, Moeini M, Wendt O. A matheuristic for the vehicle routing problem with drones and its variants. Transp Res Part C Emerg Technol. 2019;106:166–204.

13. Gu R, Poon M, Luo Z, Liu Y, Liu Z. A hierarchical solution evaluation method and a hybrid algorithm for the vehicle routing problem with drones and multiple visits. Transp Res Part C Emerg Technol. 2022;144:103733.

14. Tian XT, Xi QB. Real time path planning of UAVs using MILP. Comput Simul. 2009;26:72–5.

15. Lv Y, Wan X, Shi Y. Aircraft collision avoidance trajectory planning based on MILP. Flight Dyn. 2005;23:81–4.

16. Li DD, Sun XX, Sun B, Zhang Y. Mission planning for UAVs based on MILP. Flight Dyn. 2010;28:88–91.

17. Hu ZF, Chen Y, Zheng XJ, Wu HY. Cooperative patrol path planning method for air-ground heterogeneous robot system. Control Theory Appl. 2022;39:48–58.

18. Wu X, Yin Y, Xu L, Wu X, Meng F, Zhen R. Multi-UAV task allocation based on improved genetic algorithm. Mod Electron Tech. 2023;46:139–46.

19. Kim MH, Baik H, Lee S. Response threshold model based UAV search planning and task allocation. J Intell Robot Syst. 2014;75:625–40.

20. Zhang MY, Wang MY, Wang XD, Song X. Cooperative real-time task assignment of UAV group based on improved contract net. Aero Weapon. 2019;26:38–46.

21. Yan F, Zhu XP, Zhou Z, Tang Y. Real-time task allocation for a heterogeneous multi-UAV simultaneous attack. Sci Sin Inf. 2019;49:555–69.

22. Chen Y, Chen M, Chen Z, Cheng L, Yang Y, Li H. Delivery path planning of heterogeneous robot system under road network constraints. Comput Electr Eng. 2021;92:107197.

23. Wang X, Liu Z, Liu J. Mobile robot path planning based on an improved A* algorithm. Robot. 2018;40:903–10.

24. Challita U, Saad W, Bettstetter C. Interference management for cellular-connected UAVs: A deep reinforcement learning approach. IEEE Trans Wirel Commun. 2019;18:2125–40.

© 2024 The Author(s). | Published by Agro Intelligence Research Press

25. Liu W, Wang Y. Path planning of multi-UAV cooperative search for multiple targets. Electron Opt Control. 2019;26:35–8.

26. Abeywickrama HV, Jayawickrama BA, He Y, Dutkiewicz E. Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance. IEEE Access. 2018;6:58383–94.

27. Land AH, Doig AG. An automatic method for solving discrete programming. Econometrica. 1960;28:497–520.

28. Dakin RJ. A tree-search algorithm for mixed integer programming problems. Comput J. 1965;8:250–5.

© 2024 The Author(s). | Published by Agro Intelligence Research Press