



OPEN

# Integrated optimization of scheduling for unmanned follow-me cars on airport surface

Dezhou Yuan, Xiping Zhu✉, Yajun Zou &amp; Qing Zhao

To promote the application of automated vehicles in large airports, in this study, we present an integrated optimization method for scheduling Unmanned follow-me cars. The scheduling process is divided into three phases: Dispatch, Guidance, and Recycle. For the Dispatch phase, we establish a vehicle assignment model, to allocate the vehicle resource equitably. For the Guidance phase, we offer an quantitative way, to measure the spacing between Unmanned follow-me car and aircraft. To optimize the efficiency of airport operation in the three phases and ensure safety, the collaborative planning model, and the conflict prediction model are established. An improved grey wolf optimization algorithm is adopted to enhance the convergence speed and generalization performance. A case study at Ezhou Huahu Airport in China demonstrates the effectiveness of the methods. The results show that the model of collaborative planning can make the balance of path selection, Unmanned follow-me car's working time, and departure sequence. The convergence speed of the improved algorithm has been increased by 18.75%. The inequity index of vehicle assignment is only 0.015731, and the spatiotemporal distribution of conflicts is influenced by the airport's surface layout.

With the recovery of civil aviation after the epidemic of COVID-19, the introduction of unmanned vehicles at airports is considered as an innovative measure to improve airport operational efficiency and reduce costs. These vehicles, equipped with advanced sensors, controllers, enable autonomous driving functions with complex environment perception, intelligent decision-making, and collaborative intelligence. They have become the new generation of airport equipment for operations and support<sup>1</sup>. However, airport operation is a critical issue about safety, and it requires full consideration. Now that a single unmanned vehicle, has the capability for automatic control<sup>2</sup>, the overall risk control of the aviation transportation system in the hermetic environment of airport surface has become the focus. How to conduct scheduling of the fleet scientifically is a major problem for airports.

To balance the security and efficiency of the fleet, scholars have conducted research on the scheduling problem of unmanned vehicles in the airport scene. Basically, the scheduling process involves path planning and vehicle assignment<sup>3</sup>. For path planning, an optimized combination model is introduced. Because this model needs to discuss the situation in different velocities of the vehicles, it was solved with the heuristic, such as hill climbing particle swarm optimization (HC-PSO) algorithm<sup>4</sup>. For the other, the utilization ratio is considered as a kind of index, to evaluate the equity of vehicle assignment. A dynamic programming model with various charging strategies meets the needs of allocation for vehicle resource<sup>5</sup>. Generally, the scheduling problem is a classic vehicle routing problem with time windows (VRPTW). Different models and algorithms have been explored to address this challenge, including linear programming (LP), combinatorial optimization, graph theory, and network analysis<sup>6</sup>. Common algorithms include exhaustive methods, C-W savings algorithms, column generation algorithms, heuristic search, genetic algorithms (GA), and particle swarm optimization (PSO) algorithms. However, research on unmanned vehicle scheduling on airport surface mostly focused on electric tractors, which are subject to traction speed limitations<sup>7</sup> and do not meet the requirements of large-scale airports. There is a lack of research on the scheduling process of unmanned follow-me cars (UFMCs).

Regarding VRPTW on airport surface, integer linear programming (ILP) algorithms is utilized to determine the optimal allocation of ground support equipment (GSE) based on cost and time constraints<sup>8</sup>. The sorting of flight support operations for airport service vehicles is improved by GA, to optimize the structure and quantity of service vehicles. The objective is to enhance the efficiency of service vehicle utilization<sup>9</sup>. An energy consumption model for GSE is developed by the adaptive large neighborhood search (ALNS) algorithm, and provides scheduling solutions for scenarios involving both fuel-powered and electric vehicles, addressing the challenge of

College of Air Traffic Management, Civil Aviation Flight University of China, Deyang 618307, China. ✉email: zxp@cafuc.edu.cn

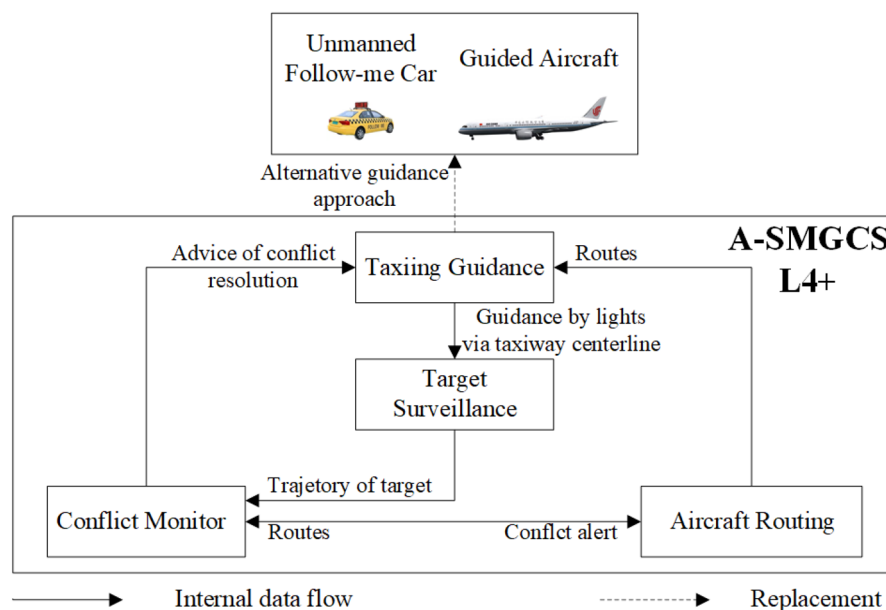
minimizing energy consumption<sup>10</sup>. Dispatching issue for electric GSE in the apron is addressed by hybrid neural networks (HNN), contributing to the overall improvement in efficiency<sup>11</sup>. An intelligent dispatching model for support vehicles is solved by GA, with a focus on optimizing the allocation of GSE and efficiency<sup>12</sup>. A theoretical system based on ILP is developed to solve the assignment problem of airport operational units, providing an effective framework for optimizing the allocation of operational units within the airport<sup>13</sup>. Collaborative decision-making for parking allocation and shuttle scheduling is successfully implemented through column generation (CG) algorithm, aiming to improve the efficiency of parking allocation and shuttle scheduling processes<sup>14</sup>. An objective function that minimizes operating costs and flight delay costs and optimizes GSE scheduling is devised with the help of ILP, to reduce operational costs and minimize flight delays through GSE scheduling<sup>15</sup>. Heuristic search methods are employed in a real-time scheduling decision system to efficiently assign GSE<sup>16</sup>.

Regarding aircraft path planning on airport surfaces, an ILP mathematical model for conflict-free aircraft trajectories in various traffic density scenarios is established with a rolling time domain calculation strategy<sup>17</sup>. A mixed ILP (MILP) model for gate allocation is developed by combining operation modes of multiple runway airports, aiming to optimize gate allocation<sup>18</sup>. Heuristic methods are implemented to search for the optimal speed profile for ground taxiing process, providing valuable references for aerodrome control units<sup>19</sup>. Optimal taxiing speed profiles were generated with an exhaustive method based on required time of arrival (RTA)<sup>20</sup>. Surface routing planning problems are also studied through multi-agent simulation<sup>21</sup>.

In conclusion, there are various optimization techniques employed in scheduling UFGCs on airport surfaces, including ILP, GA, ALNS, HNN, and heuristic search methods. Additionally, mathematical models such as ILP and MILP are utilized for path planning on airport surface. Some professional concepts such as RTA, energy management and flow of transportation management are introduced, that makes mathematical models meet the demands of business processes. However, further exploration is needed to develop more efficient and accurate optimization techniques for the scheduling of UFGCs on airport surface.

While previous research has provided insights into the issue of scheduling on airport surfaces, these studies have some limitations. Firstly, they mainly consider vehicles for small to medium-scale operations, typically involving up to 50 flights, which may not meet the demands of larger-scale vehicle scheduling. Secondly, although the vehicle assignment models may generate satisfactory solutions, they often fail to take uncertainties and flight support requirements into account. Finally, the evaluation of potential movement conflicts is not thoroughly addressed in these studies. The conflicts are either treated as disturbances or some mitigation suggestions are provided.

On the other hand, in advanced surface movement guidance and control systems (A-SMGCS) Level 4 and above, the surface movement of aircraft relies on three functional modules: target surveillance, conflict alert, and routing. However, the conventional method of surface taxiing guidance with lights of centerline on the taxiway proves to be expensive in terms of maintenance. It is not suitable for large and busy airports. To address this issue, an alternative approach within a generalized A-SMGCS implementation framework is proposed, which utilizes UFGCs as a replacement for taxiing guidance. Figure 1 shows the basic working principle of this approach, where the UFGC scheduling system integrates with the existing functional modules. In this concept, the aircraft follows the UFGC during taxiing, requiring close attention from the pilot for guidance. The advantage of UFGCs over traditional manual follow-me cars lies in their ability to formulate efficient vehicle scheduling schemes, enabling prompt departures from the parking lot and seamless integration with guidance tasks. Furthermore, UFGCs overcome the drawback of slower towing speeds associated with electric tractors, ensuring the efficiency of



**Figure 1.** Working principle of the UFGC scheduling system.

airport surface operations. Therefore, this paper aims to study the integrated optimization of UFMC scheduling to address this guidance requirement.

### Problem description

The integrated scheduling process for UFMCs to facilitate aircraft taxiing guidance is illustrated in Fig. 2. This process comprises vehicle trajectory planning and vehicle assignment. Vehicle trajectory planning involves generating optimal trajectories for the UFMCs across three phases: dispatch from the parking lot to the start of guidance (Dispatch), implementation of the guidance process (Guidance), and return to the parking lot after completing the guidance (Recycle). These trajectories are generated based on specific safety and efficiency objectives. Each vehicle's trajectory consists of a sets of position coordinates  $(x, y)$  along the route, corresponding velocities  $(v)$ , and timestamps  $(t)$ . Vehicle assignment refers to the selection of a specific UFMC from a pool of candidates based on certain optimization objectives to perform ground taxiing guidance for the corresponding flight.

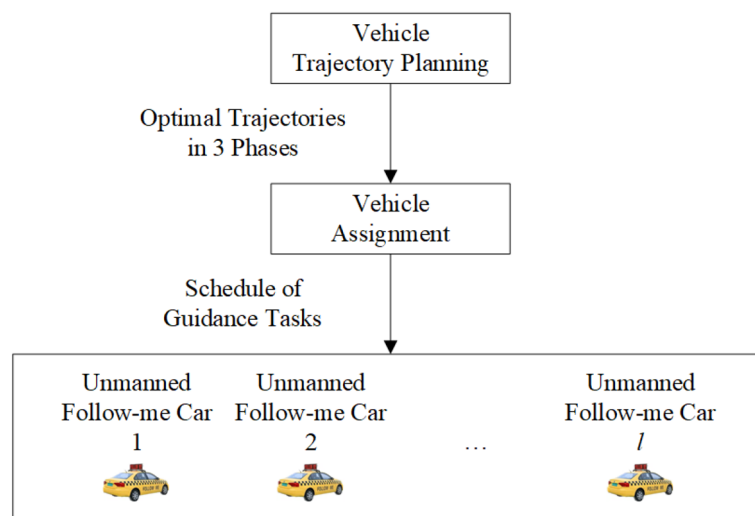
As can be observed, the precise planning of UFMCs in terms of starting, stopping, and speed profiles enables a higher level of refinement compared to manually operated follow-me cars. This allows vehicles to promptly reach the starting point of the guidance task and proactively plan to avoid potential conflicts during their operations. For the convenience of discussion, the UFMC and the aircraft being guided by it are considered as a "Guidance Unit (GU)" in the surface movement scenario.

From Fig. 3, during the Guidance phase, there exists a following relationship between the aircraft and the UFMC. The spacing between them affects the spatial occupancy of the GU. It also influences the scheduling decisions. To address this, a definition and a quantification method for the distance are provided:

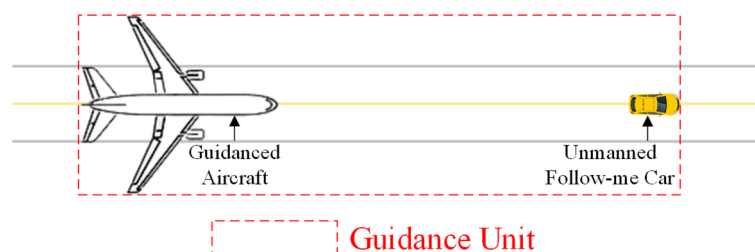
**Definition 1** (*Guidance following spacing*) Within the GU, the spacing  $\Delta L$  from the front of the aircraft, to the rear of the UFMC.

As shown in Fig. 4a, the measurement of  $\Delta L$  is similar to the concept of Following Spacing in transportation engineering<sup>22</sup>:

$$\Delta L = d_{AV} - \frac{l_A}{2} - \frac{l_V}{2} \tag{1}$$



**Figure 2.** The integrated scheduling process for UFMCs.



**Figure 3.** Schematic diagram of a GU.

where  $l_A$  and  $l_V$  are the lengths of the aircraft and the UFMC, respectively.  $d_{AV}$  represents the distance between their geometric centers.

During the following process, the visual influence to the pilot must be considered in order to comprehensively evaluate the safe guidance following spacing for each type of aircraft. As depicted in Fig. 4b, there exists the following relationship between the visual spacing  $d_{visual}$  in the cockpit, and the guidance following spacing  $\Delta L$  is calculated by the Pythagorean theorem.

### Model formulation

The scheduling model for UFMCs consists of two modules: vehicle trajectory planning and vehicle assignment. Figure 5 shows the details about this model:

- The vehicle trajectory planning module comprises the method for guidance trajectory deduction and a conflict prediction model. The former is responsible for rapidly deducting and generating a solution set of trajectories based on pre-planned vehicle routes and delivery times, with the input of guidance tasks. The latter is used to evaluate potential conflicts and determine the feasibility of the solution set. Finally, all feasible solution sets are integrated to generate an optimal solution set of trajectories.
- The vehicle assignment module transforms the optimal into corresponding guidance tasks, employing the UFMC assignment model to achieve the assignment function. Ultimately, the optimal scheduling scheme, i.e., the timetable of each guidance task, is obtained.

### Collaborative planning model for surface guidance trajectories

The set of flights to be guided denoted as  $G = \{g_1, g_2, \dots, g_s\}$ , where  $s$  means the total number of guidance tasks, i.e., the total number of flights. The three phases of the guidance task are denoted as  $P = \{p_1, p_2, p_3\}$ , corresponding to the Dispatch phase, Guidance phase, and Recycle phase.

#### Objective function

To meet safety and efficiency requirements, the objective function is designed from the following perspectives:

- Minimizing the number of conflicts

The number of conflicts be denoted as  $N_{conflict}$ , and set the objective to minimize the number of conflicts:

$$\min N_{Conflict} \tag{2}$$

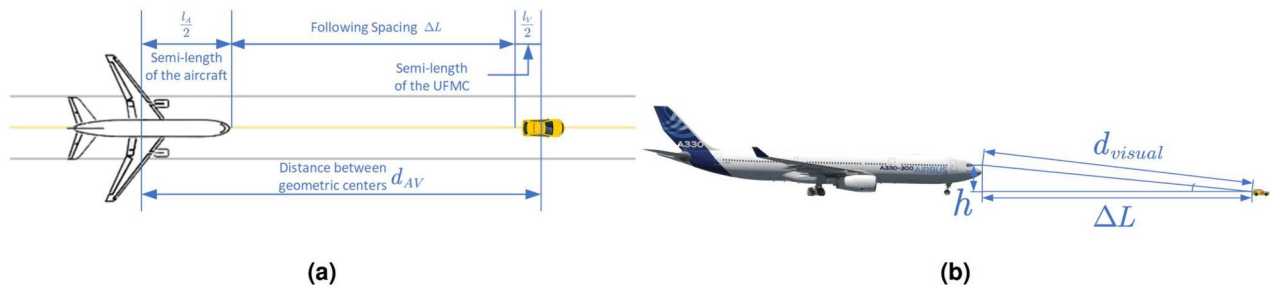


Figure 4. Measurement method for guidance following spacing. (a) Vertical view. (b) Lateral view.

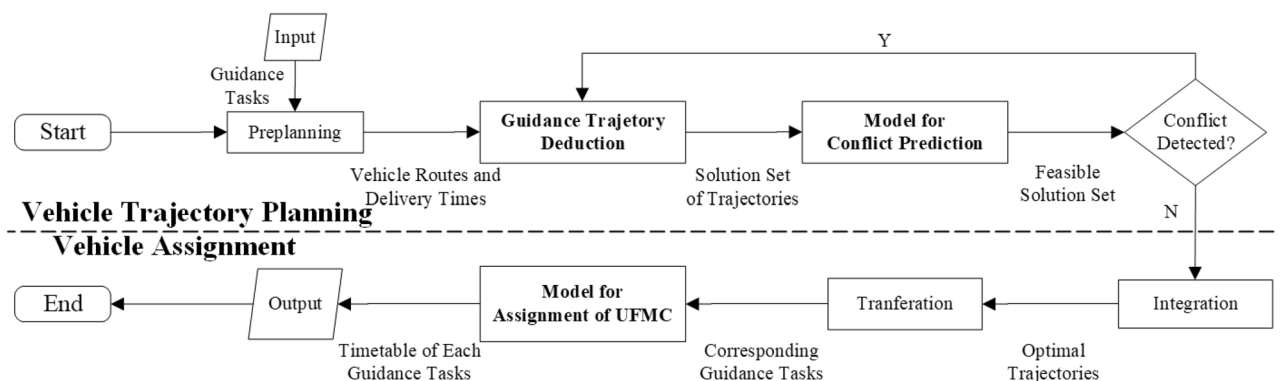


Figure 5. Framework of the scheduling model for UFMCs.

- Minimizing the working time of UFGCs

Minimizing the working time of UFGCs means reasonable route planning, and reducing the ineffective waiting time of UFGCs at the gap between two phases as possible. Therefore, the objective is to minimize the working time of UFGCs:

$$\min \sum_{i=1}^s \sum_{j=1}^3 t_{ij} \quad (3)$$

Here,  $i$  represents the  $i$ -th guidance task in the timetable,  $j$  represents the phase at which the UFGC is located, and  $s$  is the total number of flights in the timetable. The working time of UFGCs is the sum of the working times in each phase of the guidance tasks, and it is measured in seconds. Therefore, it could reach the magnitude of 100,000. Meanwhile, the number of conflicts is up to 1000 extremely.

To achieve the comprehensive optimization of the two objectives, the range of working time of UFGCs is normalized to  $[0,1]$  as follows:

$$\begin{cases} T = \sum_{i=1}^s \sum_{j=1}^3 t_{ij} \\ T' = 1 - \frac{1}{T} \end{cases} \quad (4)$$

Here, the working time of UFGCs is transformed into  $T'$ , to remove the influence of dimension while ensuring the monotonicity of the indicator.

Therefore, the objective function for vehicle trajectory planning is:

$$\min Z = \theta_1 T' + \theta_2 N_{\text{conflict}} \quad (5)$$

While setting  $\theta_1 = 1$ , the integral part of the objective function value  $Z$  maps the number of conflicts on the field, and the decimal part maps the total working time of UFGCs on airport surface. The value of  $\theta_2$  is determined by the scale of the airport surface, ensuring that  $\forall N_{\text{conflict}} \in \mathbb{N}$ ,  $\theta_2 N_{\text{conflict}}$  is one order of magnitude larger than  $\theta_1 T'^{23}$ , achieving simultaneous optimization of the two components.

#### Constraints

- Constraints of routing for UFGCs

For the three phases of vehicle guidance, the UFGC scheduling system has prestored one shortest route and  $(k^* - 1)$  sub-optimal routes for each pair of origins and destinations. During the process of scheduling, the system selects one from the  $k^*$  candidates as the route of UFGC, i.e.,

$$R_{ij} \leq k^*, \quad \forall i \in G, j \in P \quad (6)$$

where  $R_{ij}$  is an integer decision variable that represents the number of the route. The subscript  $i$  and  $j$  indicate that route selection is performed for each guidance task and each phase in the trajectory planning module.

- Constraints of time windows for UFGCs

The aerodrome control unit and the Airport Operation Center (AOC) adjust the delivery time windows of the UFGCs through collaborative decision-making throughout the flight process to achieve integrated optimization of scheduling of UFGCs and control of flights:

$$\gamma_{ij} \leq \delta_j, \quad \forall i \in G, j \in P \quad (7)$$

where  $\gamma_{ij}$  is the decision variable, corresponding to the delivery time adjustment for the  $i$ -th guidance task in phase  $j$ .  $\delta_j$  represents the upper limit of adjustment for each phase.

Taking guiding flight  $i$  as an example, the delivery times for the three phases of the UFGC are shown in Table 1. ETA and ETD are obtained from the flight schedule. They represent the Estimated Time of Arrival and Estimated Time of Departure, respectively. LDR is the time from the arrival aircraft touching down, to reaching the starting point of the Guidance. LUP is the time from the departure aircraft waiting at the runway entrance to having clearance of lining up the runway, respectively. CRS represents the time taken for the UFGC to pass through the taxiway on the apron, while UT is the time for the UFGC to make a U-turn in front of an arrival aircraft.

- Constraints of domain

Flight Type	Phase	Regular delivery time (RDT)	Actual delivery time (ADT)
Arrival	1	ETA+LDR- $t_{i1}$ -UT	RDT- $\gamma_{i1}$
Arrival	2	ETA+LDR	RDT- $\gamma_{i2}$
Arrival	3	ETA+LDR+ $t_{i2}$ +CRS	RDT- $\gamma_{i2} + \gamma_{i3}$
Departure	1	ETD-LUP- $t_{i2}$ - $t_{i1}$ -CRS	RDT- $\gamma_{i1} - \gamma_{i2} - \gamma_{i3}$
Departure	2	ETD-LUP- $t_{i2}$	RDT- $\gamma_{i2} - \gamma_{i3}$
Departure	3	ETD-LUP	RDT- $\gamma_{i3}$

**Table 1.** Delivery time of UPMC guiding flight  $i$  in three phases.

To ensure the meaningfulness of each variable, it is necessary to define their domain, including the number of conflicts as a natural number, the route number of the UPMC as a positive integer, and the non-negativity of the UPMC working time and delivery time adjustment:

$$N_{Conflict} \in \mathbb{N} \tag{8}$$

$$k, R_{ij} \in \mathbb{N}^+, \quad \forall i \in G, j \in P \tag{9}$$

$$t_{ij}, \gamma_{ij}, \delta_j \geq 0, \quad \forall i \in G, j \in P \tag{10}$$

*Trajectory deduction*

The purpose of guidance trajectory deduction is to deduce the trajectory of an UPMC or GU, for each guidance task at different phases based on a predetermined route and the delivery time. For UPMCs, precise control to parameters can be achieved based on the setting of guidance speeds in different zones, vehicle performance parameters, as well as road conditions, enabling accurate trajectory deduction to support the prediction of potential conflict in trajectories.

**Definition 2** *Trajectory of UPMC* The coordinates the UPMC’s geometric center passes through, along with corresponding timestamps and velocities.

**Definition 3** *Trajectory of GU* The coordinates the GU’s geometric center passes through, along with corresponding timestamps and velocities.

Since the preplanning process has generated several candidate routes, in order to accurately match the UPMC with the guidance requirements, further algorithm design is required, to generate high-precision velocity profiles and achieve synchronization between the actual delivery time and the start of the timestamp.

The input to the algorithm of Speed Profile Generation for UPMC is a set of nodes formed by a single candidate route. Then, correct the speed limit and determine the safety throttle/braking distance for each road segment, by basic kinematic equations. Finally, generate the final velocity profile:

$$\begin{cases} v_n = v_{n-1} \\ s_{n-1}^n = v_{n-1}(t_n - t_{n-1}) \end{cases} \tag{11}$$

$$\begin{cases} v_n = v_n' \\ t_n - t_{n-1} = \frac{s_{n-1}^n - s_{safe}}{v_n} + \frac{v_n - v_{n-1}}{a} \end{cases} \tag{12}$$

$$\begin{cases} v_n^2 - v_{n-1}^2 = 2as_{n-1}^n \\ s_{n-1}^n = \frac{v_n + v_{n-1}}{2}(t_n - t_{n-1}) \end{cases} \tag{13}$$

The algorithm is showed as following:

**Input**

1. Set of nodes on selected route  $\{(x_0, y_0), (x_1, y_1), (x_2, y_2), \dots, (x_t, y_t)\}$ , the sample of route is shown in Fig. 15;
2. Distance  $s_m^n$  between nodes  $m, n$  on the route;
3. Set of speed limits  $\{v'_1, v'_2, \dots, v'_t\}$  at each node;
4. Vehicle acceleration and deceleration  $a_a, a_d$ .

**Initialize**  $n = 1$ ;

Actual speeds of each node are set to zero:  $v_1 = 0, v_2 = 0, \dots, v_t = 0$ .

**Step 1 Adjust Speed Limits:**

- 1.1 If  $v'_n > v'_{n+1}$ , calculate  $v_{required}^m$  for its upstream nodes  $m = n - 1, n - 2, \dots, 1: v_n'^2 - v_{required}^m{}^2 = 2as_m^n$ ; otherwise, go to Step 1.3.
- 1.2 For each upstream node, if  $v'_m > v_{required}^m$ , set  $v'_m = v_{required}^m$ ;
- 1.3 If  $n = t - 1$ , set  $n = 1$  and go to Step 2; otherwise, set  $n = n + 1$  and back to Step 1.1.

**Step 2 Generate Speed Profile:**

- 2.1 If  $v'_n = v_{n-1}$ , generate  $v_n, t_n$  by Eq. (11); otherwise, go to Step 2.2.
- 2.2 If  $s_{n-1}^n$  is greater than  $s_{safe}$  calculated by  $v_n'^2 - v_{n-1}^2 = 2as_{safe}$ , set  $v_n = v'_n$ , and generate  $v_n, t_n$  by Eq. (12); otherwise, generate  $v_n, t_n$  by Eq. (13), and go to Step 2.3.
- 2.3 If  $n = t$ , terminate the algorithm; otherwise, set  $n = n + 1$  and back to Step 2.1.

**Output**

Speed profile  $\{(v_1, t_1), (v_2, t_2), \dots, (v_t, t_t)\}$  of corresponding trajectory, the sample of profile is shown in Fig. 16.

**Algorithm 1.** Speed Profile Generation for Unmanned Follow-me Car.

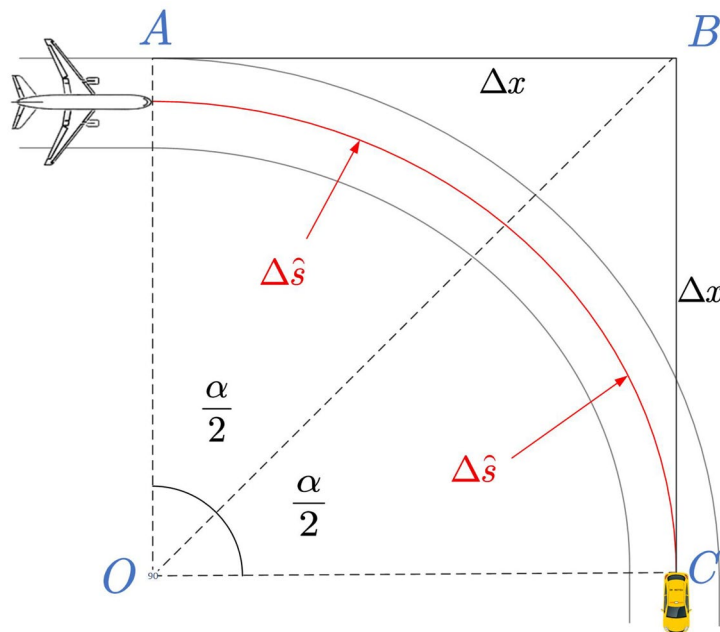
The process for the guidance trajectory deduction is as follows:

- Input the current phase  $j$ , the number of parking stand where flight  $i$  is located, and the selected route  $R_{ij}$ . Generate the set of nodes  $\{(x_0, y_0), (x_1, y_1), (x_2, y_2), \dots, (x_t, y_t)\}$  corresponding to the guidance route;
- Input the delivery time adjustment  $\gamma_{ij}$  and determine the delivery time, denoted as  $t_0$ , according to Table 1;
- Traverse the nodes on the trajectory, and determine the turning points via the dot product formula:

$$\cos \alpha = \frac{\vec{AB} \cdot \vec{BC}}{|\vec{AB}| |\vec{BC}|} \tag{14}$$

Where point B is the point to be determined, and points A and C are the previous and next nodes, respectively. If  $\alpha > 60^\circ$ , point B is identified as a turning point.

- Determine the speed limits  $\{v'_1, v'_2, \dots, v'_t\}$  for each point based on actual operational conditions:



**Figure 6.** Smoothing for segments at turning areas.



$$v'_i = v_{node} \quad (node \in E_{Apron}, E_{Taxiway}, E_{Turn}), \quad \forall i \in [1, t] \tag{15}$$

Where  $v_{node}$  represents the speed limit at the next node, i.e., the current section's speed limit.  $E_{Apron}$ ,  $E_{Taxiway}$ , and  $E_{Turn}$  represent the apron (including service roads), the taxiways in the maneuvering area, and the turning points, respectively;

- Call Algorithm 1 to generate the velocity profile  $\{(v_1, t_1), (v_2, t_2), \dots, (v_t, t_t)\}$ ;
- Smooth the segments at the turning areas as shown in Fig. 6.  $\forall node \in E_{Turn}$ , adjust the timing as follows:

$$\begin{cases} \Delta x = r \tan\left(\frac{\alpha}{2}\right) \\ \Delta \bar{s} = \left|\frac{\alpha}{2}\right| r \\ t_{Turn}' = t_{Turn} - \frac{2(\Delta x - \Delta \bar{s})}{v_{Turn}} \end{cases} \tag{16}$$

Where  $\alpha$  is the turning angle,  $r$  is the turning radius,  $\Delta x$  is the advance distance for the turn,  $\Delta \bar{s}$  is half of the arc length corresponding to the turning angle, and  $t'_{Turn}$  is the adjusted timing for passing the turning point.

- Output the guidance trajectory:  $T\{i, j\} = \{(x_n, y_n, v_n, t_n) | n = 0, 1, 2, \dots, t\}$ .

The guidance trajectory  $T\{i, j\}$  provides information about the working time of UFMC:

$$t_{ij} = t_t - t_0, \forall i \in G, j \in P \tag{17}$$

Moreover, the guidance trajectory  $T$  reflects the situation of the UFMC on surface, and serves as the foundation for the conflict prediction model.

### Conflict prediction

A conflict prediction model based on protected zones is established by using the coordinates of each position in the trajectory as the geometric centers. The properties of a protected zone are determined by trajectory's parameters of corresponding position, enabling the prediction of conflicts. Among commonly used protected zones, the elliptical demonstrates good prediction accuracy and intuitiveness.

The lengths of the semi-major axis "a" and the semi-minor axis "b" of the elliptical protected zone are calculated as following:

$$\begin{cases} a = \left(\frac{l_{target}}{2} + \frac{v^2}{2a_d}\right) \cdot k_{security} \\ b = \left(\frac{w_{target}}{2}\right) \cdot k_{security} \end{cases} \tag{18}$$

Here,  $k_{security}$  represents the safety factor.  $l_{target}$  and  $w_{target}$  denote the length and width of the UFMC or GU, respectively. During the Guidance phase, the moving target refers to the GU. During the Dispatch or Recycle phase, the target refers to the UFMC:

$$l_{target} = \begin{cases} l_V + l_A + \Delta \bar{L} & \text{if target is a Guidance Unit} \\ l_V & \text{else} \end{cases} \tag{19}$$

$$w_{target} = \begin{cases} w_A & \text{if target is a Guidance Unit} \\ w_V & \text{else} \end{cases} \tag{20}$$

Here,  $w_A$  represents the wingspan of the guided aircraft, and  $w_V$  represents the width of the UFMC.

The purpose of setting the safety factor  $k_{security}$  is to minimize the possibilities of both false alarms and missed alerts in conflict prediction. Figure 7 shows the scenarios of different safety factors corresponding to the protected area.

The black protected zone represents a safety factor of 1, which cannot envelop the entire moving target most of the time, potentially leading to missed alerts. The red protected zone corresponds to a safety factor of 1.2, which

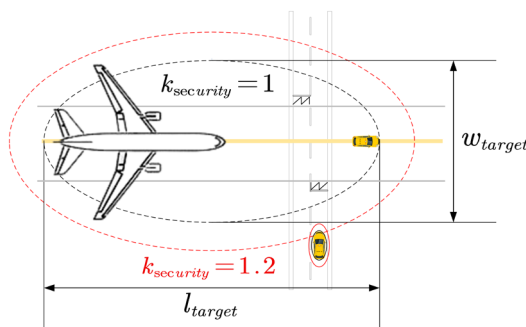
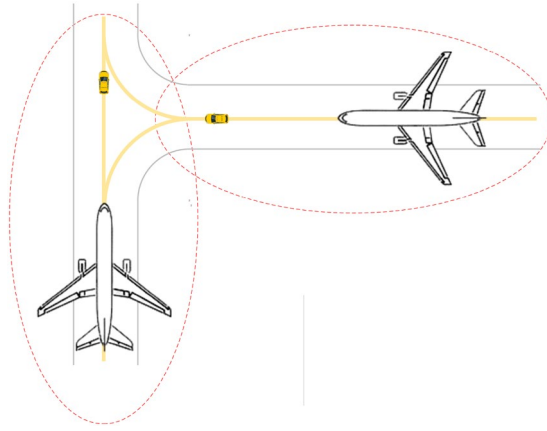


Figure 7. Schematic diagram of a protected zone.





**Figure 8.** Schematic diagram of a typical situation of conflict.

can envelop the entire moving target and minimize the probability of false alarms when potential conflicts are predicted. The Reference<sup>24</sup> has already discussed incidents of wingtip collisions on the apron, it is sufficient for their model to recognize and deal with such incidents. Therefore, our model eliminates the wingtip clearance  $\varepsilon$  proposed by Giuseppe S [7], and uses the length of the GU instead of the length of the aircraft in their tractor problem to ensure that there is sufficient space for our GU within the protected zone.

Figure 8 visually shows a typical scenario of conflict between GUs. The elliptical protected zone could accurately represent the spatial occupation of GU, and maintain safe spacing both longitudinally and laterally. To reproduce the protected zone, and determine whether the ellipses overlap or not, We have formulated the elliptical equation as following:

$$\left( \frac{(x - x_n) \sin \theta - (y - y_n) \cos \theta}{a} \right)^2 + \left( \frac{(x - x_n) \cos \theta + (y - y_n) \sin \theta}{b} \right)^2 = 1 \quad (21)$$

Subsequently, the Algorithm 2 is used to output the number of conflicts, denoted as  $N_{Conflict}$ , thereby completing the conflict prediction.

### Input

1. Set of trajectories  $T\{i, j\} = \{(x_n, y_n, v_n, t_n) | n = 0, 1, 2, \dots, t\}, \forall i \in G, j \in P$ ;
2. Set of nodes  $\{V_1, V_2, \dots, V_m\}$  on aerodrome surface.

**Initialize**  $i = 1, j = 1$ ;

$m \times 86400$  table  $\Omega$  filled with zeros.

**Step 1** Set  $n = 0$ .

**Step 2** Generate corresponding ellipse by Eq. (21).

**Step 3** For each node  $V_\lambda (\lambda = 1, 2, \dots, m)$ , if  $V_\lambda$  is in the ellipse, add one to element at row  $\lambda$ , column  $t_n$  of table  $\Omega$ .

**Step 4** If  $n = t$ , go to **Step 5**; otherwise, set  $n = n + 1$  and return to **Step 2**.

**Step 5** If iteration of  $i, j$  are finished, go to **Step 6**; otherwise, set next combination  $i, j$  and return to **Step 1**.

**Step 6** Compute the summation  $N$  of all the elements exceeding the threshold value of 1 in table  $\Omega$ .

**Step 7** Count the number  $M$  of elements in the table  $\Omega$  with values exceeding 1.

**Step 8** Set  $N_{Conflict} = N - M$ , terminate the algorithm.

### Output

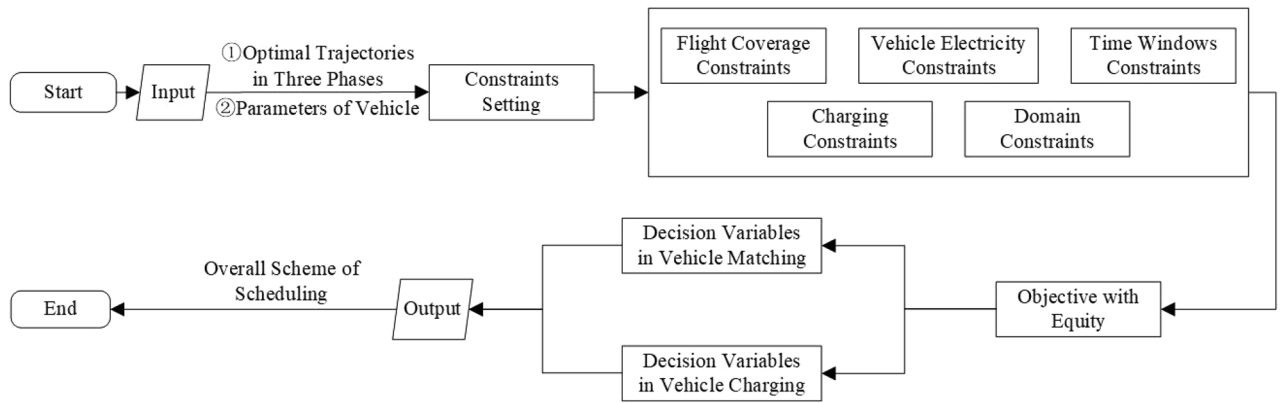
Overall frequency of conflict  $N_{Conflict}$ .

Table  $\Omega$  saving frequency of Conflict at each node.

**Algorithm 2.** Conflict Prediction Algorithm.

### Model for UFMC assignment

Compared to traditional manually-operated vehicles that rely on human experience for scheduling, operation, and maintenance, UFMCs overcome the limitations of manual driving through fully automated control. Building upon precise trajectory planning, further considerations for vehicle maintenance and optimal allocation of vehicle resources are necessary. In this regard, an UFMC assignment model shown in Fig. 9 is designed in the UFMC scheduling system, wherein the optimal solution set of trajectories is transformed into specific guidance tasks, to achieved the function of assignment through ILP.



**Figure 9.** Framework of the model of UFMC Assignment.

Considering that the source assignment problem can attribute to 0-1 programming. According to the general method in management of civil aviation<sup>13</sup>, we designed a model for vehicle assignment as following:

$$\min G = \frac{k+1}{k} - \frac{2}{k} \sum_{i=1}^s \frac{\sum_{l=1}^k (k-l+1)x_{il}}{\sum_{j=1}^3 (t_{ij} + \eta)}$$

s.t.

$$\sum_{l=1}^k x_{il} = 1, \quad \forall i \in G$$

$$x_{ol} + x_{pl} \leq 1, \quad \forall l \in F$$

$$\begin{cases} a_o \leq a_p + \eta + \sum_{j=1}^3 t_{pj} \\ a_o + \eta + \sum_{j=1}^3 t_{oj} \geq a_p \end{cases}$$

$$\begin{cases} q \sum_{i=1}^s d_{il} - \sum_{i=1}^{i^*} \left( \left( \sum_{j=1}^3 t_{ij} \right) + \eta \right) x_{il} \geq 0 \\ q \sum_{i=1}^s d_{il} - \sum_{i=1}^{i^*} \left( \left( \sum_{j=1}^3 t_{ij} \right) + \eta \right) x_{il} \leq q + q_{reserve} \end{cases}, \forall i^* \in G, l \in F$$

$$x_{ol} + x_{rl} \leq 2 - d_{ol}, \forall l \in F$$

$$\begin{cases} a_o \leq a_r + \eta + C + \sum_{j=1}^3 t_{rj} \\ a_o + \eta + C + \sum_{j=1}^3 t_{oj} \geq a_r \end{cases}, \forall l \in \mathbb{C}_F \{f_k\}$$

$$x_{il} = \begin{cases} 1 & \text{if vehicle } l \text{ guides flight } i \\ 0 & \text{else} \end{cases}$$

$$d_{il} = \begin{cases} 1 & \text{if vehicle } l \text{ is charged after flight } i \\ 0 & \text{else} \end{cases}$$

$$x_{il} - d_{il} \geq 0, \forall i \in G, l \in F$$

*Model assumptions*

- The UFMCs at the end of the Recycle phase must return to the parking lot where they started, in order to perform the next guidance task.
- A buffer time of  $\eta$  is allocated between two guidance tasks to ensure orderly traffic flow in and out of the parking lot, or transitioning to the corresponding parking lot for the next guidance task.
- A vehicle work continuously until its battery level reaches a low state, then return to the parking lot for a full recharge. The charging time  $C$  and the safe endurance  $q$  are fixed values specific to each type of vehicle. Factors such as battery decay and seasonal variations are ignored, but the buffer time  $\eta$  is included in the continuous working time.

Let  $F = \{f_1, f_2, \dots, f_k\}$  represent the set of  $k$  UFMCs in the airport.

*Objective function*

This model aims to achieve the most equitable vehicle assignment, so we adopt the Gini coefficient as the evaluation metric. In the field of economics, a Gini coefficient between 0.3 and 0.4 is considered fair<sup>25</sup>. However,

considering that the working time of UFMCs is a “negative asset”, the Gini coefficient is referred to as the “assignment inequality index” in this context. Therefore, the objective function is set to minimize the assignment inequality index of the working time for UFMCs:

$$\begin{aligned} \min G &= \sum_{l=1}^k P_l Y_l + 2 \sum_{l=1}^k P_l (1 - V_l) - 1 \\ &= \frac{k+1}{k} - \frac{2}{k} \sum_{i=1}^s \frac{\sum_{l=1}^k (k-l+1)x_{il}}{\sum_{j=1}^3 (t_{ij} + \eta)} \end{aligned} \tag{22}$$

Where  $P_l = \frac{1}{k}$ , and  $Y_l$  represents the proportion of working time for the  $l$ -th vehicle to the total working time, and  $V_l$  is the cumulative distribution of  $Y_l$ .  $\sum_{j=1}^3 t_{ij}$  denotes the accumulated time for the three phases of a single guidance task.  $x_{il}$  is a binary decision variable indicating whether vehicle  $l$  guides flight  $i$ .

*Constraints*

- Constraints of flight coverage

Each flight should be assigned to one UFMC ONLY:

$$\sum_{l=1}^k x_{il} = 1, \quad \forall i \in G \tag{23}$$

- Constraints of consecutive guidance tasks

The time windows of guidance tasks assigned to the same vehicle should not overlap. Let  $o$  and  $p$  represent a pair of guidance tasks corresponding to flights where their time windows overlap each others, then the constraints can be formulated as follows:

$$x_{ol} + x_{pl} \leq 1, \quad \forall l \in F \tag{24}$$

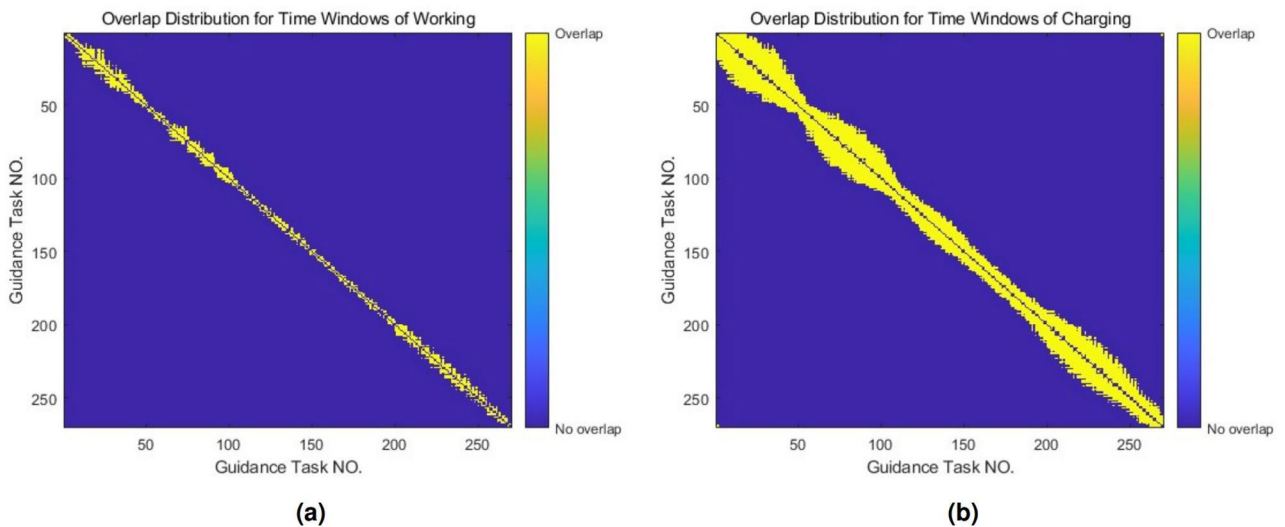
The condition that the time windows of guidance tasks  $o$  and  $p$  overlap can be expressed as follows:

$$\begin{cases} a_o \leq a_p + \eta + \sum_{j=1}^3 t_{pj} \\ a_o + \eta + \sum_{j=1}^3 t_{oj} \geq a_p \end{cases} \tag{25}$$

Here,  $a_o$  and  $a_p$  correspond to the actual delivery times (ADT) of guidance tasks for flights  $o$  and  $p$ , respectively.

Usually, guidance tasks are sorted by ADT. Fig. 10a shows the overlap situation of time windows between adjacent index of guidance tasks.

- Constraints of vehicle electricity



**Figure 10.** Overlap contribution. (a) For time windows of working. (b) For time windows of charging.

Constraints of vehicle electricity ensure that the quantity of electricity of the UFGCs meets both upper and lower limit requirements during operation and forms a closed loop for charging:

$$\begin{cases} q \sum_{i=1}^s d_{il} - \sum_{i=1}^{i^*} \left( \left( \sum_{j=1}^3 t_{ij} \right) + \eta \right) x_{il} \geq 0 \\ q \sum_{i=1}^s d_{il} - \sum_{i=1}^{i^*} \left( \left( \sum_{j=1}^3 t_{ij} \right) + \eta \right) x_{il} \leq q + q_{reserve} \end{cases} \quad (26)$$

$, \forall i^* \in G, l \in F$

Here,  $d_{il}$  is the binary representing whether vehicle  $l$  requires charging after completing task  $i$ .  $q_{reserve}$  is a backup of the quantity of electricity that ensures the UFGC can safely return to the parking lot after operating for a continuous duration of  $q$ .

- Constraints of charging

For single vehicle, the guidance task should not have conflict with the charging process:

$$x_{ol} + x_{rl} \leq 2 - d_{ol}, \quad \forall l \in F \quad (27)$$

The conflict condition between the time windows of guidance tasks  $o$  and  $r$  can be expressed as follows:

$$\begin{cases} a_o \leq a_r + \eta + C + \sum_{j=1}^3 t_{rj} \\ a_o + \eta + C + \sum_{j=1}^3 t_{oj} \geq a_r \end{cases} \quad (28)$$

Figure 10b shows the overlap situation between the time windows of a guidance task with charging demand, and its follow-up tasks.

- Constraints of vehicle sorting

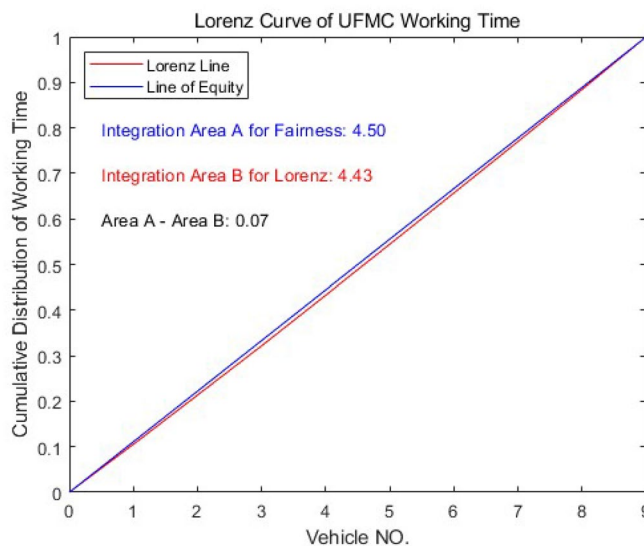
As Fig. 11 is shown, the calculation of assignment inequity index is related to the equity curve and the Lorenz curve. Both of them are formulated on the basis of cumulative distribution of UFGC working time. Therefore, each element of the set of vehicles  $F = \{f_1, f_2, \dots, f_n\}$  should be sorted in ascending order based on its working time:

$$\sum_{i=1}^s \sum_{j=1}^3 (t_{ij} + \eta) x_{il} \leq \sum_{i=1}^s \sum_{j=1}^3 (t_{ij} + \eta) x_{i(l+1)} \quad (29)$$

$, \forall l \in \mathbb{C}_F \{f_k\}$

- Constraints of domain

$$x_{il} = \begin{cases} 1 & \text{if vehicle } l \text{ guides flight } i \\ 0 & \text{else} \end{cases} \quad (30)$$



**Figure 11.** Calculation of assignment inequity index with equity curve and Lorenz curve.

$$d_{il} = \begin{cases} 1 & \text{if vehicle } l \text{ is charged after flight } i \\ 0 & \text{else} \end{cases} \quad (31)$$

$$x_{il} - d_{il} \geq 0, \quad \forall i \in G, l \in F \quad (32)$$

## Solution algorithm

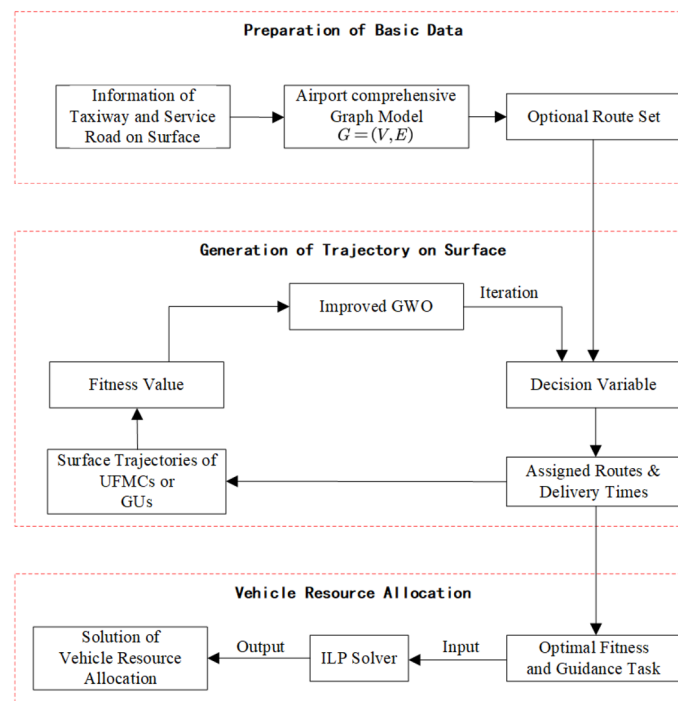
In this section, we design corresponding solution algorithm based on characteristics of models mentioned, and construct an overall solution framework for the scheduling problem for UFMCs.

As Fig. 12 is shown, the solution process for the scheduling model for UFMCs consists of three steps: preparation of basic data, generation of trajectory on surface, and vehicle resource allocation.

- In the step of basic data preparation, a comprehensive graph model  $G = (V, E)$  is generated, where the edges  $E$  correspond to various road sections in the scene, including  $E_{Apron}$  (the service roads, taxiways and taxi links within the apron),  $E_{Taxiway}$ , taxiways in the maneuvering area, as well as  $E_{Turn}$ , the various intersections and turns. The vertices  $V$  represent the endpoints of the edges, including key points like turning points, entrances/exits, and crossings. Then, generate a set of candidate routes for each pair of origins and destinations (ODs) on airport surface<sup>26</sup>.
- In the step of generation of trajectory on surface, the decision variables  $\{(R_{ij}, \gamma_{ij}) | i \in G, j \in P\}$  are transformed into the assigned routes and delivery times in three phases of each guidance task. A function of fitness is formed by integrating the collaborative planning model for surface guidance trajectories and the conflict prediction model based on protection zones. The optimal set of surface guidance trajectories are formed iteratively by the IGWO algorithm.
- In the step of vehicle resource allocation, the optimal set serves as the input for the UPMC assignment model. The model is solved directly using a solver to output the allocation scheme for vehicle resource.

The complexity analysis shows that the scale of feasible solution of the collaborative planning model for surface guidance trajectories is extremely large, with  $2^s$  possible combinations of ONLY the route components. When the number of daily flights exceeds 50, there is likely to be a combinatorial explosion problem. The scale of the decision variable for the UPMC assignment problem is  $(2ns)$ , which is a relatively manageable classic ILP problem and does not require a heuristic algorithm additionally.

Therefore, to address the collaborative planning model for surface guidance trajectories, an IGWO algorithm based on integer encoding is designed to improve the efficiency of generation for the optimal set of surface trajectories.



**Figure 12.** Framework of solution method.

### Encoding strategy and initialization

When we apply this algorithm, it is necessary to define the meaning of each dimension in the position vector of a grey wolf individual. Based on the decision variables of the collaborative planning model, targeting guidance tasks for  $s$  flights, we construct an integer space with  $6s$  dimensions as shown in Table 2.

Considering that the UPMC scheduling system may avoid conflicts by adjusting routes, shifting time windows, or using a combination of both, conflict-free solution sets would be dispersed at multiple locations, we randomize the wolf pack initially, then integer by rounding to construct the integer space. This serves as the start for iteration and optimization.

### Fitness function

The calculation of the fitness function needs to consider the total working time on airport surface, which is based on the selected route, and needs to deduct the situation of potential conflict between the selected route and the variation in delivery time. Therefore, the fitness function  $f$  is defined as follows:

$$f = \theta_1 \left( 1 - \frac{1}{\sum_{i=1}^s \sum_{j=1}^3 t_{ij}} \right) + \theta_2 \sum_{t^*=1}^{86400} \text{card}(node \in \{Ellipses_{t^*}\}) \tag{33}$$

In this equation, the first term, which relates to the target parking number and selected route, can be obtained by from the pre-computed values of  $t_{ij}$  based on the given timetable of flights and information of grey wolves. The second term requires traversing each unit of time for movement on the surface, and  $\forall node$ , we need to count the number of times a node falls in different ellipses.

### Enhancing global search

In the local search, the slight decimal-level changes in the positions of the wolf pack in the traditional GWO algorithm are not suitable for integer encoding, so we adopt the IGWO algorithm proposed by Yang Z<sup>27</sup> to enhance the prominence and efficiency of global search during the iteration process. The formula for updating the convergence factor is given as follows:

$$a' = 2e^{-\frac{t}{T}} \tag{34}$$

Where the improved convergence factor  $a'$  starts from the default value of 2 and exponentially decays to  $\frac{2}{e}$  with the iteration process. When approximately 69.3 percents of the iteration process is completed, the improved algorithm enters the local search, and the global-to-local search ratio shifts from 1:1 to 7:3.

### Improvement for local search

For the local search, we adopt a probabilistic method for updating position<sup>28</sup>:

$$X(t + 1) = \begin{cases} X_1 & r_c \in (0, \frac{1}{3}] \\ X_2 & r_c \in (\frac{1}{3}, \frac{2}{3}] \\ X_3 & r_c \in (\frac{2}{3}, 1] \end{cases} \tag{35}$$

Here,  $r_c$  is a random number that satisfies  $U(0, 1)$ . The formula for updating position based on integer space not only reflects the dynamic process of the wolf pack following the leader, but also ensures the meaningfulness of the operation of updating position for  $X(t + 1)$  in the local search.

	$g_1$	$g_2$	...	$g_s$	Explanation
$X_{i1}$	1	1		1	Route NO. at Phase 1
$X_{i2}$	2	1		1	Route NO. at Phase 2
$X_{i3}$	1	2		1	Route NO. at Phase 3
$X_{i4}$	55	6		45	Time adjustment at Phase 1
$X_{i5}$	488	20		271	Time adjustment at Phase 2
$X_{i6}$	33	276		54	Time adjustment at Phase 3

**Table 2.** The vector of grey wolf  $X$  with integer encoding.

### IGWO algorithm

In summary, the flow of the IGWO algorithm is shown as following:

#### Input

1. Initialized vector  $X$  introduced in **Section 5.1**;
2. The total iterations;
3. Random variables  $r_1, r_2$  meeting  $|r_1|, |r_2| \in U(0, 1)$ .

**Step 1** Calculate  $a'$  by Eq. (34).

**Step 2** Simulate the hunting process as traditional GWO:

$$\text{set } A = 2a' \cdot r_1 - a', C = 2 \cdot r_2,$$

$$D = |C \cdot X_\phi(t) - X(t)|,$$

$$X_\phi(t+1) = X_\phi(t) - A \cdot D.$$

**Step 3** If  $a \geq 1$ , Renew  $X$  by:  $X(t+1) = \text{round}\left(\frac{X_1+X_2+X_3}{3}\right)$ ;

otherwise, renew  $X$  by Eq. (35).

**Step 4** Analysis fitness  $f$  corresponding  $X$  as traditional GWO.

**Step 5** If Iteration is finished, terminate the algorithm; otherwise, begin next iteration and back to **Step 1**.

#### Output

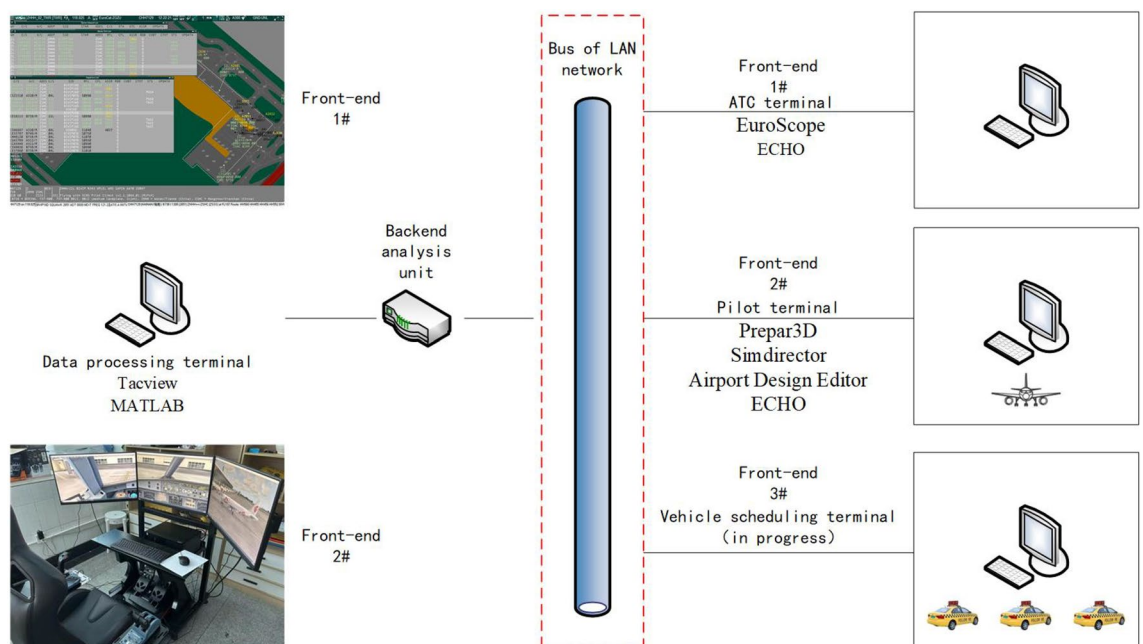
Best fitness  $f$  and corresponding  $X$ .

**Algorithm 3.** IGWO Prediction.

#### Preparation of data

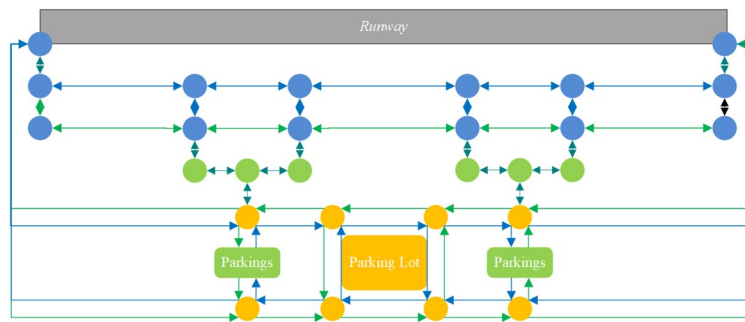
Ezhou Huahu Airport is a specific cargo airport in China, it is also an important experiment field for unmanned ground support vehicles, but there is a lack of overall scheduling for large-scale fleet. Taking Ezhou Huahu Airport for simulation experiment, we investigate the optimization of scheduling for UFGCs during a typical working day in 2030, with 270 flights per day, and assume that runway-to-stand assignments are based on the principle of proximity.

Before starting the experiment, we has built a simulation platform. It is divided into air traffic control (ATC) terminal, pilot terminal, and data processing terminal. The interaction between terminals, and the details about hardware and software are shown in Fig. 13.

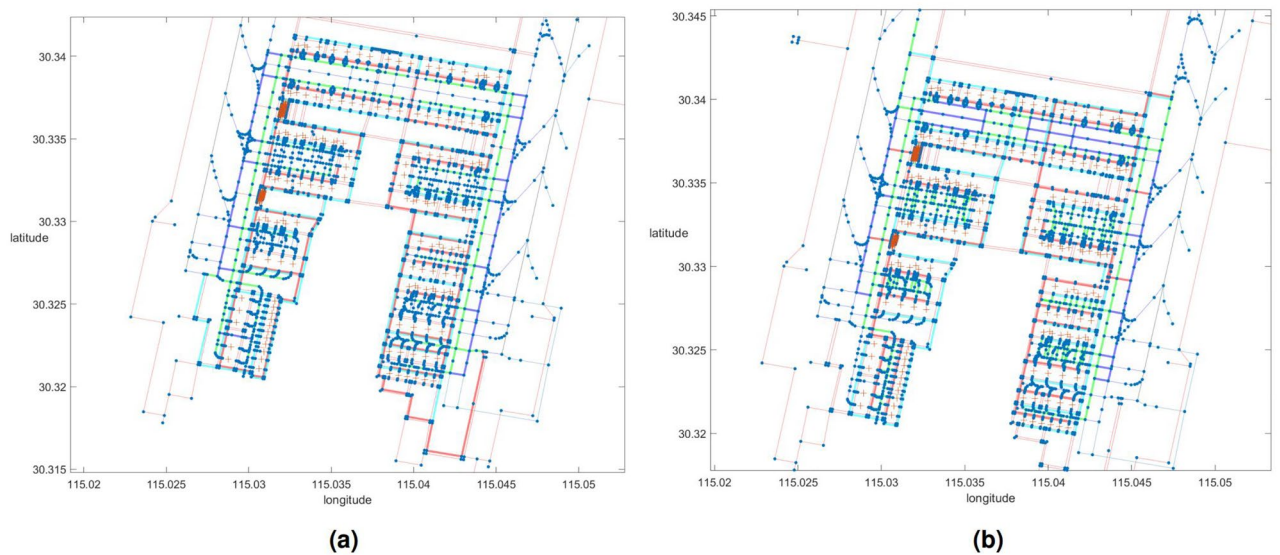


**Figure 13.** Schematic diagram of the UFMC's routes.





**Figure 14.** Schematic diagram of the UFMC’s routes.



**Figure 15.** Schematic diagram of optional route set for Ezhou Airport. **(a)** Route set for Departure. **(b)** Route set for Arrival.

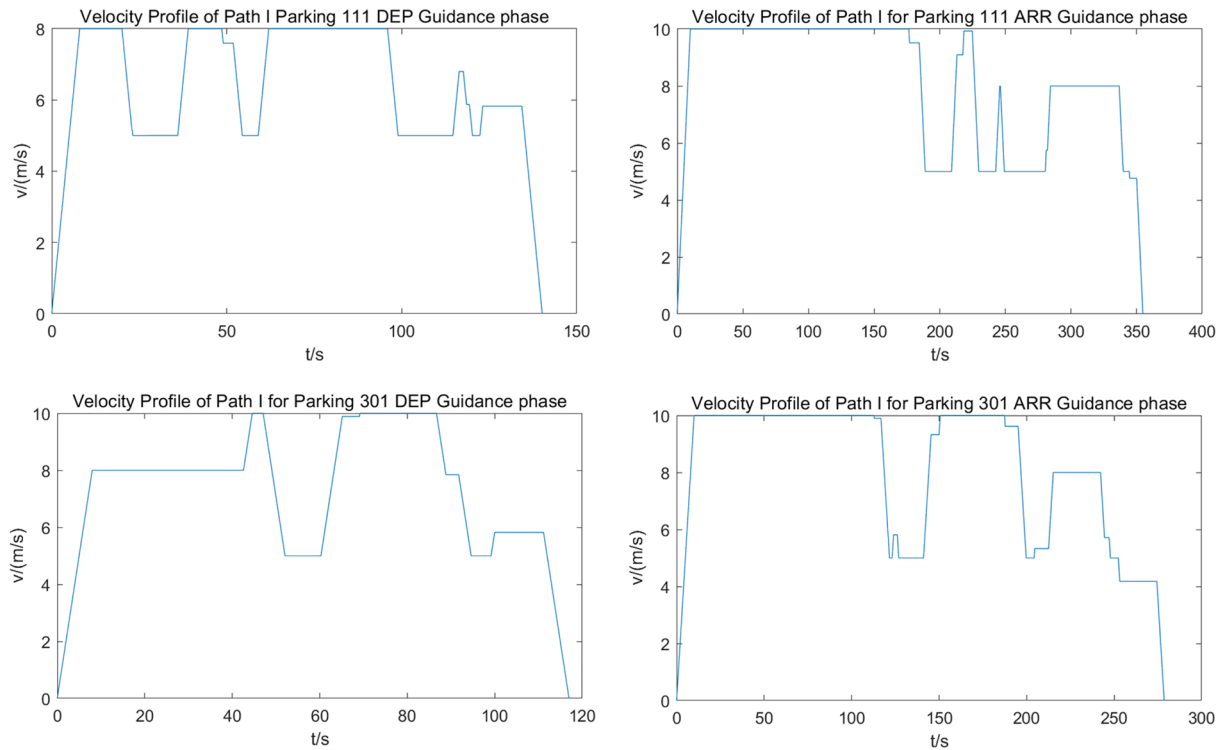
**Basic routes**

Figure 14 shows the schematic diagram of the UFMC’s routes. During phases 1 and 3 (operations on service roads), the strategy of separating inner and outer loops is employed. In phase 2 (on-taxiway) operations, the “fixed, one-way, directional, cyclic” principle<sup>29</sup> is followed, where alternative routes are assigned to vehicles, to prevent and alleviate congestion, and to minimize the occurrences of conflict initially. The number of alternative routes, denoted as  $k$ , is set to 2. The blue and green arrows in the figure represent Route 1 and Route 2, respectively.

Based on the strategy for route allocation, Ezhou Airport’s UFMCs have a set of selectable routes shown in Fig. 15. Figure 15a and b show the departure and arrival scenarios, respectively. In these figures, Route 1 is highlighted in blue (shared with Route 2 in green) during the Guidance phase, while Route 2 is labeled in green. The Dispatch and Recycle phases are marked by the red and cyan lines for Route 1 and 2, respectively.

Parameter	Explanation	Value	Parameter	Explanation	Value
$q$ (h)	Safety endurance	12	$a_a$ [m/s <sup>2</sup> ]	Acceleration	1
$q_{reserve}$ (min)	Reserve endurance	15	$a_d$ [m/s <sup>2</sup> ]	Deceleration	1
$v_{Taxiway}$ (m/s)	Velocity on main taxiways	10	$l_{car}(m), w_{car}(m)$	Length and width	4.8, 1.8
$v_{Apron}$ (m/s)	Velocity in apron	8	$C$ (min)	Required Time for Charging	45
$v_{turn}$ (m/s)	Velocity at turns	5	$\Delta L$ (m)	Average Following Space	60

**Table 3.** Parameter settings for the fleet of UFMCs.



**Figure 16.** The velocity profiles along typical routes.

### Parameter settings

According to the actual operation of Ezhou Airport and the requirements of Chinese regulations [20], the parameters of the UPMC are shown in Table 3. The UPMC is allowed to arrive in advance and wait for a maximum of 60s. The time windows for air traffic controllers to adjust the ETD and ETA of flights are 600 s and 300 s, respectively. Therefore, in the case of arrival, the upper limits of the change in delivery times for UPMCs ( $\gamma_{i1}$  to  $\gamma_{i3}$ ) are 60s, 300s, and 60s, respectively; for departure, those are 60s, 60s, and 600s, respectively.

### Profiles generation

To illustrate this process, we selected Route 1, along with guarded flights at Stand 111 and 301 as examples, and generated velocity profiles by Algorithm 1. Figure 16 shows the typical velocity profiles.

In Fig. 16, the velocity of the GU remains stable at 5 m/s, 8 m/s, and 10 m/s, corresponding to turns, aprons, and straight sections of taxiways in maneuvering areas, respectively. As shown in the second half of each profile, the UPMC adjusts its cruising speed, throttle, and brake, based on a comprehensive modification of the length of each segment and the initial settings, through the application of the Algorithm 1. The velocity is changed when the GU passes through different areas and segments, suggests that the efficiency of the UPMCs scheduling is related to the number of turns, taxiing distance, and timing of crossing the apron area, which confirms the importance of route selection in the trajectory planning module.

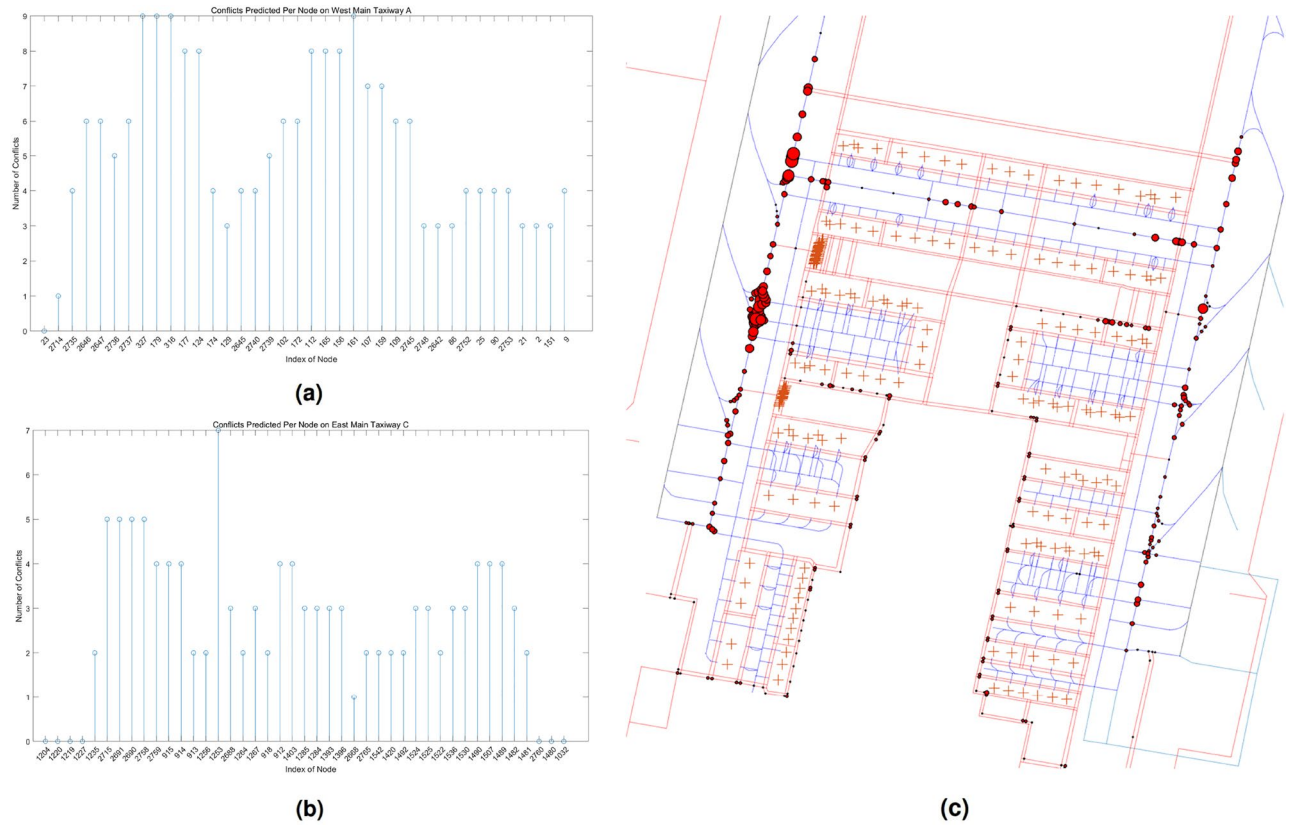
### Conflicts analysis

An analysis of conflict characteristics before the optimization of scheduling was conducted. Without considering the selection of surface guidance trajectories and adjustments to delivery times, Algorithm 2 predicted 824 instances of conflicts among the trajectories. Figure 17a, b, and c show the characteristics distribution of conflicts spatially, for the west main taxiway, the entire surface, and the east main taxiway, respectively. The characteristic of “high in the north, low in the south” of the distribution of conflicts indicates that, when operating towards the north, the conflicts during arrival are higher than those during departure. On the other hand, due to the coexistence of passenger and cargo parking stands on the west side of Ezhou Airport, the distribution also exhibits the characteristic of “high in the west, low in the east”.

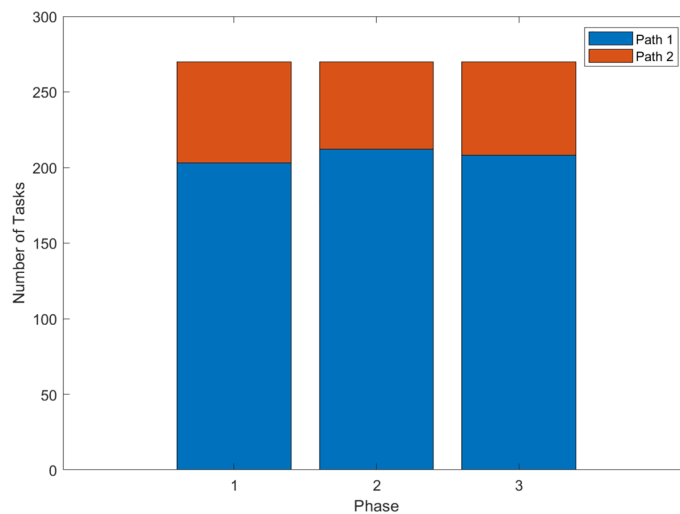
## Results and discussion

### Regards to vehicle trajectory planning

After the computation of the vehicle trajectory planning module, the UPMC scheduling system generated a set of conflict-free guidance trajectories, with a total working time of 170,463s. The surface guidance trajectory is mainly planned collaboratively in three aspects: route selection, UPMC’s working time, and delivery sequence, achieving the effect of optimization. Among them, the distribution of route selection is shown in Fig. 18. The histogram indicates that Route 1 with shorter travel time remains the top choice for UPMCs. In phases 1 to 3,



**Figure 17.** Distribution of conflicts before optimization. The southern end of the taxiway corresponds to the right-hand side of the horizontal axis. (a) Conflicts on the west main taxiway. (b) Conflicts on the east main taxiway. (c) Conflicts on airport surface.

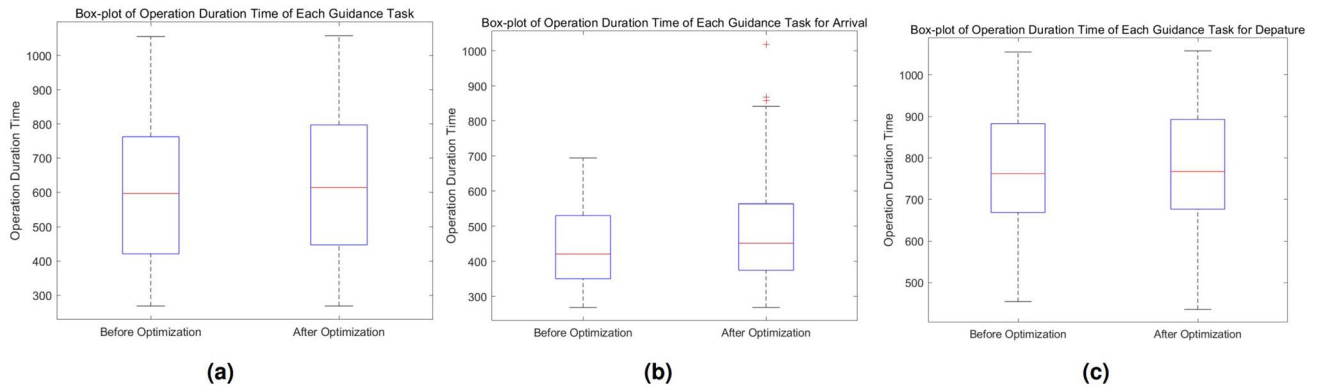


**Figure 18.** Histogram for frequency distribution of route selection.

24.81 %, 21.48 %, and 22.96 % of guidance tasks respectively choose Route 2 to avoid potential conflicts at the cost of detour.

Adjustments in UFMC working time are shown in Fig. 19. Figure 19a shows that by extending the working time of UFMCs by 2.85%, the vehicle trajectory planning module achieves the resolution for all the conflicts. Figure 19b and c show that sacrificing operational efficiency is targeted. For arrival guidance tasks with higher conflict frequency, 7.15% of efficiency is sacrificed, while for departure, the change is minimal, with only a 0.66% extension of UFMC working time.

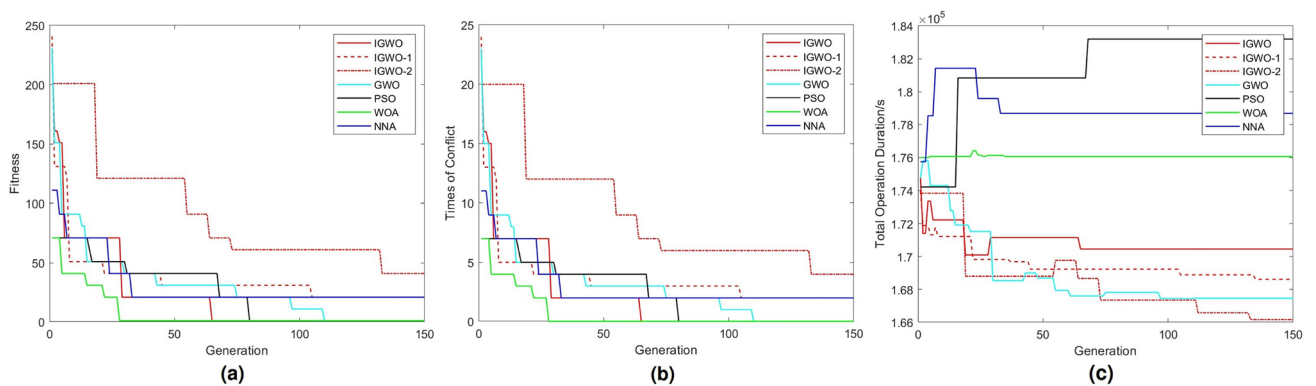
Regarding the result of delivery sequence shown in Table 4, a significant number of guide tasks have been adjusted by fine-tuning the delivery time to avoid conflicts. This has altered the regular delivery sequence,



**Figure 19.** Box-plot of UFMC working time. (a) Overall distribution of working time. (b) Distribution of working time for arrival. (c) Distribution of working time for departure.

Regular delivery sequence	Actual delivery sequence	Actual delivery time
1	<b>2</b>	0:07:57
2	<b>1</b>	0:07:36
3	3	0:15:14
...	...	...
39	<b>40</b>	2:06:41
40	<b>39</b>	2:05:48
41	<b>42</b>	2:24:17
42	<b>41</b>	2:14:52
43	<b>45</b>	2:35:36
44	<b>43</b>	2:34:02
45	<b>44</b>	2:35:10
46	<b>47</b>	2:44:54
...	...	...
270	270	23:25:06

**Table 4.** Result of delivery sequence. Significant values are in [bold].



**Figure 20.** Convergence analysis of five algorithms. IGWO-1 ONLY incorporates the improvement defined by Eq. (34), while IGWO-2 employs Eq. (34) and Eq. (35) uniformly in both the global search and local search. (a) Fitness values. (b) Conflict counts. (c) Total working time of UFMCs.

ensuring safety without the need for detours. Out of a total of 270 guide tasks, 64.44% have undergone adjustments in the delivery sequence.

Sorties of takeoff and landing	Algorithm	Rate of conflict resolution[%]	Required generation finding a feasible solution set[s]	Total working time of UFMCS[s]	Actual number of iterations
270	PSO	<b>100</b>	80	183,192	80
	GWO	<b>100</b>	110	<b>167,471</b>	110
	<b>IGWO</b>	<b>100</b>	<b>65</b>	<b>170,463</b>	<b>65</b>
	IGWO-1	91.67	> 150	168,625	> 150
	IGWO-2	80	> 150	166,155	> 150
	WOA	<b>100</b>	<b>28</b>	176,057	<b>35</b>
	NNA	81.82	> 150	178,684	> 150
12	PSO	100	Feasible Initially	6,843.1	<b>2</b>
	<b>IGWO</b>	100	Feasible Initially	<b>6,744.9</b>	18
	IGWO-1	100	Feasible Initially	6,801.2	<b>10</b>
	IGWO-2	100	Feasible Initially	6,788.9	32
	WOA	100	Feasible Initially	6,851.2	17
	NNA	100	Feasible Initially	<b>6,734.3</b>	149
26	PSO	100	<b>Feasible Initially</b>	15,157	<b>2</b>
	<b>IGWO</b>	100	3	<b>13,366</b>	149
	IGWO-1	100	3	13,632	89
	IGWO-2	100	38	13,478	137
	WOA	100	4	<b>13,232</b>	<b>22</b>
	NNA	100	<b>2</b>	14,357	119
83	PSO	100	<b>5</b>	52,513	<b>5</b>
	<b>IGWO</b>	100	18	<b>43,867</b>	145
	IGWO-1	100	<b>5</b>	<b>44,507</b>	<b>107</b>
	IGWO-2	100	67	45,219	149
	WOA	100	<b>3</b>	48,575	110
	NNA	100	79	49,665	143
160	PSO	<b>100</b>	<b>28</b>	104,807	<b>28</b>
	<b>IGWO</b>	<b>100</b>	44	98,233	<b>44</b>
	IGWO-1	<b>100</b>	89	<b>96,215</b>	89
	IGWO-2	50	> 150	<b>96,531</b>	> 150
	WOA	<b>100</b>	<b>6</b>	97,369	70
	NNA	<b>100</b>	44	102,518	123

**Table 5.** Quantitative assessment for algorithms. In the same experiment group, the black bold comments mean advantageous indicators. Significant values are in [bold].

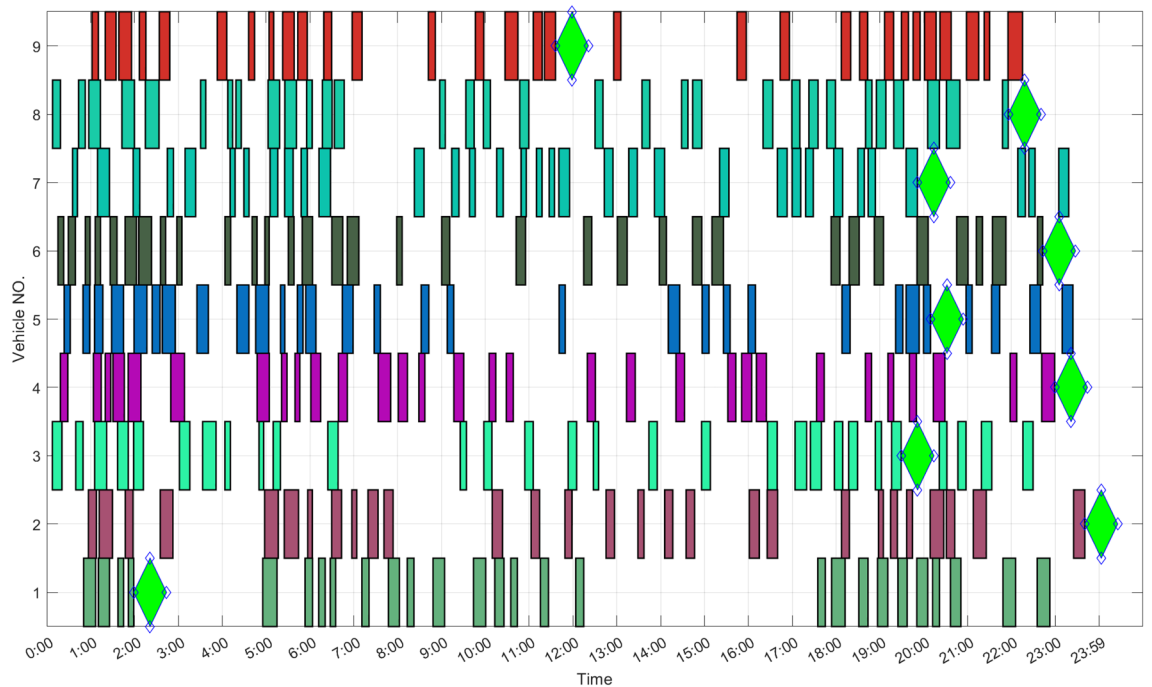
### Regards to IGWO Algorithm

To evaluate the convergence speed of the IGWO algorithm, seven different algorithms, including PSO, GWO, WOA<sup>30</sup>, NNA<sup>11,31</sup>, two variations of IGWO algorithm (IGWO-1 and IGWO-2), and the algorithm in this paper (IGWO), were employed in the case study. A population size of 20 and 150 iterations were set, the number of iterations is enough to make the function variation converge within 1%. The results are showed in Fig. 20.

IGWO-1, IGWO-2 and NNA failed to generate any feasible conflict-free solution set for the scale of 270 flights. On the other hand, the remaining algorithms successfully produced feasible solution sets. Among the feasible, GWO exhibited the highest quality but had relatively slower convergence speed. Although PSO and WOA generated the feasible with ideal iterations, their total working times of UFMCS significantly deviated from that of the other algorithms, indicating inefficient operational performance. In contrast, IGWO showed the fastest convergence speed.

Furthermore, in Table 5, quantitative analyses were conducted on four major indicators for handling the scales of 12, 26, 83, 160, and 270 flights per day. In general, IGWO has got the most advantageous indicators (as shown in bold black in the table). WOA is competitive in the convergence speed, but performs poorly in the working time. In contrast, NNA seems to be adapted to small-scale problem.

The performance of GWO, IGWO-1 and IGWO-2 showed improvements compared to the PSO algorithm mentioned in Reference<sup>23</sup>. In the case of 270 flights, the proposed IGWO algorithm achieved an 18.75% improvement in convergence speed, which is consistent with the qualitative analysis. Simultaneously, it incurred a relatively minor loss in operational efficiency, with only 1.76% more total working time compared to GWO. Comparative analysis across different scales revealed that IGWO exhibited strong generalization performance compared to the other algorithms. Meanwhile, IGWO-1 demonstrated potential advantages in solving medium-scale (83 and 160 flight movements) scheduling problems, showing promise for generating more efficient guidance trajectories.



**Figure 21.** Gantt chart for guidance tasks and charging decisions. The green diamond represents the occurrence of a charging event.

### Regards to vehicle assignment

According to the consultation notice<sup>32</sup>, the estimated number- $k$  of UFMCs in this case study is 9. The results for optimization of UPMC assignment are shown in Fig. 21. The results indicate that each UPMC is assigned 28 to 34 flight tasks, with a standard deviation of 582s in working time. The assignment inequality index is only 0.015731, achieving fair vehicle assignment. Each UPMC only needs to be charged once per day, which is consistent with the total working time of 170,463s mentioned in Section 5.5. When 9 UFMCs are required, the average working time per vehicle is 18,940s, less than half of the safe endurance ( $q$ ). In fact, for each vehicle, the charging process shown in Fig. 21 needs to be performed only every two days. During the redundant charging time and idle time of UFMCs, they can serve as backups for each other. Considering that Ezhou is a specialized cargo airport, the guidance tasks at night at this airport are relatively active, which is related to the characteristics of distribution of freight flight.

### Conclusion

In this study focused on the integrated optimization of scheduling for UFMCs on airport surface, we proposed the concept of GUs and a quantification method for guidance following spacing. We established an integrated UFMCs scheduling model, consisting of trajectory planning and vehicle assignment, considering the demands of precise connecting among guidance tasks and the vehicle charging. Additionally, considering the vehicle dynamics, we designed a high-precision method of guidance trajectory deduction and a conflict prediction model based on protected zones. We also developed an IGWO algorithm fitting the integer encoding.

Secondly, we validated the proposed approaches using the case of Huahu Airport in Ezhou. The IGWO demonstrated significant improvements in generating conflict-free feasible solution sets and generalization performance. The IGWO-1 algorithm showed potential in generating high-quality feasible solution sets for medium-scale optimization of scheduling for UFMCs. 64.44% and 25% of guidance tasks successfully avoided potential conflicts through fine-tuning timing and detour, sacrificing only 2.85% of efficiency to ensure safety. In the vehicle assignment module, the assignment inequality index was as low as 0.015731, indicating the achievement of collaborative planning and the achievement of balanced allocation of guidance tasks.

Finally, future research will explore tactical optimization strategies for flight delays or unexpected incidents encountered by UFMCs, aiming to enhance the robustness of algorithms and models.

### Data availability

The dataset used and analyzed in the current study is available from the corresponding author on reasonable.

Received: 27 November 2023; Accepted: 4 April 2024

Published online: 12 April 2024



## References

- CAAC. 2021. Roadmap for application of airport un-manned driving equipment (2021–2025); civil aviation order [2021] no. 46. Tech. Rep., Civil Aviation Administration of China, Beijing, China.
- Zong, F., He, Z., Zeng, M. & Yixuan, L. Dynamic lane changing trajectory planning for cav: A multi-agent model with path pre-planning. *Transportmetrica B* <https://doi.org/10.1080/21680566.2021.1989079> (2021).
- Morris, R. *et al.* Planning, scheduling and monitoring for airport surface operations. In *AAAI-16 Workshop on Planning for Hybrid Systems* (2016).
- Sirigu, G., Battipede, M., Clarke, J.-P. & Gili, P. A fleet management algorithm for automatic taxi operations. In *International Conference on Research in Air Transportation—Doctoral Symposium* (2016).
- van Oosterom, S. & Mitici, M. Analyzing the impact of battery capacity and charging protocols on the dispatchment of electric towing vehicles at a large airport. In *AIAA AVIATION 2022 Forum*, <https://doi.org/10.2514/6.2022-3920> (2022).
- He, D. (2018). *Research on Scheduling of Ground Service Vehicles for Large Airport Airside Transition—Illustrated by The Case of Ferry Bus*. Master's thesis, Beijing Jiao-tong University, Beijing, China.
- CAAC. 2022. Rules governing the administration of air traffic management for civil aviation; ccar-93tm-r6. Tech. Rep., Civil Aviation Administration of China.
- Wang, S., Che, Y., Zhao, H. & Lim, A. Accurate tracking, collision detection, and optimal scheduling of airport ground support equipment. *IEEE Internet Things J.* **PP**, 1–1. <https://doi.org/10.1109/JIOT.2020.3004874> (2020).
- Zhang, J., Chong, X., Wei, Y., Bi, Z. & Yu, Q. Optimization of apron support vehicle operation scheduling based on multi-layer coding genetic algorithm. *Appl. Sci.* **12**, 5279. <https://doi.org/10.3390/app12105279> (2022).
- Bao, D.-W., Zhou, J.-Y., Zhang, Z.-Q., Chen, Z. & Kang, D. Mixed fleet scheduling method for airport ground service vehicles under the trend of electrification. *J. Air Transp. Manag.* **108**, 102379. <https://doi.org/10.1016/j.jairtraman.2023.102379> (2023).
- Shang, D. *Research on Optimal Scheduling Method of the Special Electric Vehicle Working on Parking Apron*. Master's thesis, Civil Aviation University of China, Tianjin, China (2017).
- Cheung, A., Ip, W., Lu, D. & Lai, C. An aircraft service scheduling model using genetic algorithms. *J. Manuf. Technol. Manag.* **16**, 109–119. <https://doi.org/10.1108/17410380510574112> (2005).
- Bazargan, M. *Airline Operations and Scheduling* 2nd edn. (China Civil Aviation Publishing House, 2020).
- Diepen, G., Akker, J.M., Hoogeveen, J.A. & Smeltink, J.W. *Using column generation for gate planning at Amsterdam Airport Schiphol*. Tech. Rep., Department of Information and Computing Sciences, Utrecht University (2007).
- Kuhn, K. & Loth, S. Airport service vehicle scheduling. *Air Traffic Control Q.* **18**, 63–83. <https://doi.org/10.2514/atcq.18.1.63> (2010).
- Andreatta, G., Giovanni, L. & Monaci, M. A fast heuristic for airport ground-service equipment-and-staff allocation. *Proc. Soc. Behav. Sci.* <https://doi.org/10.1016/j.sbspro.2013.12.817> (2014).
- Zhang, T., Ding, M. & Zuo, H. Airport ground movement disruption recovery via mixed-integer programming. *J. Beijing Univ. Aeronaut. Astronaut.* **44**, 2356 (2018).
- Wang, Y., C. and Ren. A model of gate allocation for parallel multi-runway hybrid operation from the perspective of fuel-saving and carbon emission reduction. *J. Transp. Inform. Saf.* **39**: 144 (2021).
- Li, N. *et al.* An empirical study on low emission taxiing path optimization of aircrafts on airport surfaces from the perspective of reducing carbon emissions. *Energies* **12**, 1649. <https://doi.org/10.3390/en12091649> (2019).
- Liu, J., Tang, X. & Zhang, Y. E. A. Optimization of airborne guidance velocity profile for required time of arrival. *Aeronaut. Comput. Tech.* **51**, 525 (2021).
- Tang, Y., Hu, M. & Huang, R. E. A. Aircraft taxi routes planning based on free time windows and multi-agent for A-SMGCS. *Acta Aeronaut. Astronaut. Sinica* **36**, 1627 (2015).
- Ji, X., Xu, Y., Pu, Y., Hao, J. & Qin, W. Risk prediction model of passenger car following behavior under truck movement interruption of two-lane high-way in mountainous area. *J. Jilin Univ. Eng. Technol. Ed.* <https://doi.org/10.13229/j.cnki.jdxbgxb20220744> (2022).
- Sirigu, G., Clarke, J.-P., Battipede, M. & Gili, P. Hybrid particle swarm optimization with parameter fixing: Application to automatic taxi management. *J. Air Transp.* **28**, 1–13. <https://doi.org/10.2514/1.D0172> (2019).
- Zhang, T., Zhu, X., Li, J., Chen, H. & Li, Z. Research on conflict detection model for taxi-in process on the apron based on aircraft wingtip keypoint detection. *IET Intell. Transp. Syst.* **17**, 878. <https://doi.org/10.1049/itr2.12314> (2022).
- Li, H., Delage, E., Zhu, N., Pinedo, M. & Ma, S. Distributional robustness and inequity mitigation in disaster preparedness of humanitarian operations. *Manuf. Serv. Operations Manag.* <https://doi.org/10.1287/msom.2023.1230> (2023).
- Zhu, X., Tang, X. & Han, S. Aircraft initial taxiing route planning based on petri net and genetic algorithm. *J. Southwest Jiaotong Univ.* **48**, 565 (2013).
- Zhang, Y. & Zhou, X. Modified grey wolf optimization algorithm for global optimization problems. *J. Univ. Shanghai Sci. Technol.* <https://doi.org/10.13255/j.cnki.jusst.20200331002> (2021).
- Li, N., Hu, R., Qian, B., Jin, H. & Yu, N. Research on time-dependent vehicle routing problem with multiple time windows. *J. Syst. Simul.* **34**, 1775 <https://doi.org/10.16182/j.issn1004731x.joss.21-0244> (2022).
- Zhu, X., Xu, C., Qu, J. & Su, T. Design and simulation-based evaluation of taxiway operation scheme for multi-runway airport maneuvering area. *J. Syst. Simul.* <https://doi.org/10.16182/j.issn1004731x.joss.21-0789> (2022).
- Zhou, K. *et al.* Research on solving flexible job shop scheduling problem based on improved GWO algorithm SS-GWO. *Neural Process. Lett.* <https://doi.org/10.1007/s11063-024-11488-1> (2024).
- Sadollah, A., Sayyaadi, H. & Yadav, A. A dynamic metaheuristic optimization model inspired by biological nervous systems: Neural network algorithm. *Appl. Soft Comput.* **71**, 747. <https://doi.org/10.1016/j.asoc.2018.07.039> (2018).
- CAAC. 2015. Guidelines for the configuration of specialized equipment for flight support in civil transport airport. Tech. Rep. AC-139-CA-2015-01, Civil Aviation Administration of China, Beijing, China.

## Acknowledgements

This work is supported by the Central University Basic Research Fund of China [grant number ZHMH2022-008] and the National Key Research and Development Program of China [grant number 2022YFB2602004].

## Author contributions

D.Y.: Software, Experiment, Data processing, Writing - Original draft. X.Z.: Conceptualization, Methodology, Software. Y.Z.: Help writing. Q.Z.: Case of car-following, Idea.

## Competing interests

The authors declare no competing interests.

## Additional information

Correspondence and requests for materials should be addressed to X.Z.



**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2024