

Research Article

Open Access



# Cooperative search for moving targets with the ability to perceive and evade using multiple UAVs

Ziyi Wang, Wencheng Zou, Sheng Li

School of Automation, Nanjing University of Science and Technology, Nanjing 210094, Jiangsu, China.

**Correspondence to:** Ziyi Wang, School of Automation, Nanjing University of Science and Technology, No.200 Xiao Lingwei Street Xuanwu District, Nanjing 210094, Jiangsu, China. E-mail: mariowang@njust.edu.cn; ORCID: 0009-0004-2527-7409

**How to cite this article:** Wang Z, Guo J, Zou W, Li S. Cooperative search for moving targets with the ability to perceive and evade using multiple UAVs. *Intell Robot* 2023;3(4):538-64. <http://dx.doi.org/10.20517/ir.2023.30>

**Received:** 1 Aug 2023 **First Decision:** 29 Aug 2023 **Revised:** 8 Sep 2023 **Accepted:** 16 Oct 2023 **Published:** 28 Oct 2023

**Academic Editor:** Haibin Duan **Copy Editor:** Yanbin Bai **Production Editor:** Yanbin Bai

## Abstract

This paper focuses on the problem of regional cooperative search using multiple unmanned aerial vehicles (UAVs) for targets that have the ability to perceive and evade. When UAVs search for moving targets in a mission area, the targets can perceive the positions and flight direction of UAVs within certain limits and take corresponding evasive actions, which makes the search more challenging than traditional search problems. To address this problem, we first define a detailed motion model for such targets and design various search information maps and their update methods to describe the environmental information based on the prediction of moving targets and the search results of UAVs. We then establish a multi-UAV search path planning optimization model based on the model predictive control, which includes various newly designed objective functions of search benefits and costs. We propose a priority-encoded improved genetic algorithm with a fine-adjustment mechanism to solve this model. The simulation results show that the proposed method can effectively improve the cooperative search efficiency, and more targets can be found at a much faster rate compared to traditional search methods.

**Keywords:** Unmanned aerial vehicle (UAV), moving target search, path planning, fine-adjustment mechanism

## 1. INTRODUCTION

In recent years, with the continuous development of unmanned aerial vehicle (UAV) technology, the application of UAVs for searching civilian or military targets has been increasing. Path planning for target search



© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.



has become an important research direction<sup>[1,2]</sup>. In common search problems, target search can be divided into stationary target search and moving target search based on the motion ability of targets. In stationary target search, it is usually required to plan the UAVs' search path in advance to achieve coverage of the mission area<sup>[3-5]</sup>. However, since the target location may change at any time, UAVs need to update the search path in real time based on the search results to achieve real-time and dynamic target search, leading to the problem of moving target search.

Searching for moving targets typically involves constructing an environmental model with one or several search information maps. These maps usually include a probability map<sup>[6]</sup>, a certainty map (also known as uncertainty map)<sup>[7]</sup>, and a pheromone map<sup>[8]</sup>. UAVs then update these search information maps based on the dynamic search results, and the search path is planned based on the updated environmental information to improve the search efficiency and probability of discovering targets. There are several concerns raised about this problem in recent research.

Firstly, addressing the motion model of moving targets, Zhong *et al.*<sup>[9]</sup> used a Markov chain to represent a hidden movement of targets and predict the position of targets. Hu *et al.*<sup>[10]</sup> assumed a type of moving target that collaborates with fixed sensors providing UAV-sensing ability distributed in the mission area and proposed a method to predict the distribution of targets. Yue *et al.*<sup>[11]</sup> proposed a specialized search map for the problem of cooperative search in unknown sea areas, which achieves dynamic changes in target movement estimation by assuming a fixed probability diffusion coefficient matrix for probability.

Regarding the target perception method of UAVs, Li *et al.*<sup>[12]</sup> studied the use of knowledge distillation (KD) for target perception by presenting a comprehensive survey of KD-based object detection models developed in recent years and offered valuable perspectives on incorporating object detection into the target search strategy. Li *et al.*<sup>[13]</sup> provided a multi-modal perception method using the spatial-temporal graph obtained from videos to promote latent space alignment in unsupervised multi-modal machine translation (UMMT), which intersects with UAV perception capabilities.

On the topic of the encoding method of the search path, Shorakaei *et al.*<sup>[14]</sup> proposed a path planning method that considers obstacles or threat areas using a novel encoding method by a matrix, which uniformly considers all UAVs and the coordinates of their waypoint positions. Alanezi *et al.*<sup>[15]</sup> propose a motion-encoded genetic algorithm with multiple parents, which realizes a unified motion-encoding on a series of UAVs. For optimization, Luo *et al.*<sup>[16]</sup> used the fruit fly optimization algorithm to solve the search path, in which multiple fruit fly swarms are used to enhance the global search ability. They adopted different search strategies through a strategy switching method in the odor search and visual search stages, making the planning process more effective and stable. Similarly, other heuristic algorithms, such as differential evolution<sup>[17]</sup> and pigeon-inspired optimization<sup>[18]</sup>, have been used in search path planning.

To address the resource allocation problem, Fang *et al.*<sup>[19]</sup> provide several policies and optimization algorithms to find a near-optimal solution associated both with high age of information as well as high power consumption. Zhang *et al.*<sup>[20]</sup> investigated the reliable transmission scheme of downlink Non-Orthogonal Multiple Access (NOMA) systems and provided valuable insights into realizing reliable transmission using NOMA with randomly deployed receivers.

When searching for targets, the idea of model predictive control (MPC) is commonly used for long-term search. Zhou *et al.*<sup>[21]</sup> applied distributed MPC (DMPC) combined with digital pheromone maps to realize the path planning of regional cooperative search. The DMPC method based on Nash equilibrium can achieve global optimization by locally optimizing the newly designed performance indicators. Yao *et al.*<sup>[22]</sup> considered the communication network and information fusion between UAVs, designed a consensus algorithm with state

predictor based on the minimum spanning tree structure to realize the fusion of predicted target probability map, and proposed a future-dependent MPC framework to realize the cooperative trajectory optimization and obtain the optimal control input. In addition, deep reinforcement learning (DRL) is also a promising tool for solving such problems. Wei *et al.* [23] proposed a joint design of the unmanned aerial/surface/underwater vehicle (UAV-USV-UUV) network for cooperative underwater target hunting and conceived a novel deep Q-learning (DQN) algorithm to solve the target hunting problem. These research results provide several new perspectives to solve the target search problem.

Based on the studies mentioned above, this paper considers a multi-UAV regional cooperative search problem for targets with the ability to perceive and evade. The targets are equipped with a UAV-sensing device and move randomly in the mission area in a Markovian fashion. However, they can perceive the UAVs and take corresponding evasive actions by increasing the distance from the UAVs to achieve the ability of autonomous evasion, which increases the difficulty of target search.

This paper proposes a novel Cooperative Search Method for Targets with the ability to Perceive and Evade (CSMTPE). Firstly, we define the motion model of such targets in detail and design various search information maps together with their update methods based on the prediction of moving targets and the search results of UAVs. Secondly, we establish a multi-UAV search path planning optimization model based on MPC and design objective functions of search benefits and costs. Finally, we propose an improved genetic algorithm with a fine-adjustment mechanism (IGAFA) to solve this optimization model. The simulation results confirm the effectiveness of the proposed method.

Several contributions are made in this paper to the regional cooperative search problem for targets with the ability to perceive and evade:

- (i) A new motion model of moving targets and a detection model for UAVs are proposed, which are more consistent with real-world search scenarios compared to the traditional methods.
- (ii) The updated formula for the traditional probability map used to predict target probability is improved, enhancing UAV search efficiency when dealing with the evasive maneuvers of moving targets.
- (iii) A search information map that reflects the detection response of moving targets is introduced, providing real-time feedback to UAVs on their search results.
- (iv) A multi-UAV search path planning optimization model is established, and a series of objective functions are designed for this model.
- (v) A priority-encoding method for multi-UAV search paths and a priority-encoded IGAFA algorithm are proposed, effectively solving the optimization problem of multi-UAV search path planning effectively.

Overall, the effectiveness and efficiency of cooperative search for moving targets with the ability to perceive and evade are improved by these contributions, which could have practical applications in fields such as search, rescue, surveillance, and military operations.

The rest of this paper is structured as follows: Section 2 describes the search problem and presents a model for it. Section 3 defines the search information maps and their update methods. Section 4 introduced a multi-UAV cooperative search method and optimization model. Section 5 presents the encoding method, the IGAFA algorithm, and the complete search method. In Section 6, simulation experiments confirm the effectiveness of the proposed method. Finally, in Section 7, we present the conclusion.

## 2. PROBLEM FORMULATION

Assuming that there are  $N_U$  UAVs equipped with detection sensors searching for  $N_T$  moving targets with the ability to perceive and evade within a mission area  $E$ , where the width and height of  $E$  are  $L_x$  and  $L_y$ , and the

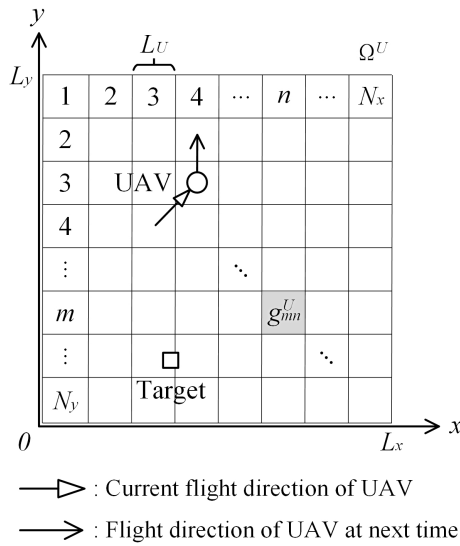


Figure 1. Definition of mission grid.

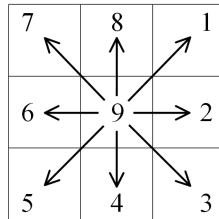


Figure 2. Definition of flight directions of UAV.

targets move freely in  $E$  and perceive and evade the UAVs that are searching for them. Based on this definition, a cooperative search task requires UAVs to discover and track all targets using a cooperative search method [24].

### 2.1. Motion model of UAVs

Mission area  $E$  is divided into  $N_x \times N_y$  grid cells with  $L_U$  as the side length of a single cell, which are used as mission grid or UAV motion grid and denoted as  $\Omega^U$ . It is agreed that the UAV has limited movement between mission grid cells, where the cell with  $m$ -th row and  $n$ -th column in  $\Omega^U$  is marked as  $g_{mn}^U$ . Figure 1 shows the specific representation of the mission grid.

Based on this definition, the multi-UAV system executing cooperative search tasks can be considered as a complete control system. The state variables of the UAVs at time  $t_k$  can be represented as  $X(t_k) = \{X_1(t_k), X_2(t_k), \dots, X_i(t_k), \dots, X_{N_U}(t_k)\}$ , where  $X_i(t_k), i = 1, 2, \dots, N_U$  represents the position at time  $t_k$  in  $\Omega^U$  of  $i$ -th UAV. Each UAV can move to the adjacent cells or stay in the current cell every motion cycle of  $\Delta T$  time. Figure 2 shows the definition of flight directions of UAVs.

The control input of  $i$ -th UAV at time  $t_k$  can be represented as  $u_i(t_k) \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ , where  $i = 1, 2, \dots, N_U$ ; flight direction "9" represents staying in the current cell, and flight directions 1-8 represent moving one step towards adjacent cells. Therefore, the control input of the multi-UAV system at time  $t_k$  can be represented as  $U(t_k) = \{u_1(t_k), u_2(t_k), \dots, u_{N_U}(t_k)\}$ , and the state model of the multi-UAV system can be recorded as  $X(t_{k+1}) = F(X(t_k), U(t_k))$ , where  $F(\cdot)$  is the state transition function of the system.

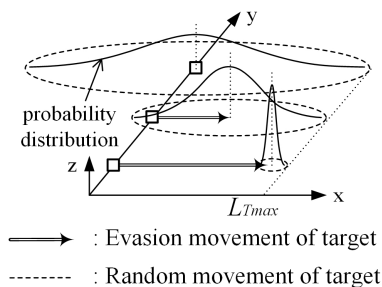


Figure 3. Combination of movement with different evasion intensities.

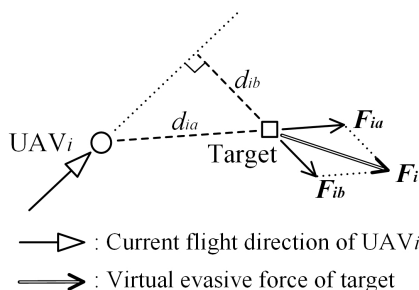


Figure 4. Virtual evasive force of moving targets.

### 2.2. Motion model of moving targets

As the motion law of the moving targets is unknown, it is assumed that the moving target has a maximum movement distance of  $L_{Tmax}$ , and the motion of each target can be represented by a Gauss-Markov motion model [9]. The movement of each target in every motion cycle can be considered as a combination of random movement and directional evasive movement. Figure 3 shows the movement of the moving target with different evasion intensities.

In order to effectively evade the search of UAVs, a virtual evasive force is defined in this paper, which includes two directions of evasion: away from the current position of the UAV and away from the forward path of the UAV. Figure 4 shows the virtual evasive force of a moving target when facing the search for  $i$ -th UAV.

In Figure 4,  $\mathbf{F}_{ia}$  represents the virtual evasive force A to keep away from the current position of  $i$ -th UAV,  $\mathbf{F}_{ib}$  represents the virtual evasive force B to keep away from the forward path of  $i$ -th UAV, and  $\mathbf{F}_i$  represents the comprehensive virtual evasive force of  $i$ -th UAV. Considering the distance from the target to the UAV, the virtual evasive force A is defined as:

$$\mathbf{F}_{ia} = A_a e^{-\frac{3d_{ia}}{\delta_a}} \cdot \mathbf{e}_{ia} \tag{1}$$

where  $\mathbf{e}_{ia}$  is the unit direction vector from  $i$ -th UAV to the target,  $A_a$  is the response amplitude of virtual evasive force A, and  $\delta_a$  is the corresponding straight response distance. Considering that the target cannot effectively determine the flight direction and threat level of the UAV at a long distance, it is reasonable to assume that the evasive force is negatively correlated with distance. The virtual evasive force B is defined as:

$$\mathbf{F}_{ib} = \begin{cases} A_b e^{-\left(\frac{3d_{ia}}{\delta_{b1}} + \frac{3d_{ib}}{\delta_{b2}}\right)} \cdot \mathbf{e}_{ib}, & \mathbf{e}_{ia}^T \mathbf{e}_i > 0 \\ \mathbf{0}, & \mathbf{e}_{ia}^T \mathbf{e}_i \leq 0 \end{cases} \quad (2)$$

where  $\mathbf{e}_i$  is the unit direction vector of the flight direction of  $i$ -th UAV,  $\mathbf{e}_{ib}$  is the unit direction vector pointing vertically to the target from the direction of  $\mathbf{e}_i$ ,  $A_b$  is the response amplitude of virtual evasive force B, and  $\delta_{b1}$  and  $\delta_{b2}$  represent the corresponding response distances in straight and vertical directions, respectively. This assumption reasonably estimates the target's evasive behavior toward the UAV's pursuit. The final evasive virtual force of the moving target can be represented as:

$$\mathbf{F} = \begin{bmatrix} F_x \\ F_y \end{bmatrix} = \sum_{i=1}^{N_U} \mathbf{F}_i = \sum_{i=1}^{N_U} \mathbf{F}_{ia} + \mathbf{F}_{ib} \quad (3)$$

In summary, based on the Gauss-Markov motion model, it is assumed that the motions in the x and y axes are independent of each other, the transition probability density function of the target from the position  $(x_k, y_k)$  to the position  $(x_{k+1}, y_{k+1})$  after one motion cycle  $\Delta T$  can be represented as:

$$P\{(x_{k+1}, y_{k+1})|(x_k, y_k)\} = \frac{1}{2\pi\sigma_{xy}^2} e^{-\frac{1}{2\sigma_{xy}^2} [(x_{k+1}-\mu_x)^2+(y_{k+1}-\mu_y)^2]} \quad (4)$$

In order to appropriately allocate the proportion of random movement and the directional evasive movement, the hyperbolic tangent (tanh) function is used to saturate the virtual evasive force of the target in this paper. Therefore, the means of  $\mu_x$  and  $\mu_y$  and variance of  $\sigma_{xy}^2$  can be defined as:

$$\begin{aligned} \omega_{sat} &= \tanh(\|\mathbf{F}\|) \\ \mu_x &= x_k + L_{Tmax} \cdot \frac{\omega_{sat} F_x}{\|\mathbf{F}\|} \\ \mu_y &= y_k + L_{Tmax} \cdot \frac{\omega_{sat} F_y}{\|\mathbf{F}\|} \\ 3\sigma_{xy} &= (1 - \omega_{sat}) \cdot L_{Tmax} \end{aligned} \quad (5)$$

where  $\omega_{sat} \in [0, 1]$  is the saturated partition coefficient of evasive movement. According to the common definition of the  $3\sigma$  rule,  $L_{Tmax}$  is appropriately allocated by the guidance of virtual evasive force, and consequently, the evasive maneuver of targets can be realized.

### 2.3. Detection model of UAV

The UAV detection range is defined as a square area with a side length of  $L_D$  centered on the UAV's current position, which can be represented as the mission grid cell and its adjacent cells where the UAV is located. Considering the real-time changes in UAV position, denote the detection grid of  $i$ -th UAV as  $\Omega^{Di}$ , which can be regarded as a certain range in  $\Omega^U$ , and its side length is  $\gamma_D$  times that of a single mission grid cell. Same as above, the detection grid cell of  $i$ -th UAV with  $m$ -th row and  $n$ -th column is marked as  $g_{mn}^{Di}$ . Figure 5 shows the specific representation of the detection grid of  $i$ -th UAV.

The  $i$ -th UAV can only detect the area within  $\Omega^{Di}$ . The detection is carried out in every single detection grid cell, and the detection probability corresponding to  $g_{mn}^{Di}$  is  $p_{mn}^D$ , which can be defined as a general form<sup>[25]</sup> as:

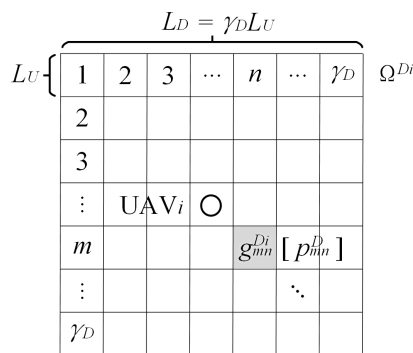


Figure 5. Definition of detection grid of *i*-th UAV.

$$p_{mn}^D = \begin{cases} p_{max}^D, & d_{mn} < \delta_{Dmin} \\ p_{min}^D + (p_{max}^D - p_{min}^D) \cdot \frac{\delta_{Dmax} - d_{mn}}{\delta_{Dmax} - \delta_{Dmin}}, & \delta_{Dmin} \leq d_{mn} < \delta_{Dmax} \\ p_{min}^D, & d_{mn} \geq \delta_{Dmax} \end{cases} \tag{6}$$

$$d_{mn} = L_U \sqrt{\left(m - \left\lceil \frac{\gamma_D}{2} \right\rceil\right)^2 + \left(n - \left\lceil \frac{\gamma_D}{2} \right\rceil\right)^2}$$

where  $p_{max}^D$  and  $p_{min}^D$  are the maximum and minimum detection probabilities, and  $\delta_{Dmin}$  and  $\delta_{Dmax}$  are their corresponding boundary detection distances. In order to comply with the actual detection situation, it is assumed that the probability of a false alarm occurring during the detection process is  $p^F$ ; that is, during a motion cycle  $\Delta T$ , UAV may generate a false alarm signal within the detection range with a probability of  $p^F$ .

Moreover, the additional conventions are given here:

- (i) To avoid collisions between UAVs, at most one UAV is allowed to search within each mission grid cell at any time instance.
- (ii) Communications between UAVs are ideal, and there is no communication delay.
- (iii) There is at most one target in a single mission grid cell.
- (iv) After discovering the target, the UAV will detach from the search sequence, enter a tracking state, and will not participate in subsequent search missions.

### 3. SEARCH INFORMATION MAP

When the mission area is discretized into grid cells, each search step of the UAV will be regarded as a search for each corresponding cell and give feedback on the results, and search information maps can be used to express the dynamic process of target search for each cell.

#### 3.1. Probability map

A probability map is a search information map that reflects the existence probability of targets. Considering the difference in speed between a UAV and a moving target, the mission grid  $\Omega^U$  is refined and divided as the region division of the probability map. After division, the mission area is divided into  $M_x \times M_y$  grid cells with  $L_T$  as the side length of a single grid cell where  $L_{Tmax} \leq L_T$ , and the side length of a mission grid cell is  $\gamma_T$  times of  $L_T$ . Denote the grid of probability map as  $\Omega^T$ , in which the cell with  $m$ -th row and  $n$ -th column is marked as  $g_{mn}^T$ . Figure 6 shows the specific representation of the grid definition of a probability map.

Denote the existence probability of the target in  $g_{mn}^T$  as  $p_{mn}$ . Considering that the detection is carried out in a

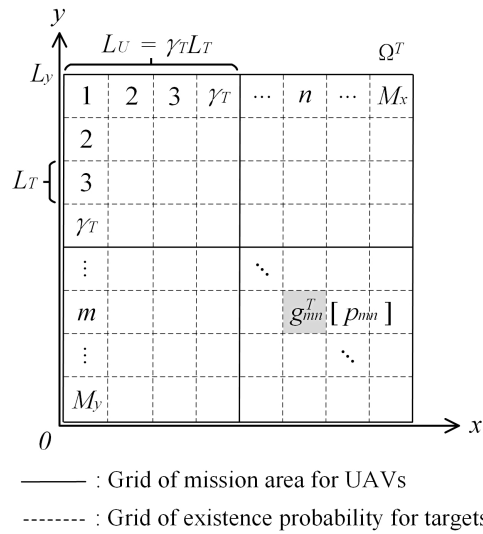


Figure 6. Grid definition of a probability map.

single detection grid cell, i.e., a mission grid cell, if the existence of a target in the mission grid cell with  $r$ -th row and  $c$ -th column is denoted as  $T_{rc}$ , then the probability of  $T_{rc}$  is:

$$P(T_{rc}) = 1 - \prod_{g_{mn}^T \in g_{rc}^U} [1 - p_{mn}(t_k)] \tag{7}$$

where  $g_{mn}^T \in g_{rc}^U$  denotes that  $g_{mn}^T$  is within the area of  $g_{rc}^U$  in terms of regional division, and the definition of such notations in the rest of the paper is the same as here. It can be determined that the target is found in  $g_{rc}^U$  when  $P(T_{rc})$  is not less than a certain threshold  $p_{thold}^D$ .

The transition of the existence probability based on target motion prediction under the influence of the movement of UAVs in any grid cell of the probability map can be represented by the probability diffusion coefficient of one cell to another. Denote the probability diffusion coefficient for transferring the existence probability of the target from  $g_{rc}^T$  to  $g_{mn}^T$  at time  $t_k$  as  $\Phi_{rc}^{mn}(t_k)$ , which can be expressed based on target motion prediction with Equations (4) and (5) as follows:

$$\Phi_{rc}^{mn}(t_k) = \int_{Y_r - \frac{L_T}{2}}^{Y_r + \frac{L_T}{2}} \int_{X_c - \frac{L_T}{2}}^{X_c + \frac{L_T}{2}} \int_{Y_m - \frac{L_T}{2}}^{Y_m + \frac{L_T}{2}} \int_{X_n - \frac{L_T}{2}}^{X_n + \frac{L_T}{2}} P\{(x_{k+1}, y_{k+1})|(x_k, y_k)\} dx_{k+1} dy_{k+1} dx_k dy_k \tag{8}$$

where  $(Y_r, X_c)$  and  $(Y_m, X_n)$  are the centers of  $g_{rc}^T$  and  $g_{mn}^T$ . Obviously, the sum of the probability diffusion coefficient from any grid cell to all cells remains constant at 1. According to the definition of maximum movement distance, it can be further constrained to have a sum of 1 from any grid cell to the adjacent cells and itself:

$$\sum_{m=1}^{M_y} \sum_{n=1}^{M_x} \Phi_{rc}^{mn}(t_k) = \sum_{m=r-1}^{r+1} \sum_{n=c-1}^{c+1} \Phi_{rc}^{mn}(t_k) = 1 \tag{9}$$



Based on the definition above, the updated existence probability of the target based on target motion prediction in  $g_{mn}^T$  after time  $t_k$  is denoted as  $p_{mn}^-(t_{k+1})$ , which is as follows:

$$p_{mn}^-(t_{k+1}) = \sum_{r=m-1}^{m+1} \sum_{c=n-1}^{n+1} \Phi_{rc}^{mn}(t_k) \cdot p_{rc}(t_k) \tag{10}$$

After the update based on target motion prediction, the probability map will be updated again based on the real-time detection results of airborne detection sensors of every UAV. In this paper, Bayesian criteria are used to dynamically update the probability map [8]. The update of existence probability after time  $t_k$  can be classified into the following three situations:

- (i)  $i$ -th UAV finds a target in  $g_{rc}^{Di}(t_{k+1})$  with  $g_{mn}^T \in g_{rc}^{Di}(t_{k+1})$ :

$$p_{mn}(t_{k+1}) = \frac{p_{rc}^D \cdot p_{mn}^-(t_{k+1})}{(p_{rc}^D - p^F) \cdot p_{mn}^-(t_{k+1}) + p^F} \tag{11}$$

- (ii)  $i$ -th UAV finds nothing in  $g_{rc}^{Di}(t_{k+1})$  with  $g_{mn}^T \in g_{rc}^{Di}(t_{k+1})$ :

$$p_{mn}(t_{k+1}) = \frac{(1 - p_{rc}^D) \cdot p_{mn}^-(t_{k+1})}{1 + (p^F - p_{rc}^D) \cdot p_{mn}^-(t_{k+1}) - p^F} \tag{12}$$

- (iii)  $g_{mn}^T$  is outside the area of the detection range of any UAV:

$$p_{mn}(t_{k+1}) = p_{mn}^-(t_{k+1}) \tag{13}$$

### 3.2. Certainty map

A certainty map is a search information map that reflects the degree of UAV exploration of the mission area. Considering that UAVs move in mission grid cells, the region division of the certainty map is the same as  $\Omega^U$ , and the certainty map varies with the UAVs' detectability and access situation. Denote the grid of a certainty map as  $\Omega^H$ , in which the cell with  $m$ -th row and  $n$ -th column is denoted as  $g_{mn}^H$  and the corresponding certainty is denoted as  $\chi_{mn}$ . The update of a certainty map after time  $t_k$  can be classified into the following two situations based on the access and the detection range of every UAV:

- (i)  $i$ -th UAV flies past with the situation that  $g_{mn}^H$  has the same regional definition as  $g_{rc}^{Di}(t_{k+1})$  in  $\Omega^{Di}(t_{k+1})$ :

$$\chi_{mn}(t_{k+1}) = \chi_{mn}(t_k) + p_{rc}^D \cdot [1 - \chi_{mn}(t_k)] \tag{14}$$

As in Equation (14), the certainty of the grid cell increases with the detectability and access of UAVs.

- (ii)  $g_{mn}^H$  is outside the area of the detection range of any UAV:

$$\chi_{mn}(t_{k+1}) = \eta_H \cdot \chi_{mn}(t_k) \tag{15}$$

where  $\eta_H \in (0, 1)$  is the attenuation coefficient of certainty.

### 3.3. Detection response map

A detection response map is a search information map defined in this paper to provide feedback on the target detection results of UAVs, which solves the problem of the lack of direct search results in conventional search information maps. The region division of the detection response map is the same as  $\Omega^U$ , and the detection

response map varies with the UAVs' detection results. Denote the grid of detection response map as  $\Omega^R$ , in which the cell with  $m$ -th row and  $n$ -th column is denoted as  $g_{mn}^R$  and the corresponding response value is denoted as  $\psi_{mn}$ . The update of the detection response map after time  $t_k$  can be classified into the following three situations based on the detection results of every UAV:

- (i)  $i$ -th UAV finds a target in  $g_{rc}^{Di}(t_{k+1})$  with the situation that  $g_{mn}^R$  has the same regional definition as  $g_{rc}^{Di}(t_{k+1})$  in  $\Omega^{Di}(t_{k+1})$ :

$$\psi_{mn}(t_{k+1}) = \psi_{mn}(t_k) + (p_{rc}^D)^{\eta_{R1}} \cdot [1 - \psi_{mn}(t_k)] \quad (16)$$

where  $\eta_{R1} \in (0, 1)$  is the sensitivity coefficient of detection response. In order to avoid missed detections, the response value is sensitive to detected targets and insensitive to undetected targets. The increase in response value is related to the detection probability of UAVs.

- (ii)  $i$ -th UAV finds nothing in  $g_{rc}^{Di}(t_{k+1})$  with the same situation above:

$$\psi_{mn}(t_{k+1}) = \left[1 - (p_{rc}^D)^{\frac{1}{\eta_{R1}}}\right] \cdot \psi_{mn}(t_k) \quad (17)$$

- (iii)  $g_{mn}^R$  is outside the area of the detection range of any UAV:

$$\psi_{mn}(t_{k+1}) = \eta_{R2} \cdot \psi_{mn}(t_k) \quad (18)$$

where  $\eta_{R2} \in (0, 1)$  is the attenuation coefficient of response value.

## 4. COOPERATIVE SEARCH PATH PLANNING

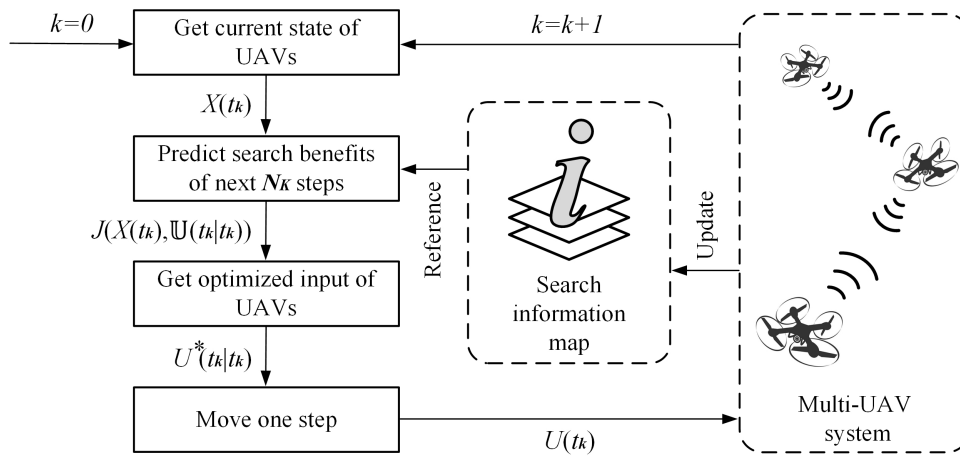
In order to improve the path planning efficiency of the search problem, a cooperative search method CSMTPE based on MPC is designed with the basic idea of time-domain rolling optimization, which transforms the large time-domain search problem into a continuous short-term path planning problem and guides the UAVs to search and track targets faster and more effectively<sup>[26]</sup>.

### 4.1. MPC method

According to the idea of MPC, by predicting and evaluating the search behavior in a limited period of time under different search decisions, the control input required by the current UAV swarm can be determined based on the current environment information. Figure 7 shows the cooperative search decision-making process for multi-UAV systems based on the idea of MPC.

It is assumed that an optimization process will consider the cooperative search process of  $N_K$  steps in future and evaluate the search benefits associated with different control inputs. Denote the search decision set of certain control inputs in future for multi-UAV systems at time  $t_k$  as  $\mathbb{U}(t_k|t_k) = \{\mathbb{U}_1(t_k|t_k), \mathbb{U}_2(t_k|t_k), \dots, \mathbb{U}_i(t_k|t_k), \dots, \mathbb{U}_{N_U}(t_k|t_k)\}$ ,  $1 \leq i \leq N_U$ , in which  $\mathbb{U}_i(t_k|t_k) = \{u_i(t_k|t_k), u_i(t_{k+1}|t_k), \dots, u_i(t_{k+N_K-1}|t_k)\}$  is the set of the control input sequence in future for  $i$ -th UAV and  $u_i(t_{k+q}|t_k)$  is the corresponding control input at time  $t_{k+q}$ .

Obviously, the quantitative search effectiveness of the search path can be uniquely obtained through an objective function  $J(\cdot)$ , which is based on the state  $X(t_k)$  and search decision set  $\mathbb{U}(t_k|t_k)$  of the multi-UAV system at time  $t_k$ , and the prediction evaluation process of the search can be achieved. Finally, the optimal search decision set  $\mathbb{U}^*(t_k|t_k)$  can be obtained by the optimization model of MPC as follows:



**Figure 7.** Cooperative search decision-making process for multi-UAV systems based on the idea of MPC.

$$\begin{aligned}
 \mathbb{U}^*(t_k|t_k) &= \{\mathbb{U}_1^*(t_k|t_k), \mathbb{U}_2^*(t_k|t_k), \dots, \mathbb{U}_{N_U}^*(t_k|t_k)\} \\
 &= \arg \min_{\mathbb{U}(t_k|t_k)} J(X(t_k), \mathbb{U}(t_k|t_k)) \\
 \text{s.t.} & \\
 &\begin{cases} X(t_{k+q+1}|t_k) = F(X(t_{k+q}|t_k), U(t_{k+q}|t_k)) \\ U(t_{k+q}|t_k) = \{u_1(t_{k+q}|t_k), u_2(t_{k+q}|t_k), \dots, u_{N_U}(t_{k+q}|t_k)\} \\ X(t_k|t_k) = X(t_k) \\ q = 0, 1, \dots, N_K - 1 \end{cases}
 \end{aligned} \tag{19}$$

where  $\mathbb{U}_i^*(t_k|t_k) = \{u_i^*(t_k|t_k), u_i^*(t_{k+1}|t_k), \dots, u_i^*(t_{k+N_K-1}|t_k)\}$  is the optimal set of the control input sequence in future for  $i$ -th UAV, and  $u_i^*(t_{k+q}|t_k)$  is the corresponding optimal control input at time  $t_{k+q}$ . Finally, by using the first term of the optimal decision set as the system optimal control input  $U^*(t_k|t_k) = \{u_1^*(t_k|t_k), u_2^*(t_k|t_k), \dots, u_{N_U}^*(t_k|t_k)\}$  at time  $t_k$  and guiding the UAV swarm to perform a one-step search, the cooperative search decision-making process for the current step can be completed.

### 4.2. Objective function

Considering that no actual search actions occurred during the estimation of the search results in prediction, the virtual search process of future steps is based on the premise that no targets are found in each step.

#### 4.2.1. Index of search effectiveness

The index of search effectiveness  $J_E$  reflects the focus on the grid cells with a high existence probability of the target, which is defined as:

$$J_E(t_k) = \frac{1}{N_U} \sum_{i=1}^{N_U} \sum_{q=1}^{N_K} \left\{ \omega_{IC}^q \cdot \prod_{g_{rc}^{Di} \in \Omega^{Di}(t_{k+q}|t_k)} \prod_{g_{mn}^T \in g_{rc}^{Di}} [1 - p_{mn}^-(t_{k+q}|t_k) \cdot p_{rc}^D] \right\} \tag{20}$$

where  $p_{mn}^-(t_{k+q}|t_k)$  is the estimation of target probability before the detection occurs at time  $t_{k+q}$  during the virtual search process with the updating method, as shown in Equation (10).

In addition,  $\omega_{IC}^q \in (0, 1)$  is the influence coefficient of the step  $q$  in future. In order to balance search effec-

tiveness with time,  $\omega_{IC}^q$  will decrease with the search step. Therefore, the definition of  $\omega_{IC}^q$  is as follows:

$$\omega_{IC}^q = e^{-\frac{2q}{N_K}} \quad (21)$$

#### 4.2.2. Index of uncertainty

The index of uncertainty  $J_H$  reflects the focus on the grid cells with a high existence probability of the target and not being visited for a long time, which is defined as:

$$J_H(t_k) = \frac{1}{N_U} \sum_{i=1}^{N_U} \sum_{q=1}^{N_K} \left\{ \omega_{IC}^q \cdot \prod_{g_{rc}^H \in \Omega^{D^i}(t_{k+q}|t_k)} \prod_{g_{mn}^T \in g_{rc}^H} [1 - p_{mn}^-(t_{k+q}|t_k) \cdot (1 - \chi_{rc}^-(t_{k+q}|t_k))] \right\} \quad (22)$$

where  $\chi_{rc}^-(t_{k+q}|t_k)$  is the estimation of certainty before the detection occurs at time  $t_{k+q}$  during virtual search processes, and it is equivalent to the certainty of the previous step with the updating method, as shown in Equations (14) and (15).

#### 4.2.3. Index of detection response

The index of detection response  $J_R$  reflects the focus on the grid cells with previously discovered targets, which is defined as:

$$J_R(t_k) = \frac{1}{N_U} \sum_{i=1}^{N_U} \sum_{q=1}^{N_K} \left\{ \omega_{IC}^q \cdot \prod_{g_{mn}^R = g_{rc}^{D^i} \in \Omega^{D^i}(t_{k+q}|t_k)} [1 - \psi_{mn}^-(t_{k+q}|t_k) \cdot p_{rc}^D] \right\} \quad (23)$$

where  $\psi_{mn}^-(t_{k+q}|t_k)$  is the estimation of response value before the detection occurs at time  $t_{k+q}$  during virtual search processes, and it is equivalent to the response value of the previous step with the updating method, as defined by Equations (17) and (18).

#### 4.2.4. Index of motion cost

The index of motion cost  $J_C$  reflects the penalty for UAV motion cost, which is defined as:

$$J_C(t_k) = \frac{1}{N_U} \sum_{i=1}^{N_U} \left\{ \omega_{IC}^1 \cdot \mathbf{C}(u_i(t_{k-1}), u_i(t_k|t_k)) + \sum_{q=1}^{N_K-1} \left[ \omega_{IC}^{q+1} \cdot \mathbf{C}(u_i(t_{k+q-1}|t_k), u_i(t_{k+q}|t_k)) \right] \right\} \quad (24)$$

where  $\mathbf{C} = \{c_{mn}\}$  is the motion cost matrix, in which  $c_{mn}$  represents the UAV motion cost with the previous flight direction  $m$  and the next flight direction  $n$ .

Based on the indices above, the objective function in Equation (19) is defined as follows:

$$J(X(t_k), \mathbb{U}(t_k|t_k)) = \lambda_E J_E(t_k) + \lambda_H J_H(t_k) + \lambda_R J_R(t_k) + \lambda_C J_C(t_k) \quad (25)$$

where  $\lambda_E$ ,  $\lambda_H$ ,  $\lambda_R$ , and  $\lambda_C$  are the weighting coefficients of each index.

## 5. SOLUTION ALGORITHM

It is the key to obtaining the optimal control input in the current step among numerous control decision sets for CSMTPE; therefore, this problem can be transformed into a standard nonlinear optimization problem. Among the normal solutions, the genetic algorithm (GA) is a general optimization algorithm with strong adaptability<sup>[27]</sup>. GA is designed and proposed according to the evolution law of organisms in nature. It is a computational model simulating the natural selection and genetic mechanism of Darwin's biological evolution theory and is a method of searching for the optimal solution by simulating the natural evolution process.

In this section, a priority-encoding method for CSMTPE is proposed, and a fine-adjustment mechanism based on this encoding method is designed. Finally, this mechanism is applied to various chromosome operations of traditional GA, forming a priority-encoded IGAGA algorithm to solve the problem above.

### 5.1. Priority-encoded IGAGA algorithm

#### 5.1.1. Priority-encoding method

Considering the potential issue of chromosomes being sensitive to small changes and having a high probability of illegitimate offspring caused by traditional direct encoding method, a priority list of the indices of mission grid cells is used as an indirect encoding method in this paper. The chromosome is encoded as  $P = \{p_1, p_2, \dots, p_i, \dots, p_{N_L}\}$  where  $p_i, i = 1, 2, \dots, N_L$  represents the index of destination cell in mission grid and  $N_L$  is the length of the priority list. The UAVs sequentially obtain their current destination cells from the priority list, and similarly obtain the next destination cells after reaching the current ones until the end of the search steps. In order to move to the destination cell, a UAV will move one cell toward the nearest, unoccupied, and reachable cell in the direction of the destination cell in each search step. Due to the fixed search step size, denoted as  $N_K$ , the probability of activated genes in chromosomes decreases with the shift of chromosomal location of genes, and the chromosome with such characteristic is defined as the priority-encoded chromosome in this paper.

Firstly, this encoding method can solve the problem of illegal offspring chromosomes, ensuring the efficiency of the iterative process by complying with the UAV moving rules. Secondly, changing a single gene will not alter or only slightly alter the search path of the destination cells before and after that gene, reducing the sensitivity of the chromosome while ensuring the randomness of the search path; in other words, the priority-encoded path will be more conducive to focusing on the long-term search benefits, rather than the short-term benefits of each move.

#### 5.1.2. Fine-adjustment mechanism

A fine-adjustment mechanism is proposed to improve the efficiency of evolution in traditional GA in this paper, which mainly optimizes the stochastic process in the evolution to fine-adjust the evolution direction. The fine-adjusted localization and mutation of genes are included in the specific applications as follows:

(i) Fine-adjusted localization of genes

Considering the problem of uneven distribution of key genes in priority-encoded chromosomes, a random probability density function for selecting genes with the iteration step  $t$  is designed as follows:

$$f_t(x) = \frac{c_t}{1 - e^{-c_t}} e^{-c_t \cdot x} \quad (26)$$

where  $x \in [0, 1)$  is the random variable, and  $c_t$  is the distribution influence factor that changes with the iteration step  $t$ , which is defined as:

$$c_t = A_{IT} \cdot e^{B_{IT} \cdot \left(\frac{t}{N_{IT}} - 1\right)} \quad (27)$$

where  $A_{IT}$  is the amplification factor of iteration that determines the selection probability of selecting the head genes in the later iteration stage.  $B_{IT}$  is the evolution factor of iteration that determines the rate of the evolution of probability distribution in iteration.  $N_{IT}$  is the maximum iteration step.

At the early stage of the iteration, the probability density function will show the characteristics of approximate uniform distribution. As the iteration process proceeds, the probability density function will gradually evolve from the uniform distribution to the exponential distribution. By using the probability density function above, a random number with a specific distribution pattern can be achieved as follows:

$$y_t = F_t^{-1}(\xi) = -\frac{1}{c_t} \ln[1 - (1 - e^{-c_t}) \cdot \xi] \quad (28)$$

where  $\xi \sim U(0, 1)$  is a uniformly distributed random number within  $[0, 1)$ , and  $y_t$  is the corresponding random number with the specific distribution pattern. In summary, the location of the selected gene can be represented as:

$$r_t = \lfloor N_L \cdot y_t \rfloor + 1 \quad (29)$$

By using the method above, a gene can be uniformly selected in the early stage of the iteration, and gradually evolved in the later stage to achieve precise optimization of the higher priority part for a better optimization process.

(ii) Fine-adjusted mutation of gene

In order to make the mutation operation in traditional GA more purposeful and enable the overall population to effectively jump out of the local optimal solution during the iteration process, a weight matrix  $\mathbf{W} = \{w_{mn}\}$  is designed to represent the optimizing benefits of candidate genes at different locations, where  $w_{mn}$  represents the optimizing benefits of  $m$ -th candidate gene for  $p_n$ . The higher the optimizing benefit, the more saturated the corresponding gene is in the dominant chromosome with the same location.

$\mathbf{W}$  is defined as a Zero matrix at the beginning of the iteration, which means that the optimization benefits are unknown. The adjustment of each element in  $\mathbf{W}$  in the iteration step  $t$  is as follows:

$$w_{mn}(t+1) = w_{mn}(t) + \frac{\eta_W}{N'_P} \sum_{i=1}^{N'_P} \Delta w_{mn}^i(t) \quad (30)$$

where  $\eta_W$  is the learning factor of  $\mathbf{W}$ ,  $N'_P$  is the number of all chromosomes involved in matrix adjustment, and  $\Delta w_{mn}^i \in [0, 1)$  represents the increased value of the optimization benefit of  $m$ -th candidate gene for  $p_n$  of  $i$ -th chromosome. If the  $i$ -th chromosome is denoted as  $P_i = \{p_{i,1}, p_{i,2}, \dots, p_{i,N_L}\}$ , the corresponding adjustment value is defined as follows:

$$\Delta w_{mn}^i(t) = \begin{cases} 0, & m \neq p_{i,n} \\ \tanh\left(\frac{\max\{J\} - J_i}{\max\{J\} - \min\{J\}}\right), & m = p_{i,n} \end{cases} \quad (31)$$

where  $J_i$  is the fitness of  $i$ -th chromosome,  $\max\{J\}$  and  $\min\{J\}$  are the maximum and minimum fitness of all chromosomes. Finally, the probability distribution of candidate genes for  $p_n$  is denoted as  $\mathbf{Prob}_n$ , and it is defined as:

$$\mathbf{Prob}_n = \text{softmax}(-\mathbf{w}_n) \tag{32}$$

where  $\mathbf{w}_n$  is the  $n$ -th column of  $\mathbf{W}$ , and the softmax function is used to normalize the probabilities based on optimizing benefits. The method above will be more purposeful to generate other feasible solutions that are different from the local optimal solution during the iteration.

### 5.1.3. Improved chromosome operations

The fine-adjustment mechanism is used in the chromosome operations of traditional GA in IGAGA. Considering the characteristics of the priority-encoding method, the traditional crossover is divided into two types: sexual and asexual crossover. The specific chromosome-operations are as follows:

(i) Sexual crossover

Sexual crossover refers to the crossover with two chromosomes as parents, which is similar to the traditional crossover in GA. Firstly, two random chromosomes are selected as parents, the chromosomal location of the start of crossover  $r$  is selected by Equation (29) with a random length  $l$ , and then the selected gene segments are replaced with each other.

$$\begin{aligned} \text{child}_1(p_i) &= \begin{cases} \text{parent}_1(p_i), & r \leq i < r + l \\ \text{parent}_2(p_i), & \text{others} \end{cases} \\ \text{child}_2(p_i) &= \begin{cases} \text{parent}_2(p_i), & r \leq i < r + l \\ \text{parent}_1(p_i), & \text{others} \end{cases} \end{aligned} \tag{33}$$

$$i = 1, 2, \dots, N_L$$

(ii) Asexual crossover

Asexual crossover refers to the crossover with only one chromosome as a parent, which is used to adjust the access priority or location of destination cells. There are four crossover operators defined as follows:

- a. Swap: Select two different genes  $r_a$  and  $r_b$  by Equation (29) and swap these genes to exchange the priority.

$$\text{child}(p_i) = \begin{cases} \text{parent}(p_{r_b}), & i = r_a \\ \text{parent}(p_{r_a}), & i = r_b \\ \text{parent}(p_i), & \text{others} \end{cases} \tag{34}$$

$$i = 1, 2, \dots, N_L$$

- b. Insert: Select two different genes  $r_a$  and  $r_b$ , by Equation (29), then insert the first gene into the location of the second gene to raise or lower the priority.

$$\text{child}(p_i) = \begin{cases} \text{parent}(p_{r_a}), & i = r_b \\ \text{parent}(p_{i+1}), & r_a < r_b \text{ and } r_a \leq i < r_b \\ \text{parent}(p_{i-1}), & r_a > r_b \text{ and } r_b < i \leq r_a \\ \text{parent}(p_i), & \text{others} \end{cases} \tag{35}$$

$$i = 1, 2, \dots, N_L$$

- c. Delete: Select one gene  $r$  by Equation (29) and move it to the end of the chromosome to give up the access of it.

$$\text{child}(p_i) = \begin{cases} \text{parent}(p_r), & i = N_L \\ \text{parent}(p_i), & i < r \\ \text{parent}(p_{i+1}), & \text{others} \end{cases} \quad (36)$$

$$i = 1, 2, \dots, N_L$$

- d. Shift: Select two different genes,  $r_a$  and  $r_b$ , by Equation (29), then shift the destination cell of the first gene towards that of the second gene by one cell to adjust the location directionally.

$$\text{child}(p_i) = \begin{cases} \text{shift}(\text{parent}(p_{r_a}), \text{parent}(p_{r_b})), & i = r_a \\ \text{parent}(p_i), & \text{others} \end{cases} \quad (37)$$

$$i = 1, 2, \dots, N_L$$

where the shift function has the definition as follows:

$$\text{shift}(p_a, p_b) = p_a + \text{sgn}((p_b - 1 \bmod N_x) - (p_a - 1 \bmod N_x)) + N_x \cdot \text{sgn}\left(\left\lfloor \frac{p_b - 1}{N_x} \right\rfloor - \left\lfloor \frac{p_a - 1}{N_x} \right\rfloor\right) \quad (38)$$

(iii) Mutation

Mutation is mainly used to attempt to access a new destination cell to increase the adequacy of access to the solution space in this problem. In advantageous circumstances, UAVs may approach new cells with high search benefits in the new search path. In this algorithm, the mutated gene is selected by Equation (29) and the mutation result is selected based on the probability distribution in Equation (32).

(iv) Selection

Similar to the traditional selection in GA, a binary tournament selection method is used to select the chromosomes in the population.

#### 5.1.4. Procedures of the priority-encoded IGAFa

The procedures of the priority-encoded IGAFa, i.e., the procedures to obtain the optimal set of control input sequences, are as shown in Table 1.

## 5.2. Complete search process

Based on the previous sections, the procedures of a complete search process for moving targets using CSMTPE, are as shown in Table 2.

## 6. RESULTS

Simulation experiments for the multi-UAV regional cooperative search problem in this paper are performed in Matlab. The computer configuration is as follows: Windows 10, CPU AMD(R) Ryzen(R) 7 4800HS of 2.9GHz up to 4.2GHz, Memory 16GB of 3200MHz.

The experiment assumes a military scenario where UAVs are used to search for targets of concealed enemy soldiers in a mission area. The UAVs search for enemies through airborne human detection equipment, while the enemies actively evade the UAVs' search. Set the size of the mission area as  $L_x = L_y = 4km$  and divide the area into  $40 \times 40$  grid cells with  $L_U = 100m$ . Assuming that the UAV moves at a speed of  $15m/s$  and the target moves at a maximum speed of  $6m/s$ , set the motion configuration for moving targets to be:  $L_{Tmax} = 40m$ ,  $A_a = 1$ ,  $\delta_a = 600m$ ,  $A_b = 0.8$ ,  $\delta_{b1} = 500m$ , and  $\delta_{b2} = 300m$ , which implies  $L_T = 50m$  and  $\gamma_T = 2$ . Set the



**Table 1. Procedures of the priority-encoded IGAFa**

<b>Name:</b>	<b>Priority-encoded IGAFa</b>
<b>Goal:</b>	<b>Obtain the optimal set of control input sequence</b>

---

- 1: Input: probability map  $\Omega^T(t_k)$ , certainty map  $\Omega^H(t_k)$ , detection response map  $\Omega^R(t_k)$ , multi-UAV system state  $X(t_k)$ , maximum iteration step  $N_{IT}$ , population size  $N_P$ , probability of sexual crossover  $P_{c1}$ , probability of asexual crossover  $P_{c2}$ , probability of mutation  $P_m$ ;
- 2: Output: the optimal set of control input sequence  $\mathbb{U}^*(t_k|t_k)$ ;
- 3: Generate  $N_P$  random chromosomes as current population  $G$ , let parent population  $P = G$ , child population  $F = \emptyset$ , minimum fitness  $J_{min} = \text{inf}$ , weight matrix  $\mathbf{W} = \mathbf{0}$ ;
- 4: for  $t=1:N_{IT}$
- 5:     From  $P$ , generate  $2 \cdot \lceil N_P \cdot P_{c1}/2 \rceil$  children by sexual crossover and move them into  $F$ ;
- 6:     From  $P$ , generate  $\lceil N_P \cdot P_{c2} \rceil$  children by asexual crossover and move them into  $F$ ;
- 7:     From  $P$ , generate  $\lceil N_P \cdot P_m \rceil$  children by mutation and move them into  $F$ ;
- 8:      $G \leftarrow P + F$ ;     //Let all parents and children be the current population
- 9:     for  $i=1:\text{size}(G)$
- 10:         if the fitness of  $i$ -th chromosome is not calculated
- 11:             Decode  $i$ -th chromosome to a set of control input sequence  $\mathbb{U}(t_k|t_k)$ ;
- 12:             Calculate the fitness  $J(X(t_k), \mathbb{U}(t_k|t_k))$  by Equation (25);
- 13:             if  $J(X(t_k), \mathbb{U}(t_k|t_k)) < J_{min}$
- 14:                  $J_{min} \leftarrow J(X(t_k), \mathbb{U}(t_k|t_k))$ ;     //Note the minimum fitness
- 15:                  $\mathbb{U}^*(t_k|t_k) \leftarrow \mathbb{U}(t_k|t_k)$ ;     //Note the corresponding control input sequence
- 16:             end
- 17:         end
- 18:     end
- 19:     Update weight matrix  $\mathbf{W}$  by Equation (30);
- 20:      $P \leftarrow \emptyset, F \leftarrow \emptyset$ ;     //Clear the parent and child population
- 21:     From  $G$ , select  $N_P$  chromosomes by "binary tournament selection" and move them into  $P$ ;
- 22:     end
- 23:     return  $\mathbb{U}^*(t_k|t_k)$ ;

---

**Table 2. Procedures of a complete search process using CSMTPE**

<b>Name:</b>	<b>Complete search process using CSMTPE</b>
<b>Goal:</b>	<b>Find all moving targets</b>

---

- 1: Initialize multi-UAV system state  $X(0)$  and probability map  $\Omega^T(0)$ , set search step  $k = 0$ , certainty map  $\Omega^H(0) = \mathbf{0}$  and detection response map  $\Omega^R(0) = \mathbf{0}$ ;
- 2: while not all targets are found
- 3:     Get the optimal set of control input sequence  $\mathbb{U}^*(t_k|t_k)$  by IGAFa with the procedures in Table 1;
- 4:     From  $\mathbb{U}^*(t_k|t_k)$  get  $U^*(t_k|t_k)$  as  $U(t_k)$ , then move one step and do search;
- 5:     Update  $\Omega^T(t_k)$  by Equations (11-13);
- 6:     Update  $\Omega^H(t_k)$  by Equations (14) and (15);
- 7:     Update  $\Omega^R(t_k)$  by Equations (16-18);
- 8:     for  $i=1:N_U$
- 9:         if any  $P(Trc) \geq P_{thold}^D$  for  $g_{rc}^{D^i} \in \Omega^{D^i}(t_k)$      //The existence probability in  $g_{rc}^{D^i}$  exceeds the threshold
- 10:             Turn  $i$ -th into tracking state and no longer participate in subsequent search missions;
- 11:             Clear the element of the discovered target in  $\Omega^T$  and  $\Omega^R$  to block the influence on the subsequent search;
- 12:         end
- 13:     end
- 14:      $k \leftarrow k + 1$ ;
- 15:     end

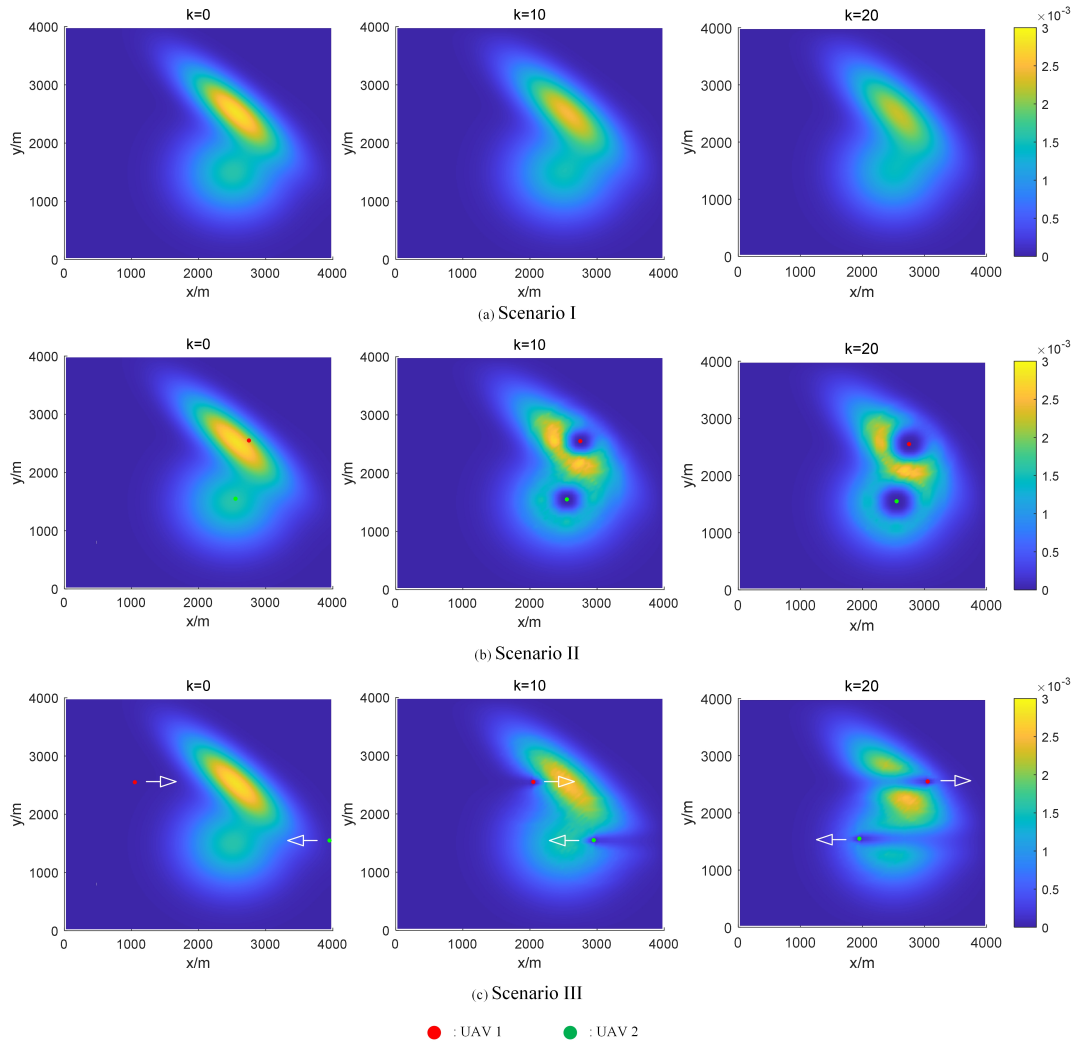
---

detection configuration for UAVs to be:  $\gamma_D = 5, p_{max}^D = 0.95, p_{min}^D = 0.2, \delta_{Dmin} = 100m, \delta_{Dmax} = 400m$ , and  $p_{thold}^D = 0.75$ .

### 6.1. Motion prediction of moving targets

Assuming that there are two targets to be searched in the mission area, the initial position of the target is subject to the two-dimensional normal distribution, where the means of position are  $[2500 \ 1500]^T$  and  $[2500 \ 2600]^T$ , and the variances of position are  $[160000 \ 0; \ 0 \ 160000]$  and  $[160000 \ -128000; \ -128000 \ 160000]$ .

To test the update process of the probability map, three experimental scenarios were used: (I) target prediction without the search interference of UAVs, (II) under the search interference of static UAVs, and (III) moving UAVs. Figure 8 shows the updating process of the probability map in different scenarios at simulation steps  $k = 0, k = 10$  and  $k = 20$ . In scenario I, as the targets are not affected by the UAV search, the target follows



**Figure 8.** Updating process of the probability map in different scenarios.

a Gauss-Markov motion model. The result shows that the existence probability of the target spreads towards adjacent grid cells, and the uncertainty of the target position increases with the simulation steps. In Scenario II, the target will be subject to the evasive virtual force A generated by the UAVs in addition to random movement, showing a trend of evading the UAVs. The result shows that there is a significant reduction in the existence probability of the target around the UAVs. In Scenario III, the target is affected by the evasive virtual forces A and B, and the result shows that the target in the UAVs' forward direction will additionally evade both sides of the forward direction, while the target behind the UAVs may return to the previous search area by random movement after the UAVs have moved away.

Compared with previous methods<sup>[10,11]</sup>, the changes in the probability map in the simulation results are consistent with the actual experimental scenario rather than purely numerical assumptions, reflecting the reasonable evasive maneuvers of moving targets and improving the efficiency of the search process.

## 6.2. Analysis on the influence of different factors

In addition to the configuration in the first experiment, set  $N_U = 3$  with the initial position of  $[850 \ 850]^T$ ,  $[1850 \ 3550]^T$  and  $[1250 \ 2250]^T$ . To avoid a lengthy coefficient tuning process, a set of coefficients that have performed well in multiple experiments is directly presented here, and the factor tuning of the objective

**Table 3. Average number of targets found with different search coefficients**

$\lambda_H \backslash \lambda_R$	$\lambda_R = 0.2$	$\lambda_R = 0.5$	$\lambda_R = 0.8$
$\lambda_H = 0.2$	1.15	1.48	1.62
$\lambda_H = 0.5$	1.28	1.55	1.68
$\lambda_H = 0.8$	1.25	1.53	1.58

function and IGAFa is mainly discussed in the following. For the coefficients in the search information map, set  $N_K = 20$ ,  $\eta_H = 0.95$ ,  $\eta_{R1} = 0.67$ ,  $\eta_{R2} = 0.9$  as a correspondence to the update method of search information map mentioned earlier. For the coefficients in the basic parameters in IGAFa, set  $P_{c1} = 0.4$ ,  $P_{c2} = 0.3$  and  $P_m = 0.3$  as the basic configuration of the following experiments.

In this experiment, the coefficients in the objective function and the factors in IGAFa will be determined. Firstly, considering the sensitivity to various indices and the balance between benefits and penalties, set  $\lambda_E = 1.0$  as the basis of the main search benefit and set  $\lambda_C = 0.1$  as the basis of the penalty for UAV motion cost. To test the influence of the configuration with different ratios of  $\lambda_H$  and  $\lambda_R$ , we use a traditional GA as the basic algorithm to search the moving target with the same probability distribution of the initial position as that in the previous experiment. Table 3 shows the search result of the number of targets found at simulation step  $k = 100$ .

Table 3 indicates that the group of  $\lambda_H = 0.5$  and  $\lambda_R = 0.8$  has the best optimization performance. Therefore, the coefficients are set accordingly for better detection-evasion effectiveness of adversarial games in the search process.

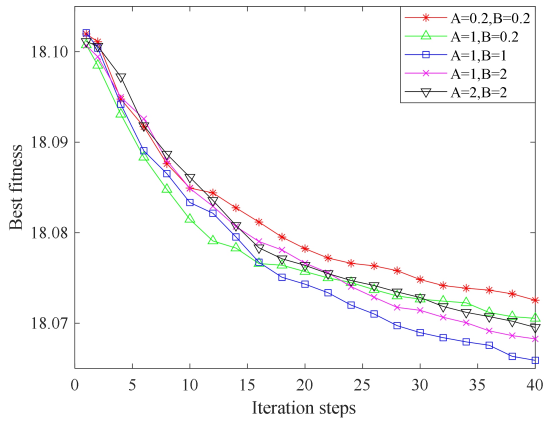
Secondly, the factors in IGAFa will be determined by calculating the optimal search path for UAVs at their initial position. First of all, without considering the weight matrix during the mutation process, test the influence of different  $A_{IT}$  and  $B_{IT}$  on the result of optimization. Figure 9 shows the best fitness under different iteration steps with different  $A_{IT}$  and  $B_{IT}$ . It indicates that the group of  $A_{IT} = 1$  and  $B_{IT} = 1$  has the best optimization performance. Therefore, the factors are set accordingly for a better evolution process of the probability distribution of selected genes.

Next, considering the weight matrix during the mutation process, we test the influence of different  $\eta_W$  on the result of optimization. Figure 10 shows the best fitness under different iteration steps with different  $\eta_W$ . It indicates that  $\eta_W = 0.5$  has the best optimization performance. Therefore, the factor is set accordingly for a better process of mutation.

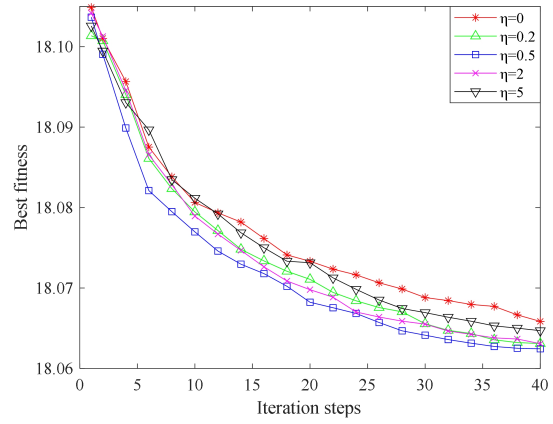
### 6.3. Process and result analysis of a complete search simulation

Based on the configuration in the second experiment, randomly arrange moving targets with pre-set position distribution and simulate the complete cooperative search process using CSMTPE with the procedures in Table 2. Figure 11 shows the search process of a complete search simulation, including the current search path, the search path for the next  $N_K$  steps, and the probability map, certainty map, and detection response map at the simulation steps  $k = 1$ ,  $k = 19$ , and  $k = 40$ , where UAV 1 finds and locks on Target 1 at step  $k = 19$  and UAV 2 finds and locks on Target 2 at step  $k = 40$ .

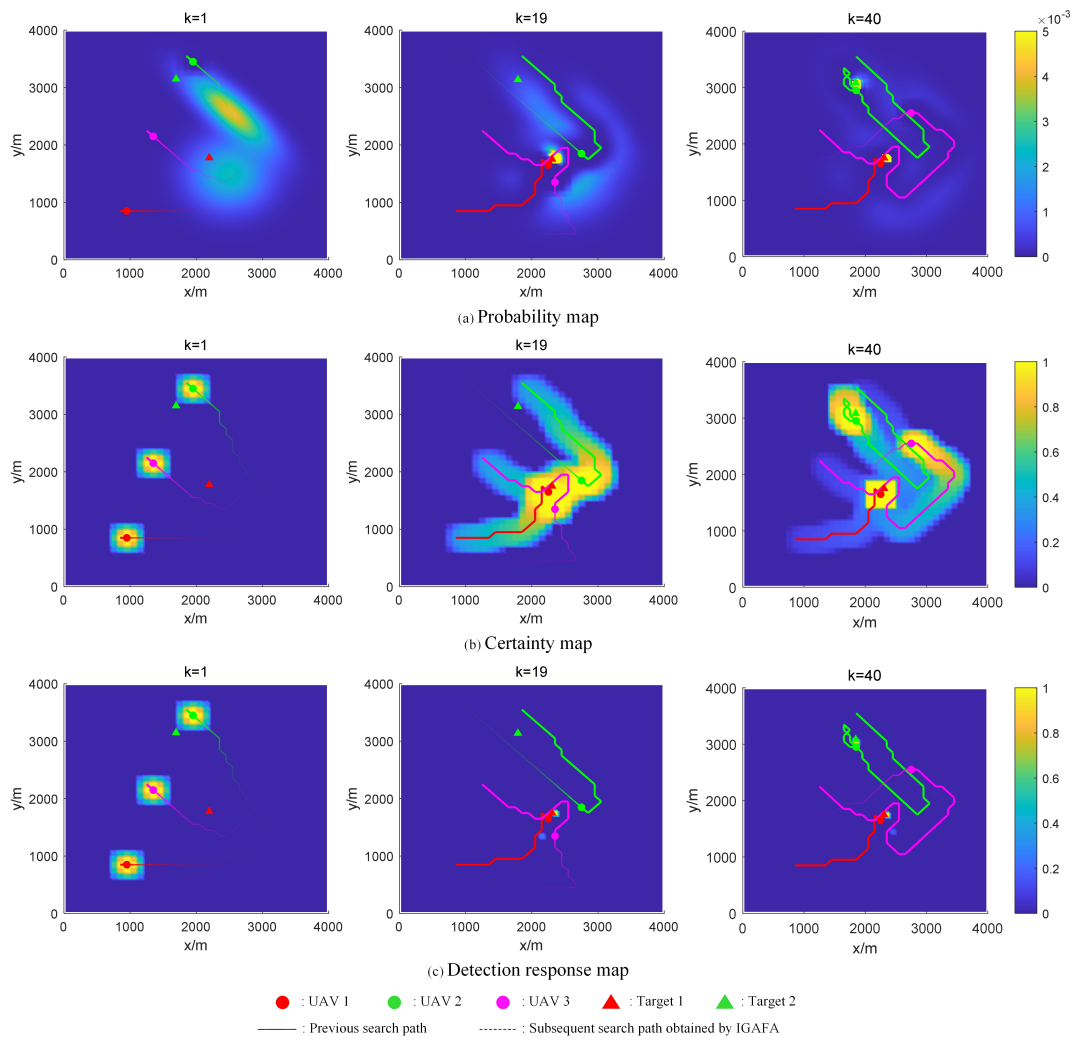
In the probability map in Figure 11(a), the existence probability of the target dynamically changes with the search process, and the UAV swarm converges and aggregates the probability to the actual position of the targets, achieving an effective search process. The certainty map in Figure 11(b) reflects the currently searched area of the UAV swarm, avoiding repeated searches of the same area in a short time, while the lower certainty after a long time will guide the UAVs to revisit the area to enhance the search order in the later stage of the



**Figure 9.** Best fitness under different iteration steps with different  $A_{IT}$  and  $B_{IT}$ .



**Figure 10.** Best fitness under different iteration steps with different  $\eta_w$



**Figure 11.** Search process under different simulation steps.

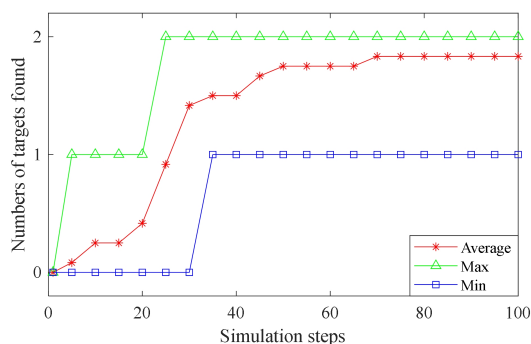


Figure 12. Numbers of targets found under different simulation steps.

Table 4. Average number of targets found with five cases of configuration

Number of targets	$N_U = 3$			$N_U = 6$		
	Step=25	Step=50	Step=100	Step=25	Step=50	Step=100
$N_T = 1$	0.68	0.93	1*	—	—	—
$N_T = 2$	0.90	1.75	1.83	1.40	1.83	2*
$N_T = 3$	—	—	—	1.93	2.50	2.75
$N_T = 4$	—	—	—	2.33	3.33	3.58

\* represents that all targets are found

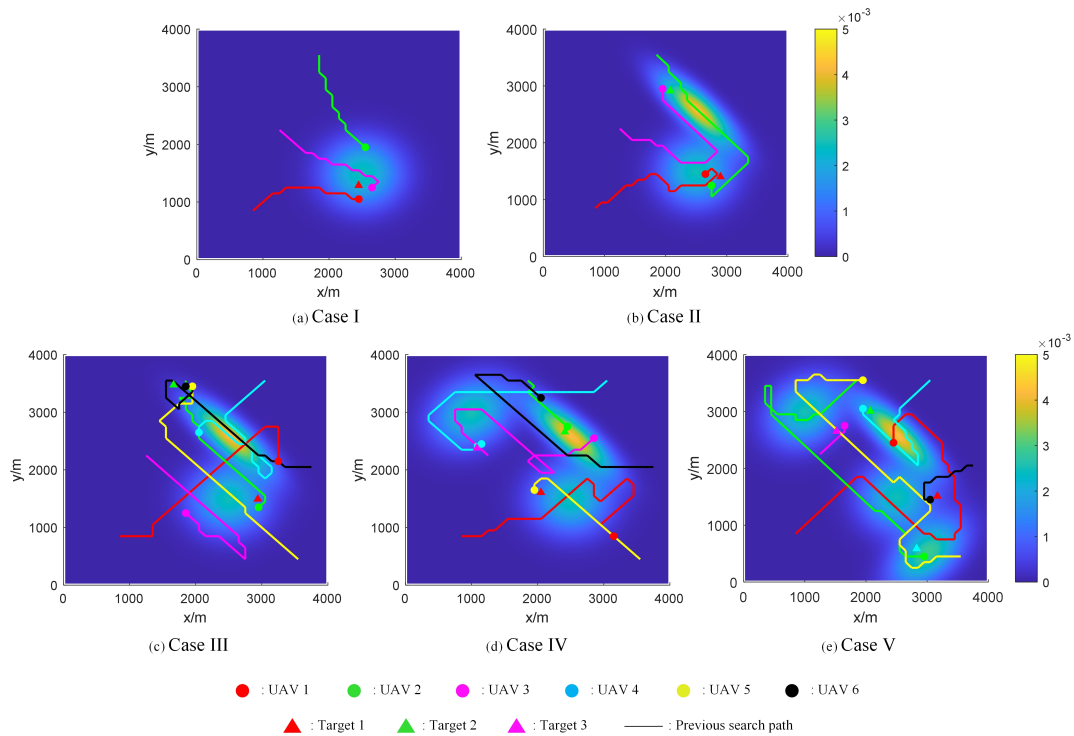
search. The detection response map in Figure 11(c) reflects the current target detection situation of the UAV swarm, which includes both the real alarm signal caused by the target and the false alarm signal of the sensor, and finally guides the UAVs to focus on searching the alarm area. Figure 11(a) also reflects that the search path guided by the CSMTPE can effectively explore the area with the initial existence probability of the target, and the priority of accessing the area with higher probabilities is greater. Meanwhile, the allocation of search areas among UAVs is also reasonable.

Furthermore, the above initial conditions are used for multiple simulation experiments, where the initial positions of the targets are randomly set differently in each simulation. The target search results within the maximum simulation step of 100 are recorded. Figure 12 shows the result of 40 Monte Carlo experiments of the search for two moving targets. The result indicates that the number of targets found gradually increases with the simulation steps, with an average of 1.83 targets found and a search completion rate of 92% within 100 simulation steps.

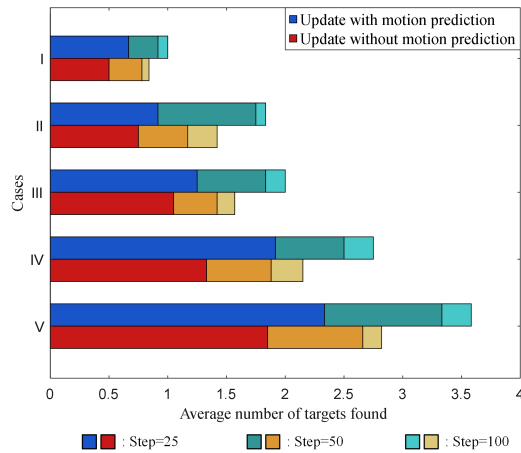
In order to simulate more search scenarios, five cases of configuration were designed here, with the number of UAVs and targets being 3:1, 3:2, 6:2, 6:3, and 6:4, respectively. Figure 13 shows the search process in different cases of UAV-target configuration. In each case, the background of each figure shows the probability distribution of the targets' initial position, while the positions of UAVs and moving targets are plotted at the moment when all targets were discovered. The result of the average number of targets found is shown in Table 4.

The result in Table 4 indicates that the larger the ratio of the target number to the UAV number, the higher the corresponding search completion rate, and the faster the target can be found, and it also proves that the CSMTPE proposed in this paper can effectively cope with the multiple target search in different target distribution situations.

In order to verify the effectiveness of the update method of probability map based on target motion prediction, the search result was tested under the same simulation conditions without using Equation (10) for the update of the probability map. Based on the search result in Table 4, Figure 14 shows an intuitive comparison of the search



**Figure 13.** Search process in different cases of UAV-target configuration.

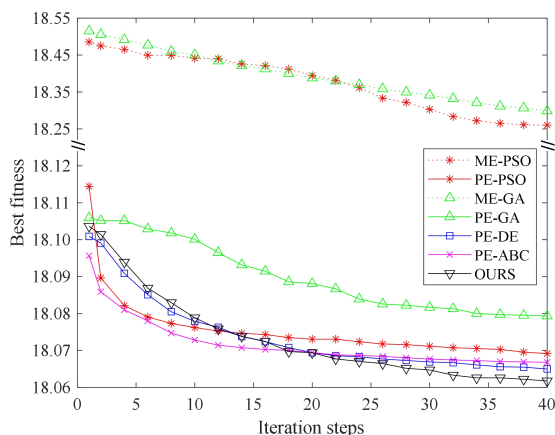


**Figure 14.** Comparison of the search results with and without the motion prediction.

result with and without the motion prediction, and it clearly indicates that when target motion prediction is not used, the average number of targets found will significantly decrease, which is because probability map cannot effectively predict the random and evasive movement of the target. Moreover, when the UAV approaches the target, the existence probability of the target shifts between adjacent grid cells without motion prediction due to the rapid movement of the target, and the probability of losing the target will be significantly increased.

### 6.4. Comparison of different optimization algorithms

In order to compare the optimization effectiveness of the proposed optimization algorithm, a set of optimization algorithms will be used as a comparison in the following experiment. First of all, a motion-encoded



**Figure 15.** Best fitness under different iteration steps using different optimization algorithms.

particle swarm optimization<sup>[10]</sup> (ME-PSO) and a motion-encoded genetic algorithm<sup>[15]</sup> (ME-GA) are chosen as the comparison group of the direct motion-encoding method as these are commonly used to solve similar problems in recent research. Secondly, to test the proposed priority-encoding method, these optimization algorithms have been rewritten with corresponding encoding methods and denoted as PE-PSO and PE-GA, respectively. In addition, to compare with more optimization algorithms, a differential evolution (DE) algorithm<sup>[28]</sup> and an artificial bee colony (ABC) algorithm<sup>[29]</sup> are chosen and rewritten with priority-encoding methods, which are PE-DE and PE-ABC.

In order to match the parameters used by IGAGA in the second experiment, the parameters in GA are set to  $P_c = 0.7$  and  $P_m = 0.3$ , and the PSO, DE and ABC have the same population size as IGAGA. **Figure 15** shows the best fitness under different iteration steps using different optimization algorithms. The result shows that the fitness optimized by ME-PSO or ME-GA is significantly higher than the fitness using the indirect priority-encoding method proposed in this paper, which indicates that the direct motion-encoding method is not appropriate when dealing with the proposed optimization problem. The fundamental reason is that the direct motion-encoding method cannot guarantee the legitimacy of its descendants, and it is difficult to achieve convergence of the optimal solution during the optimization. Each bit in the encoding has a significant influence on the results, and the optimization is inefficient under a large solution space.

Among the algorithms that use the priority-encoding method, traditional PE-GA has the slowest optimization speed. By comparison, PE-PSO, PE-DE and PE-ABC have faster optimization speeds in the early stage due to their characteristics being close to random-search. However, in the later stage, considering the continuity problem of the solution, the optimization speed significantly decreases. As an improved algorithm for GA, priority-encoded IGAGA optimizes the evolution direction while retaining the excellent genes of its parents, making the evolution process closer to the encoding characteristics, thus maintaining good optimization efficiency in the middle and later stages of the optimization process. It proves that the fine-adjustment mechanism proposed in this paper has efficient optimization performance when dealing with priority-encoding methods.

In order to further compare the influence on the target search process when using different optimization algorithms, PE-GA and ME-PSO are selected as typical optimization algorithms representing priority-encoding method and motion-encoding method, respectively.

**Table 5** shows the average execution time with different  $N_K$  and  $N_U$ . The result indicates that ME-PSO has a shorter execution time because of the defect of illegal descendant, and IGAGA has a longer execution time

**Table 5. Average execution time by typical optimization algorithms**

Number of UAVs	$N_K = 10$			$N_K = 20$		
	PE-GA	ME-PSO	OURS	PE-GA	ME-PSO	OURS
$N_U = 3$	2.8s	2.6s	3.1s	4.9s	4.8s	5.1s
$N_U = 6$	4.6s	4.5s	4.8s	7.8s	7.5s	8.5s

**Table 6. Average number of targets found using typical optimization algorithms at simulation step  $k = 100$** 

Number of targets	$N_U = 3$			$N_U = 6$		
	PE-GA	ME-PSO	OURS	PE-GA	ME-PSO	OURS
$N_T = 1$	1*	0.43	1*	—	—	—
$N_T = 2$	1.68	0.85	1.83	1.95	1.18	2*
$N_T = 3$	—	—	—	2.58	1.35	2.75
$N_T = 4$	—	—	—	3.23	1.85	3.58

\*\*\* represents that all targets are found

**Table 7. Average number of targets found using different search methods at simulation step  $k = 100$** 

Number of targets	$N_U = 3$			$N_U = 6$		
	Greedy search	Parallel search	CSMTPE	Greedy search	Parallel search	CSMTPE
$N_T = 1$	0.93	0.60	1*	—	—	—
$N_T = 2$	1.65	1.23	1.83	2*	1.38	2*
$N_T = 3$	—	—	—	2.40	2.00	2.75
$N_T = 4$	—	—	—	3.13	2.48	3.58

\*\*\* represents that all targets are found

because of the added improvement methods, while the execution time of the three algorithms is roughly the same.

We apply different algorithms to the search scenarios set in the third experiment and simulate the cooperative search using CSMTPE. The results of the average number of targets found using different optimization algorithms at simulation step  $k = 100$  are shown in [Table 6](#).

[Table 6](#) indicates that the ME-PSO has a significantly smaller number of targets found compared to other algorithms, while the search result using PE-GA is slightly worse than that using IGAGA. This is consistent with the conclusion obtained in [Figure 15](#), proving that IGAGA is more suitable for solving such problems.

### 6.5. Comparison of different search methods

We now compare the CSMTPE proposed in this paper with traditional greedy search method and parallel search method. The greedy search method calculates each grid cell within the motion range of each UAV in each simulation step according to [Equation \(10\)](#) with  $N_K = 1$ , and selects the cell with the minimum objective function value as the cell to be searched in the next simulation step. The parallel search method refers to the parallel search of targets by each UAV in the mission area, achieving a coverage search. [Table 7](#) shows the search results using different search methods at simulation step  $k = 100$  under the same configuration in the third experiment, and [Figure 16](#) shows the combination result of targets found at simulation steps  $k = 25$ ,  $k = 50$ , and  $k = 100$ .

[Table 7](#) indicates that CSMTPE has more targets in comparing search methods. Parallel search does not consider the initial distribution information of the target, and the search range is too rough, resulting in poor search efficiency. Greedy search has a similar search rate to CSMTPE in Case I and Case III, but overall, the number of targets found is lower than CSMTPE.

As shown in [Figure 16](#), when using the greedy search method, the number of targets found is significantly



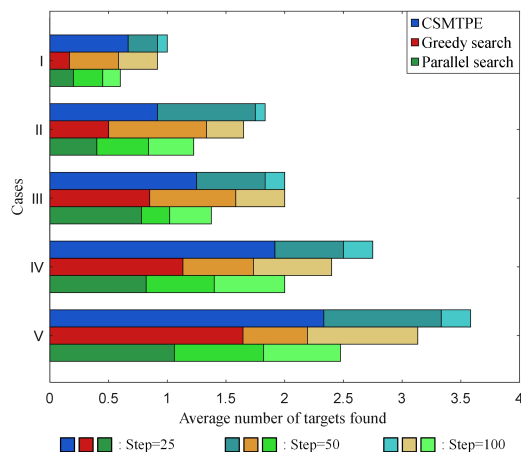


Figure 16. Comparison of the search results using different search methods.

Table 8. Average number of targets found using different search methods at simulation step  $k = 100$  without any prior information of the probability distribution of target position

Number of targets	$N_U = 3$			$N_U = 6$		
	Greedy search	Parallel search	CSMTPE	Greedy search	Parallel search	CSMTPE
$N_T = 1$	0.38	0.58	0.73	—	—	—
$N_T = 2$	0.95	1.23	1.35	1.20	1.43	1.53
$N_T = 3$	—	—	—	1.53	1.95	2.15
$N_T = 4$	—	—	—	2.25	2.53	2.68

less than that of CSMTPE at simulation step  $k = 25$ , reflecting that it wasted more search steps due to blind search in the early stage of the search, while the number of targets found in parallel search method is related to the target distribution, resulting the poor search efficiency. In contrast, the CSMTPE proposed in this paper continuously updates various search information maps to obtain more target information and takes into account both short-term and long-term search efficiency when planning search paths.

Considering the existence of situations where the prior information of the target is unknown, the original problem will degenerate into a covering search problem. Under this circumstance, another similar experiment is conducted, and the search results are shown in Table 8.

When the prior information of the target is unknown, the search results reflect the conclusion that the search efficiency of CSMTPE will degenerate into the level of that using the parallel search method. Meanwhile, it also shows that the greedy search method performs significantly worse compared to other search methods above due to the lack of long-term predictability. In summary, it proves that CSMTPE can adaptively and effectively solve the multi-UAV regional cooperative search problem.

### 7. CONCLUSIONS

This paper addresses a challenging multi-UAV regional cooperative search problem for targets with the ability to perceive and evade. In this scenario, moving targets can detect the presence of UAVs and take evasive actions based on the UAVs’ motion patterns, increasing the difficulty of target search. To solve this problem, we propose a novel search method called the CSMTPE for multi-UAV.

Firstly, we defined the motion model of such targets and design various search information maps and their

update methods. Secondly, we established a multi-UAV search path planning optimization model based on MPC and designed a CSMTPE with various objective functions of search benefits and costs. Thirdly, we proposed an IGAGA to solve this optimization model. Finally, we conducted simulation experiments to evaluate the proposed methods, and the results confirm their effectiveness.

Our experimental results show that the CSMTPE with IGAGA has higher search efficiency compared to traditional search methods, making it well-suited for the dynamic search process for targets with the ability to perceive and evade. Overall, the proposed method could have practical applications in various fields, such as search and rescue, surveillance, and military operations.

## DECLARATIONS

### Authors' contributions

Made substantial contributions to the conception and design of the study and performed simulation and interpretation: Wang Z

Provided administrative, technical, and material support: Zou W, Li S

### Availability of data and materials

Not applicable.

### Financial support and sponsorship

None.

### Conflicts of interest

All authors declared that there are no conflicts of interest.

### Ethical approval and consent to participate

Not applicable.

### Consent for publication

Not applicable.

### Copyright

© The Author(s) 2023.

## REFERENCES

1. Qu Y, Sun Y, Wang K, Zhang F. Multi-UAV cooperative search method for a moving target on the ground or sea. In: 2019 Chinese Control Conference (CCC). IEEE; 2019. pp. 4049–54. [DOI](#)
2. Wang S, Njau CE, Jiang Z. Design and implementation of multi-uav cooperation search experimental platform. In: 2021 5th International Conference on Robotics and Automation Sciences (ICRAS). IEEE; 2021. pp. 94–8. [DOI](#)
3. Chen S, Li C, Zhuo S. A distributed coverage algorithm for multi-UAV with average voronoi partition. In: 2017 17th International Conference on Control, Automation and Systems (ICCAS). IEEE; 2017. pp. 1083–86. [DOI](#)
4. Pehlivanoglu YV. A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV. *Aerosp Sci Technol* 2012;16:47–55. [DOI](#)
5. Apostolidis SD, Kapoutsis PC, Kapoutsis AC, Kosmatopoulos EB. Cooperative multi-UAV coverage mission planning platform for remote sensing applications. *Auton Robot* 2022;46:373–400. [DOI](#)
6. Fu X, Liu K, Gao X. Multi-UAVs communication-aware cooperative target tracking. *Appl Sci* 2018;8:870. [DOI](#)
7. Xiao X, Dong Z, Wu J, Duan H. A cooperative approach to multiple UAVs searching for moving targets based on a hybrid of virtual force and receding horizon. In: IEEE 10th International Conference on Industrial Informatics. IEEE; 2012. pp. 1228–33. [DOI](#)
8. Ru Cj, Qi Xm, Guan Xn, et al. Distributed cooperative search control method of multiple UAVs for moving target. *Int J Aerospace Eng* 2015;2015:1-12. [DOI](#)

9. Zhong Y, Yao P, Sun Y, Yang J. Method of multi-UAVs cooperative search for Markov moving targets. In: 2017 29th Chinese Control And Decision Conference (CCDC). IEEE; 2017. pp. 6783–89. [DOI](#)
10. Hu X, Liu Y, Wang G. Optimal search for moving targets with sensing capabilities using multiple UAVs. *J Syst Eng Electron* 2017;28:526–35. [DOI](#)
11. Yue W, Xi Y, Guan X. A new searching approach using improved multi-ant colony scheme for multi-UAVs in unknown environments. *Ieee Access* 2019;7:161094-102. [DOI](#)
12. Li Z, Xu P, Chang X, et al. When object detection meets knowledge distillation: A survey. *IEEE Trans Pattern Anal Mach Intell* 2023;45:10555-79. [DOI](#)
13. Li M, Huang PY, Chang X, et al. Video pivoting unsupervised multi-modal machine translation. *IEEE Trans Pattern Anal Mach Intell* 2023;45:3918-32. [DOI](#)
14. Shorakaei H, Vahdani M, Imani B, Gholami A. Optimal cooperative path planning of unmanned aerial vehicles by a parallel genetic algorithm. *Robotica* 2016;34:823–36. [DOI](#)
15. Alanezi MA, Bouchekara HR, Apalara TAA, et al. Dynamic target search using multi-UAVs based on motion-encoded genetic algorithm with multiple parents. *IEEE Access* 2022;10:77922–39. [DOI](#)
16. Luo R, Zheng H, Guo J. Solving the multi-functional heterogeneous UAV cooperative mission planning problem using multi-swarm fruit fly optimization algorithm. *Sensors* 2020;20:5026. [DOI](#)
17. Ma C, Zhu X, Liu S, Gui J, Yao W. A multi-UAV cooperative searching method based on differential evolution. In: 2022 34th Chinese Control and Decision Conference (CCDC). IEEE; 2022. pp. 5643–48. [DOI](#)
18. Zhikuo C, Chen W, Yuxing Z. A cooperative approach to multi-UAVs search for mobile targets based on pigeon-inspired optimization. In: 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC). IEEE; 2018. pp. 1–8. [DOI](#)
19. Fang Z, Wang J, Ren Y, et al. Age of information in energy harvesting aided massive multiple access networks. *IEEE J Select Areas Commun* 2022;40:1441–56. [DOI](#)
20. Zhang Y, Wang J, Zhang L, et al. Reliable transmission for NOMA systems with randomly deployed receivers. *Ieee T Commun* 2022;71:1179–92. [DOI](#)
21. Zhou Z, Liu D, Bao W, et al. Multi-UAV cooperative target search method based on nash equilibrium distributed model predictive control. In: Proceedings of the 4th International Symposium on Application of Materials Science and Energy Materials; 2020. pp. 631–41. [DOI](#)
22. Yao P, Wei X. Multi-UAV information fusion and cooperative trajectory optimization in target search. *IEEE Syst J* 2021;16:4325–33. [DOI](#)
23. Wei W, Wang J, Fang Z, et al. 3U: Joint design of UAV-USV-UUV networks for cooperative target hunting. *IEEE T Veh Technol* 2022;72:4085–90. [DOI](#)
24. Jia Z, Wenjun X, Qing G. Research on Multi-UAV collaborative search in dynamic environment. In: MATEC Web of Conferences. vol. 173. EDP Sciences; 2018. p. 02002. [DOI](#)
25. Kamrani F, Ayani R. Using on-line simulation for adaptive path planning of UAVs. In: 11th IEEE International Symposium on Distributed Simulation and Real-Time Applications (DS-RT'07). IEEE; 2007. pp. 167–74. [DOI](#)
26. de Alcantara Andrade FA, Reinier Hovenburg A, Netto de Lima L, et al. Autonomous unmanned aerial vehicles in search and rescue missions using real-time cooperative model predictive control. *Sensors* 2019;19:4067. [DOI](#)
27. Grefenstette JJ. Genetic algorithms and machine learning. In: Proceedings of the sixth annual conference on Computational learning theory; 1993. pp. 3–4. [DOI](#)
28. Das S, Suganthan PN. Differential evolution: a survey of the state-of-the-art. *IEEE T Evolut Comput* 2010;15:4–31. [DOI](#)
29. Gao Wf, Liu Sy, Huang Ll. A novel artificial bee colony algorithm based on modified search equation and orthogonal learning. *IEEE T Cybernetics* 2013;43:1011–24. [DOI](#)