A Multirobot Cooperative Area Coverage Search Algorithm Based on Bioinspired Neural Network in Unknown Environments

Bo Chen, Wenli Zhang, Fangfang Zhang, Yanhong Liu, *Member, IEEE*,
and Hongnian Yu, *Senior Member, IEEE*

***Abstract*—This article proposes a novel approach based on a bioinspired neural network (BIN) for multirobot area coverage search in unknown environments. We obtain the dynamic environment information in the search process of the multirobot by combining the BIN model with the grid map. We, then, design a search objective function to guide robots to explore the task area by applying the distributed model prediction control policy. Moreover, we further improve the search efficiency through a collaborative decision-making mechanism among robots. Finally, we conduct simulation studies, which demonstrate the effectiveness and superiority of the proposed approach.**

***Index Terms*—Area coverage search, bioinspired neural network (BIN), multirobot, unknown environments.**

I. Introduction

An IMPORTANT aspect for using autonomous robots to replace humans to complete some specific tasks is area coverage searching [1], [2]. The problem of area coverage searching in unknown environments widely exists in unmanned aerial vehicle reconnaissance [3], [4], the search of trapped personnel after the disaster [5], [6], land mine location [7], robot mapping [8], and other fields. The term "unknown environments" refers to the distribution of search targets and obstacles in the task area that is unknown, while the boundary of the task area is known [9]. When an autonomous robot performs area coverage search tasks in the unknown environments, the robot is required to obtain environmental information in real time and plan the next movement path under the following constraints [10], [11]:

1) the limited detection range of sensors carried by a single robot relative to the task area's size;
2) without the prior information of environments, including the distribution of targets and obstacles;
3) avoiding obstacles in real time when they appear in the detection range of the robot's sensor.

Multirobot systems have many applications and are an interesting field for researchers. Compared to the single robot limited by the individual working ability, a multirobot system (MRS) has the advantages of high parallelism, robustness, and collaboration in executing an area coverage search task [12]. However, in addition to the above three constraints, cooperation among robots to improve the search performance is another problem that must be solved when a multirobot system performs a regional search task [13]. As the multirobot system has the above requirements in the search process under an unknown environment, the robot is required to make the real-time decision on the search path according to the update of the environment information [14], [15].

Currently available research methods on multirobot area search fall into four main categories [16].

1) Grid-based methods usually divide the task area into several grids and plan the search path according to the environmental information in the grid [17], [18]. The Dempster–Shafer theory of evidence was applied in [19] to extract the information of environment from the sonar data to build a grid map of the environments. A cooperative area search algorithm for swarm robots in unknown environment was proposed in [20]. The algorithm realizes real-time path planning by using coverage rate as the search reward. What is more, a general algorithm (AdaSearch) was proposed in [21] for adaptive source seeking in the face of heterogeneous background noise. A highly scalable sampling-based planning algorithm was proposed in [22] for multirobot active information acquisition tasks in complex environments. However, owing to the lack of decision-making information exchange between robots, the cooperation of the multirobot system needs to be improved in this kind of method.

a) *Spanning tree coverage (STC)-based methods:* A multirobot coverage algorithm based on STC (MSTC) was proposed in [23] to cover known terrain with a team of robots, and an online and robust version of MSTC was considered in [24]. Besides, an auction-based STC was

Manuscript received 21 December 2021; revised 29 May 2022 and 26 July 2022; accepted 8 August 2022. Date of publication 30 August 2022; date of current version 8 June 2023. This work was supported by the National Natural Science Foundation of China under Grant 61603345, Grant 61703372, and Grant 61773351. *(Corresponding author: Fangfang Zhang.)*

Bo Chen, Wenli Zhang, Fangfang Zhang, and Yanhong Liu are with the School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: cbcb233@gs.zzu.edu.cn; zwl17806167256@163.com; zhangfangfang@zzu.edu.cn; liuyh@zzu.edu.cn).

Hongnian Yu is with the School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China, and also with the School of Engineering and the Built Environment, Edinburgh Napier University, EH10 5DT Edinburgh, U.K. (e-mail: h.yu@napier.ac.uk).

Digital Object Identifier 10.1109/JSYST.2022.3198712

proposed in [25] to solve the coverage tasks in large crowded configuration spaces. To balance well the efficiency, redundancy, and robustness in the cooperative coverage process, an artificially weighted STC algorithm is proposed in [26] for the distributed path planning of multiple flying robots. What is more, a method based on the Voronoi partition and STC was presented in [27] for multirobot simultaneous exploration and coverage. However, the STC-based methods are usually used with known environmental information and difficult to be applied in unknown environments.

b) *Evolutionary-algorithm-based methods:* The evolutionary algorithms have been introduced as the control mechanism of multirobot area search in recent years, such as particle swarm optimization (PSO) algorithm [28], wasp swarm algorithm [29], bacteria chemotaxis (BC) algorithm [30], bat algorithm (BA) [31], and so on. A swarm robot area coverage method based on the wasp swarm algorithm was proposed in [32]. Swarm robots interact with the environment based on the fixed response threshold model of wasps. An improved multirobot cooperative search method based on the PSO algorithm was proposed in [33], and an adaptive robotic PSO (A-RPSO) algorithm was further proposed in [10]. In the PSO-based methods, the potential field function is taken as the fitness function to guide the robot to search the uncovered area. A decentralized control algorithm of swarm robots inspired by the BC algorithm was designed in [34] for target search and trapping. An adaptive robot BA was proposed in [35] for multirobot area searching. However, the evolutionary-algorithm-based methods generally assume the target signal to be globally perceptive, which requires high performance of robot sensors and specific application scenes.

c) *Bioinspired neural network (BIN)-based methods:* The BIN is inspired from the Hodgkin and Huxley's membrane model for a biological neural system. Since the neural connection weights are set in the model design and the selection range is very wide, there is no need to find the optimal connection weights among neurons. Therefore, the BIN has no learning procedures, which can work in real time. Compared with the above three methods, the BIN-based methods do not rely on environmental prior information and have lower computational complexity, so it is widely used in unknown environments [36], [37], [38]. In [39], the BIN was used to guide multiple cleaning robots sweep the task area, and an extension was provided to avoid deadlock situations between the robots [40]. Furthermore, the BIN was further used for coverage search in three-dimensional underwater environments [41], [42]. However, such algorithms are prone to fall into local optima in the later stage of search. Besides, above methods still have significant room for improvement in multirobot system's search performance.

Considering the problems in the current study, this article proposes a novel multirobot cooperative coverage search algorithm based on the BIN, which provides a real-time search strategy for multiple autonomous robots to perform area search tasks in the unknown environments. Based on the above discussion, this article has the following four main contributions.

1) Develop a new information model of dynamic environments by combining the BIN and the grid map, where the unsearched grid attracts the robot globally and the obstacle grid repels the robot locally (see Section III).
2) Apply a distributed model-predictive control (DMPC) method to replace the gradient decline principle in search path planning, which can improve search efficiency by considering the short- and long-term search returns in search task (see Section IV-A).
3) Propose a novel collaborative decision-making mechanism to improve the collaboration performance of MRS, which can avoid the duplicated search of the same region caused by the decision-making conflict among robots (see Section IV-B).
4) Demonstrate that the proposed algorithm can maintain effectiveness and stability under different search environments (area's size and the number of robots), which can be applied to area search tasks under unknown environments through the experimental study (see Section V).

The rest of this article is organized as follows. Section II introduces the basic concepts used in this article. The BIN model is combined with the grid map to represent dynamic environment information in Section III. The DMPC method and the collaborative decision-making mechanism is presented in Section IV. Section V presents the experimental results. Finally, Section VI concludes this article.

II. Basic Concepts

A. Task Description

Owing to the complexity in the unknown environments, except for the constraints mentioned in Section I, the robots perform the area coverage search task under the following two assumptions.

Assumption 1: The size of the task area is large enough, and the autonomous robot is regarded as a particle in the task area.

Assumption 2: The targets in the unknown region are randomly distributed. If the target appears in the autonomous robot's detection range, the target is considered to be found, and its effect disappears.

According to Assumption 2, the number of targets detected by the robot is proportional to the size of the covered area. Under the above constraints and assumptions, the area coverage task requires multiple robots to cooperatively complete the maximum coverage of a specified area in a limited time T. The task objective function of the robots is shown as

$$OJ = \max \sum_{k=1}^{N_r} \mathrm{S}_{\mathrm{cov}}^{k} \ \text{(limited time}T) \tag{1}$$

where $\mathrm{S}_{\mathrm{cov}}^{k}$ represents the size of the covered area of the kth robot and N_r is the total number of the robots.

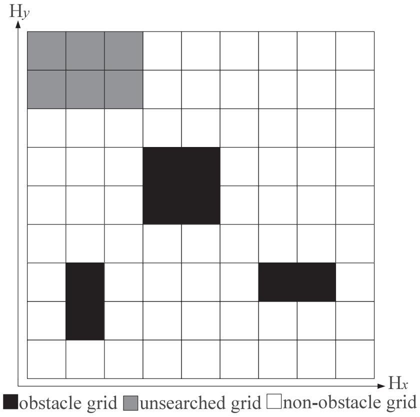


Fig. 1. Grid map.

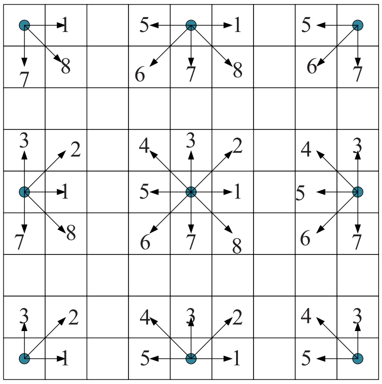


Fig. 2. Possible motion direction of robot.

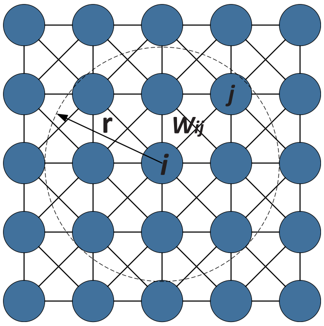


Fig. 3. Two-dimensional BIN structure diagram.

B. Task Search Area

Assume that the shape of the task search area is a rectangle. As for the representation methods of unknown environment map, currently, there are mainly three approaches: topological map [43], geometric map [44], and grid map [45]. Among the three approaches, grid map representation has the advantages of being simple and easy to implement and can better express the position distribution of objects in the actual space. As shown in Fig. 1, the task area is divided into $H_x \times H_y$ grids. Each grid has three states, as shown in (2). Since all the autonomous robots have no prior environmental information, the initial state of all the grids is K_u

$$\mathrm{S}\{G(h,v)\} = \begin{cases} K_u, & G(h,v)\text{is the unsearched grid} \\ K_s, & G(h,v)\text{is the obstacle grid} \\ K_o, & G(h,v)\text{is the nonobstacle grid} \end{cases} \quad (2)$$

where $G(h,v)$ represents the grid in position (h,v), with $h \in [1, H_x]$ and $v \in [1, H_y]$, and $\mathrm{S}\{G(h,v)\}$ represents the state of the grid $G(h,v)$, including unsearched grid (K_u), obstacle grid (K_o), and nonobstacle grid (K_s).

C. Establish Motion Model of the Robot

Suppose that there are N_r autonomous robots performing a region search task, and the robot set is expressed as $\Theta = \{\mathrm{R}_1, \mathrm{R}_2, \ldots, \mathrm{R}_{N_r}\}$. The mth robot is denoted as $\mathrm{R_m}$, $m = 1, \ldots, N_r$. When R_m moves to the kth step, the position is represented as $\{x_m(k), y_m(k)\}$; then, the position of R_m in the $(k+1)$th step is determined as

$$\begin{bmatrix} x_m(k+1) \\ y_m(k+1) \end{bmatrix} = \begin{bmatrix} x_m(k) \\ y_m(k) \end{bmatrix} + v_m(k)\Delta t \begin{bmatrix} \cos \omega_m(k) \\ \sin \omega_m(k) \end{bmatrix} \quad (3)$$

where $v_m(k)$ is the velocity of R_m, $v_m(k) \in [0, v_{\max}]$; Δt is the interval time between steps; and $\omega_m(k)$ is the turning angle of R_m, $\omega_m(k) \in [-\omega_{\max}, \omega_{\max}]$.

Fig. 2 shows the Moore neighborhood of the robot at different positions in the grid map. The circle represents the position of the robot, and the arrow represents the potential movement direction of the robot at the current position. The robot will move to one of the adjacent grids. Each robot occupies a grid in the grid map, and the detected range of the robot is $M \times N$ grids ($M, N = 3$ in this article). If a grid is in the robot's detected range, the grid is considered to be searched.

Remark 1: Because the distances between the robot and the center of the adjacent grid are different in Fig. 2, in order to simplify the calculation, this research adjusts the robot's speed so that each robot can move to the center of the next grid at the same time. According to Fig. 2, the speed of the robot in the direction of motion $\{2, 4, 6, 8\}$ should be $\sqrt{2}$ times that in the direction of motion $\{1, 3, 5, 7\}$.

III. BIN Model

The first part of the proposed method is to develop an information model of the dynamic environments to provide guidance for the robot to plan the search path. In this section, a BIN model is established, and the information of grid cell is represented by the neuron's activity values.

A. Basic Structure

To represent the dynamic environment information in search process, a two-dimensional BIN model is established on the basis of the grid map, as shown in Fig. 3. The circles in Fig. 3 represent neurons and correspond to the grids in the grid map. There are only local lateral connections among neurons. The network consists of $H_x \times H_y$ neurons. Each neuron has a matching neuron activity value Q, and the dynamic characteristic of the activity value of the ith neuron in the network is shown as follows [38]:

$$\frac{dQ_i}{dt} = -AQ_i + (B - Q_i)\left([\rho_i]^+ + \sum_{j=1}^{M} W_{ij}[Q_j]^+\right) - (D + Q_i)[\rho_i]^- \quad (4)$$

where ρ_i represents the external input signal received by the ith neuron; $([\rho_i]^+ + \sum_{j=1}^{M} W_{ij}[Q_j]^+)$ represents the excitation signal; $[\rho_i]^-$ represents the inhibition signal; $[\rho_i]^+ = \max(\rho_i, 0)$; $[\rho_i]^- = \max(-\rho_i, 0)$; Q_j represents the activity value of the neuron j adjacent to the neuron i; $[Q_j]^+ = \max(Q_j, 0)$; M is the number of adjacent neurons of the ith neuron (within its radius r); A, B, and D are all positive constants; A represents the attenuation rate of Q_i; and B and $-D$ are the upper and lower limits of Q_i, respectively.

In (4), W_{ij} represents the connection weight coefficient between the neuron i and the adjacent neuron j, defined as follows [36]:

$$W_{ij} = f_1(|e_i - e_j|) = \begin{cases} \frac{\alpha}{|e_i - e_j|} & 0 < |e_i - e_j| \leq r \\ 0 & |e_i - e_j| > r \end{cases} \quad (5)$$

where $|e_i - e_j|$ represents the Euclidean distance between the vectors e_i and e_j in the state space, r is the corresponding threshold, and α is a positive constant that makes $W_{ij} \in [0, 1]$. Since the BIN consists of $H_x \times H_y$ neurons and each neuron has at most eight local connections, the total number of neural connections is $8H$ $(H = H_x \times H_y)$. Consequently, the computational complexity of the BIN is $O(H)$.

B. Combined With the Grid Map

The purpose of the proposed model is to represent dynamic environmental information through neuron's activity landscape. According to the requirement in Section II-A, the robots should search unexplored areas and avoid obstacles. It can be seen from (4) that the change of neuron's activity value depends largely on the external input ρ_i (excitation signal or inhibition signal). Besides, the term $\sum_{j=1}^{M} W_{ij}[Q_j]^+$ ensures that the excitation signal can be transmitted globally, while the inhibition signal only acts locally. By defining appropriate external inputs, the unsearched areas will be at the peak of the neuron's activity landscape, and the obstacle will be at the lowest point. Owing to the propagation effect of positive neuron's activity, the unsearched areas will attract the robots globally. In addition, the obstacles only have local influence in a small area to avoid the collision. In the formulation of a search strategy, the robots can generally move toward the area with a large neuron's activity value, which can avoid obstacles and avoid repeatedly explore the same area.

Based on the above discussion, the combination mode of the BIN and the grid map state is shown in (6): if the autonomous robot detects the obstacle, $\rho_i = -C$, then the neuron's activity value of the corresponding grid will be at the lowest point; if a grid has no obstacle after detected by the robot, $\rho_i = 0$, then the corresponding neuron's activity value will be reduced to a certain extent; and if a grid is not searched, $\rho_i = C$, then the corresponding neuron's activity value is the maximum.

$$\rho_i(h, v) = \begin{cases} C, & \mathrm{S}\{G(h, v)\} = K_u \\ 0, & \mathrm{S}\{G(h, v)\} = K_s \\ -C, & \mathrm{S}\{G(h, v)\} = K_o \end{cases} \quad (6)$$

where $\rho_i(h, v)$ represents the external input of the ith neuron in position (h, v), $\mathrm{S}\{G(h, v)\}$ represents the state of the grid $G(h, v)$, and C is a sufficiently large positive value.

Table I illustrates some symbols in the pseudocode diagrams. Algorithm 1 shows the initialization procedure of the proposed method, including the grid map, the BIN, and the initial positions of all the autonomous robots. Algorithm 2 describes the updating process of the BIN in detail. First, each robot updates the grid state based on the information detected by the sensor in the current position; then, the robots will update the neuron's activity value based on (4) and (6).

TABLE I
PART OF NOTATION IN PSEUDOCODE DIAGRAMS

Symbol	Meaning
H_x, H_y	The size of grid map
$Q(h, v)$	Neuron's activity in position (h, v)
$\rho(h, v)$	External input to neuron in position (h, v)
$\{x_m(0), y_m(0)\}$	The initial position of the m-th robot
$\{P_x^m, P_y^m)\}$	The m-th position data entered by the user
$Q_c(h, v)$	Current neuron's activity in position (h, v)
$Q_r(h, v)$	Updated neuron's activity in position (h, v)
$\rho^v(\zeta, \xi)$	External input to virtual neuron in position (ζ, ξ)
$Q_c^v(h, v)$	Current virtual neuron's activity in position (h, v)
$Q_r^v(h, v)$	Updated virtual neuron's activity in position (h, v)

Algorithm 1: Initialization Procedure.

```
Set $\mathrm{S}\{G(h, v)\} = K_u, \forall 1 \leq h \leq H_x, 1 \leq v \leq H_y$.
// Initializes the state of grid map (set the state of all areas as unsearched)
Set $Q(h, v) = 0$; $\rho(h, v) = 0$,
    $\forall 1 \leq h \leq H_x, 1 \leq v \leq H_y$.
// Set the initial state of the BIN
for $m = 1$ to $N_r$ do
    $x_m(0) = P_x^m; y_m(0) = P_y^m$
end for
// Set the starting position of all robots
```

C. Model Stability Analysis

The neuron's activity value in (4) is bounded in the finite interval $[-D, B]$. In (4), the neuron's activity value increases at a rate of $(B - Q_i)([\rho_i]^+ + \sum_{j=1}^{M} W_{ij}[Q_j]^+)$, which is proportional to the neuron's external excitation signal $([\rho_i]^+ + \sum_{j=1}^{M} W_{ij}[Q_j]^+)$ and the automatic gain control term $(B - Q_i)$. When the neuron's activity value Q_i is less than B and the neuron receives external excitation signal, the closer Q_i gets to B, the slower Q_i grows. Once Q_i is equal to B, the value of $(B - Q_i)([\rho_i]^+ + \sum_{j=1}^{M} W_{ij}[Q_j]^+)$ become zero and Q_i will no longer keep increase. In case Q_i is bigger than B, the term $(B - Q_i)$ will become negative, which will pull Q_i back to

Algorithm 2: Update of the BIN.

```
Input:current positions of robots $x_m(k), y_m(k)$,
  $m \in [1, N_r]$; current neuron's activity value $Q_c(h, v)$,
  $h \in [1, H_x]$, $v \in [1, H_y]$; the state of grid map
  $\mathrm{S}\{G(h, v)\}$, $h \in [1, H_x]$, $v \in [1, H_y]$.
Output:renewed neuron's activity value $Q_r(h, v)$,
  $h \in [1, H_x]$, $v \in [1, H_y]$.
// Update the state of the grids
for $m = 1$ to $N_r$ do
    $\mathrm{R}_m$ update $\mathrm{S}\{G(\zeta, \xi)\}$,
    $\zeta \in [x_m(k) - 1, x_m(k) + 1]$,
    $\xi \in [y_m(k) - 1, y_m(k) + 1]$ based on (2)
// Each robot updates the grid state based on the
  information detected by the sensor
end for
// Update the neuron's activity value
for $h = 1$ to $H_x$ do
    for $v = 1$ to $H_y$ do
        update the $\rho_i(h, v)$ based on (6)
        Calculate the updated neuron's activity value
        $Q_r(h, v)$ based on (4) and $Q_c(h, v)$
    end for
end for
```

B. Therefore, B is the upper bound of neuron's activity value. Similarly, $-D$ is the lower bound of neuron's activity value. Therefore, once the neuron's activity value $Q_i \in [-D, B]$, it stays within this range for any external stimulus.

Besides, the stability and convergence of the proposed BIN model can be proved by the Lyapunov stability theory [46]. Constructing new variables, $\phi_i = Q_i - B$, three corresponding equations can be obtained

$$Q_i = B + \phi_i, \frac{dQ_i}{dt} = \frac{d\phi_i}{dt}, Q_j = B + \phi_j. \tag{7}$$

Substituting (7) into (4) leads to a new shunting equation

$$\begin{aligned}\frac{d\phi_i}{dt} = & -A(B + \phi_i) + (-\phi_i)\left([\rho_i]^+ + \sum_{j=1}^{M} W_{ij}[B + \phi_j]^+\right) \\ & - (D + B + \phi_i)[\rho_i]^-.\end{aligned} \tag{8}$$

By rearranging (8), we obtain

$$\begin{aligned}\frac{d\phi_i}{dt} = -\phi_i \Bigg\{ & \frac{1}{\phi_i}(AB + \phi_i(A + [\rho_i]^+ + [\rho_i]^-) + (B + D)[\rho_i]^-) \\ & - \sum_{j=1}^{M}(-W_{ij})[B + \phi_j]^+ \Bigg\}.\end{aligned} \tag{9}$$

Then, the proposed model in (4) can be written into Grossberg's general form [47]

$$\frac{d\phi_i}{dt} = a_i(\phi_i)\left(b_i(\phi_i) - \sum_{j=1}^{M} c_{ij} d_j(\phi_j)\right) \tag{10}$$

where

$$a_i(\phi_i) = -\phi_i \tag{11}$$

$$b_i(\phi_i) = \frac{1}{\phi_i}(AB + \phi_i(A + [\rho_i]^+ + [\rho_i]^-) + (B + D)[\rho_i]^-) \tag{12}$$

$$c_{ij} = -W_{ij} \tag{13}$$

$$d_j(\phi_j) = [B + \phi_j]^+. \tag{14}$$

According to (5), the weight of neural connection is symmetric, $W_{ij} = W_{ji}$; correspondingly, $c_{ij} = c_{ji}$. Since $Q_i \in [-D, B]$, then $\phi_i \in [-B - D, 0]$, where B and D are nonnegative constants and ϕ_i is a nonpositive number. Hence, the zoom function $a_i(\phi_i)$ is nonnegative, i.e., $a_i(\phi_i) \geq 0$. From the definition of function $[B + \phi_j]^+$, when $\phi_j < -B$, $d'_j(\phi_j) = 0$, and when $\phi_j > -B$, $d'_j(\phi_j) = 1$. Therefore, the signal function $d_j(\phi_j)$ has a nonnegative derivation ($d'_j(\phi_j) \geq 0$). According to the above derivation, (4) satisfies all the three stability conditions required by Grossberg's general form [47]. The following Lyapunov function for (4) is selected:

$$\Psi = -\sum_{i=1}^{M} \int^{\phi_i} b_i(\tau_i) d'_i(\tau_i) d\tau_i + \frac{1}{2} \sum_{j,h=1}^{M} c_{jh} d_j(\phi_j) d_h(\phi_h). \tag{15}$$

The time derivative of Ψ along all the trajectories is given by

$$\frac{d\Psi}{dt} = -\sum_{i=1}^{M} a_i d'_i \left(b_i - \sum_{j=1}^{M} c_{ij} d_j\right)^2. \tag{16}$$

Since both a_i and d'_i are nonnegative, then $d\Psi/dt \leq 0$ along all the trajectories. Hence, the proposed BIN system is stable. The dynamics of the network ensures that the system converges to an equilibrium state.

IV. Decision-Making Process

After detecting the environment information of the current position through sensors and updating the BIN, the robot needs to make decisions on the next movement path. The gradient descent method is widely used to determine the next search path after the BIN model is established [37], [38]. However, this decision method is easy to fall into local optima in the later stage of search, especially when the surrounding area of the robot is searched. Therefore, the DMPC method is applied to improve search efficiency by considering the short- and long-term search returns in a search task. In addition, in order to improve the search performance, a collaborative decision-making mechanism based on the greedy iteration idea is proposed for the multirobot system.

A. DMPC Method

The core idea of DMPC is the rolling optimization solution, which is used to design the optimal control input of the system within the prediction period by applying the control system model and optimization theory [48]. According to the equation of motion of R_m given in (3), we select $\{\omega_m(k), v_m(k)\}$ as the control input $u_m(k)$ for R_m. If the state of R_m at the kth step is $g_m(k) = \{x_m(k), y_m(k)\}$, then the state equation of R_m in the next step is denoted as

$$g_m(k+1) = f\left\{g_m(k), u_m(k)\right\} \tag{17}$$

where $f\{\cdot\}$ is the state transfer function of R_m $(m = 1, \ldots, N_r)$, determined by (3).

According to (17), the prediction model of R_m is established as

$$g_m(k+l|k) = f\left\{g_m(k+l-1|k), u_m(k+l-1|k)\right\} \tag{18}$$

where $m = 1, 2 \ldots, N_r$, $l = 1, 2 \ldots, L$, N_r is the total number of the subsystem, L is the number of predicted steps, and $g_m(k + l|k)$ is the predicted state of the next step based on the state and the control input of the $(k + l - 1)$th step of R_m.

When multiple autonomous robots perform an area search task, each robot is required to search the unexplored area and also to reduce the energy consumption caused by frequent turns in the search process. Therefore, we comprehensively consider the incremental gain of the neuron's activity value and the cost of turning energy consumption in the robots cooperative search and establishes the search objective function of R_m $(m = 1, \ldots, N_r)$ as

$$J\left\{g_m(k), u_m(k)\right\} = \beta_1 J_1(k) + \beta_2 J_2(k) \tag{19}$$

where J_1 represents the incremental gain of the neuron's activity value, J_2 represents the turning cost, β_1 and β_2 represent the weight coefficients, and their value range is $[0, 1]$, i.e., $\beta_1, \beta_2 \in [0, 1]$.

1) Incremental Gains in Neuron's Activity Values: As mentioned in Section III-B, the activity value of the unsearched area is at the peak of the neuron activity landscape and has a global attraction effect for the robot. In addition, the activity value of obstacle is at the lowest point in the landscape of neuron activity value and only has local influence. Therefore, the robot can move toward the area with high activity value, which is conducive to finding the unsearched area and realizing obstacle avoidance. Besides, R_m should be able to avoid collisions among robots. In conclusion, we design an incremental gains function of the neuron's activity value for each robot, as shown in (20). In (20), the first term is to guide robots to advance to the unsearched area, the second term is to avoid obstacles, and the third term is to avoid collisions among robots. Note that for the first term of $J_1(k)$, only the neuron's activity values of the unsearched grids and nonobstacle grids are accumulated, i.e., $\mathrm{S}\{G_m^k(\zeta, \xi) \neq K_o\}$. The purpose is to prevent excessive obstacles in the robot's detection range from adversely affecting the search efficiency

$$J_1(k) = \begin{cases} \frac{\sum_{\zeta=h-1}^{\zeta=h+1} \sum_{\xi=v-1}^{\xi=v+1} Q_m^k(\zeta,\xi)}{\eta * Q_{\max}}, & \mathrm{S}\{G_m^k(h,v) \neq K_o \\ J_{\min}, & \mathrm{S}\{G_m^k(h,v)\} = K_o \\ J_{\min}, & G_m^k(h,v) \; \in \; \emptyset(\mathrm{R}_m) \end{cases} \tag{20}$$

where $\mathrm{S}\{G_m^k(h, v)\}$ represents the state of the grid $(G_m^k(h, v))$ that R_m is located, $O(\mathrm{R}_m)$ represents the set of paths planned by other robots, $Q_m^k(\zeta, \xi)$ represents the neuron's activity value of the grid $G_m^k(\zeta, \xi)$ in the coverage range of the current position of R_m, η represents the number of the grids that the current position of R_m can cover, and $J_{\min}$ is a constant and takes a minimum value. $J_{\min} = -1$ and $Q_{\max} = B$ in this work.

2) Turning Energy Costs: When the robot is moving, the turn energy consumption of the robot in the search process should be considered. To avoid frequent turns in the search process, the following turning cost function is established:

$$J_2(k) = \frac{- \mid \omega_m(k) - \omega_m(k-1) \mid}{\omega_{\max}} \tag{21}$$

where $\omega_{\max}$ denotes the maximum turn angle of the robot; in this article, the robot can move in an omnidirectional direction, and it can rotate at most 180° to a certain angle (clockwise or counterclockwise), so $\omega_{\max} = 180^\circ$. In addition to reducing energy consumption, experiments show that another function of $J_2(k)$ is to maintain a certain degree of global search ability of the robot.

According to (17), once the control input $u_m(k)$ of R_m is given, the state of R_m is transferred from $g_m(k)$ to $g_m(k + 1)$. The state of the grid map and the BIN are also updated accordingly. Hence, the optimal control input $u_m(k)$ can be obtained by maximizing $J(g_m(k), u_m(k))$ through the optimization solution. According to (18), this method is also extended to the prediction of L steps, and the cumulative search objective function of L steps is obtained by summing the search objective function of each step:

$$J^{(L)}\left\{g_m(k), u_m(k)\right\} = \sum_{j=0}^{L-1} J\left\{g_m(k+j), u_m(k+j)\right\}. \tag{22}$$

By maximizing the cumulative search objective function $J^{(L)}(g_m(k), u_m(k))$, the optimal solution of future L steps $U^*(L)$ can be determined, as shown in (23). The first step of the optimal solution $u_m(k)$ is taken as the control input of R_m; then, at the $(k + 1)$th step, a new decision sequence is obtained based on the new state $g_m(k + 1)$

$$\begin{aligned} & U^*(L) = \mathrm{argmax} J^{(L)}\left\{g_m(k), u_m(k)\right\} \\ & \text{s.t.} \begin{cases} g_m(k+l|k) = f\left\{g_m(k+l-1|k), u_m(k+l-1|k)\right\}, \\ l = 1, 2, 3, \ldots L \\ g_m(k|k) = g_m(k) \end{cases} \end{aligned} \tag{23}$$

where $U^*(L) = \{u_m(k), u_m(k + 1) \ldots u_m(k + L - 1)\}$.

B. Collaborative Decision-Making Mechanism Among Robots

Each autonomous robot should consider the decision information of other robots when performing a search task, which is necessary to improve the team cooperation performance. Otherwise, it will be prone to decision-making conflict, resulting in low search efficiency and even collision among robots. Hence, we propose a collaborative decision-making mechanism based on the greedy iteration idea, which can realize dynamic task assignment among multiple autonomous robots. The pseudocode of iterative collaborative decision making is shown in Algorithm 4. The specific steps are as follows

Step 1: Determine the iteration order of robots. The robot that is iterated first is denoted as R_1, and the robot that is iterated last is denoted as R_{N_r}. In this article, R_1 is randomly selected from autonomous robots. R_2 is generated from the robots closest to R_1, and the generation of $\mathrm{R}_3, \ldots \mathrm{R}_{N_r}$ is the same way.

Step 2: Define R_1 actions. The current BIN is replicated and named "virtual BIN (VBIN)." Algorithm 3 describes the updating process of VBIN in detail. The main function of Algorithm 3 is to store decision information of different robots in the iterative decision-making process. From (23), R_1 obtains the L-step predicted motion control input $\{u_1(k), \ldots, u_1(k+L-1)\}$ according to the current state of the grid map and the "VBIN." Then, R_1 updates the "VBIN" (contains decision information of R_1) based on $\{u_1(k), \ldots, u_1(k+L-1)\}$ (see Algorithm 3), which will be sent to R_2.

Step 3: Perform iterative decision making. After receiving the "VBIN" sent by R_1, R_2 similarly obtains the L-step predictive control input $\{u_2(k), \ldots, u_2(k+L-1)\}$ according to it and the current grid map state. As the same as step 2, the predictive control input of R_2 is reflected on the "VBIN" by Algorithm 3 (contains decision information of R_1 and R_2), which is sent to the next decision maker, and this process is repeated for R_m $(m = 3, \ldots, N_r)$.

Step 4: Move to the next step. After R_{N_r} finishes the iteration, all the robots have been solved the L-step predictive control input. Each autonomous robot moves to the next step according to the first component of the predicted control input.

Remark 2: Algorithm 3 describes the updating process of the VBIN in detail. The main function of Algorithm 3 is to store decision information of different robots in the iterative decision-making process. In this article, the activity value of the BIN is applied to represent the actual environmental information. The VBIN is only used during the decision process, in which initial activity value is the current activity value of the BIN (before the decision making). The VBIN contains the decision information of the robots, so that the competition among robots can be avoided and the cooperative performance can be improved.

Algorithm 3: Update of the VBIN.

```
Input: The positions of the $m$th iteration robot $x_m(k), y_m(k), m \in [1, N_r]$; the $L$-step predicted motion control input $\{u_m(k), \ldots, u_m(k+L-1)\}$ of $\mathrm{R}^m$; current neuron's activity value $Q_c^v(h,v)$ of VBIN, $h \in [1, H_x], v \in [1, H_y]$; the state of grid map, $\mathrm{S}\{G(h,v)\}$ $h \in [1, H_x], v \in [1, H_y]$.
Output: renewed neuron's activity value $Q_r^v(h,v)$ of VBIN, $h \in [1, H_x], v \in [1, H_y]$.
Obtain the predicted arrival position $\{x_m^p(k+1), y_m^p(k+1)\}, \ldots, \{x_m^p(k+L), y_m^p(k+L)\}$ of $\mathrm{R}^m$ based on (3) and (18).
for $i = 1$ to $L$ do
    if $\exists\, \mathrm{S}\{G(\zeta,\xi)\} == K_u$, $\zeta \in [x_m^p(k+i)-1, x_m^p(k+i)+1]$, $\xi \in [y_m^p(k+i)-1, y_m^p(k+i)+1]$ then
        $\rho^v(\zeta,\xi) = 0$
// The area where the robot is expected to arrive is considered to be searched and the corresponding external input of virtual neuron is 0.
    end if
end for
// Update the neuron's activity value of VBIN
for $h = 1$ to $H_x$ do
    for $v = 1$ to $H_y$ do
        Calculate the updated virtual neuron's activity value $Q_r^v(h,v)$ using (4) and $Q_c^v(h,v)$
    end for
end for
```

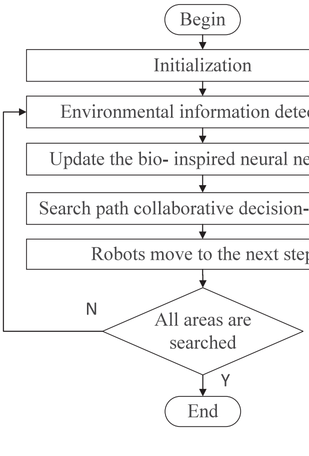

Begin
Initialization
Environmental information detection
Update the bio- inspired neural network
Search path collaborative decision-making
Robots move to the next step
N
All areas are searched
Y
End

Fig. 4. Flowchart of multirobot area coverage search.

C. Overall Description of the Proposed Algorithm

Fig. 4 describes the overall process of multirobot area coverage search. First, the initialization procedure includes setting up the state of the grid map and the BIN, as well as the initial positions of the robots (see Algorithm 1). Second, each robot detects the nearby environmental information and updates the BIN (see Algorithm 2). Then, all the robots collaborative to decide the next search path based on the BIN (see Algorithms 3 and 4). Finally, when all the areas are searched or the movement steps of the robots reach the set threshold, the entire area search process is over. Otherwise, the above process is repeated for each robot.

V. Verification and Comparison

In order to verify the effectiveness of the proposed algorithms (see Algorithms 1–4), the search process of the multirobot system is analyzed in Section V-A, and then, the influence of the number of robots and area's size on search performance are discussed in Sections V-B and V-C, respectively. MATLAB (2020a) is used as the simulation platform. The computer's main frequency is 2.9 GHz, and the memory is 16 GB. Initially, all the robots have no prior information of the environment. Each robot only know the boundary of the task area and does not know the distribution of obstacles and targets in the area. The robot's sensor detection range is the 3×3 grids. According to the parameter discussion of the BIN in [36], the performance of the BIN is not sensitive to the parameter setting. Therefore,

Algorithm 4: Iterative Collaborative Decision Making.

```
Input: The positions of the $m$th iteration robot $x_m(k), y_m(k)$, $m \in [1, N_r]$; current neuron's activity value $Q_c^v(h, v)$ of VBIN, $h \in [1, H_x]$, $v \in [1, H_y]$; the state of grid map $\mathrm{S}\{G(h, v)\}$, $h \in [1, H_x]$, $v \in [1, H_y]$.
Output: the next movement positions planned by the robots $x_m^p(k+1), y_m^p(k+1)$, $m = 1, 2, \ldots, N_r$.
The first robot for decision making ($\mathrm{R}_1$) is determined in a random way.
$m = 1$
// The $m$th robot make the decision
repeat
    obtain $\{u_m(k), \ldots, u_m(k+L-1)\}$ based on (23)
    obtain $x_m^p(k+1), y_m^p(k+1)$ based on (3) and $u_m(k)$
    update the VBIN using Algorithm 3
    $m = m + 1$
until $m > N_r$
// All robots finish the decision process and obtain the next movement positions.
```

we refer to the experience in [19] to set the parameters of the BIN as follows: $A = 0.2$, $B = 0.4$, $D = 0.5$, $C = 2$, $r = \sqrt{2}$, and $\alpha = 0.1$. In the DMPC method, set $\beta_1 = 0.9$, $\beta_2 = 0.1$, and $L = 3$.

The following three criteria are used to evaluate the performance of the algorithm under the same moving steps: 1) average regional coverage rate ($\mathrm{Cov_{ave}}$), as shown in (24); 2) standard deviation of coverage rate (Std), as shown in (25); and 3) average target search efficiency ($\mathrm{Suc_{ave}}$), as shown in (26). It is noted that $\mathrm{Suc_{ave}}$ is theoretically proportional to $\mathrm{Cov_{ave}}$ according to Assumption 2

$$\begin{cases} \mathrm{Cov_{ave}} = \frac{\sum_{i=1}^{N_v} \mathrm{cov}(i)}{N_v} \\ \mathrm{cov}(i) = \frac{S_{\text{searched}}^i}{S_{\text{Area}}} \; i = 1, 2, \ldots, N_v \end{cases} \tag{24}$$

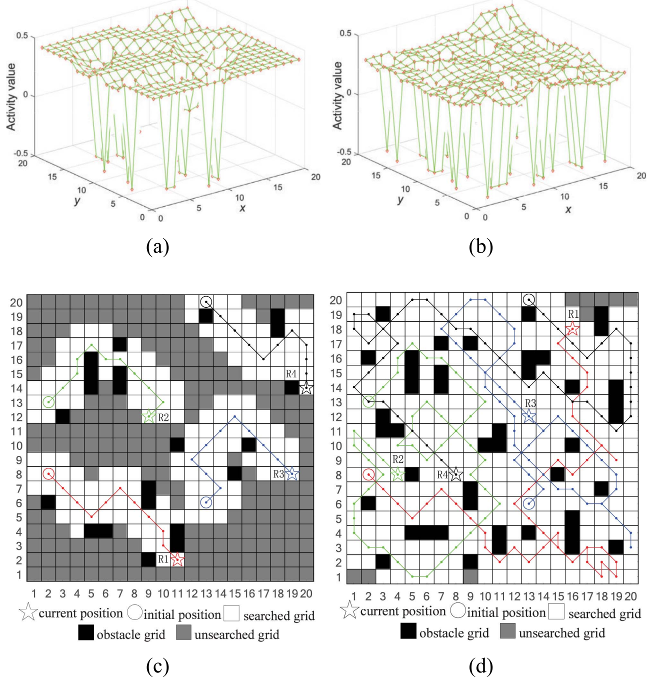
where $\mathrm{Cov_{ave}}$ represents the average regional coverage rate, $S_{\text{searched}}^i = \sum_{k=1}^{N_r} \mathrm{S}_{\text{cov}}^k(i)$ is the size of the searched area in the ith multirobot coverage experiment (number of robots is N_r), S_{Area} is the size of the task area, N_v is the total number of experiments, and $\mathrm{cov}(i)$ is the coverage rate of the ith multirobot coverage experiment

$$\mathrm{Std} = \sqrt{\frac{1}{N_v - 1} \sum_{i=1}^{N_v} (\mathrm{cov}(i) - \mathrm{Cov_{ave}})^2} \tag{25}$$

where Std is the standard deviation of the coverage rate and N_v is the number of experiments

$$\begin{cases} \mathrm{Suc}_{ave} = \frac{\sum_{i=1}^{N_v} \mathrm{suc}(i)}{N_v} \\ \mathrm{suc}(i) = \frac{O_{\text{searched}}^i}{O_{\text{All}}} \; i = 1, 2, \ldots, N_v \end{cases} \tag{26}$$

where $\mathrm{Suc_{ave}}$ is the average target search efficiency, O_{searched}^i is the number of searched targets in the ith multirobot coverage experiment, O_{All} is the number of targets in the task area, and $\mathrm{suc}(i)$ is the target search efficiency of the ith multirobot coverage experiment.



(a) (b) (c) (d)

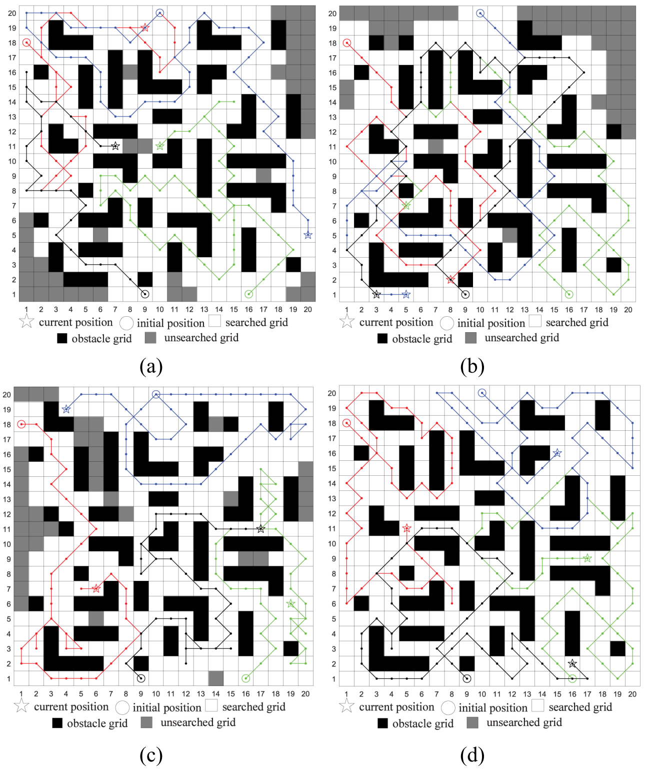
Fig. 5. Trajectories of different movement steps. (a) Neuronal activity value (steps = 10). (b) Neuronal activity value (steps = 45). (c) Movement trajectory (steps = 10). (d) Movement trajectory (steps = 45).

What is more, we chose three methods as baselines to verify the superiority of the proposed algorithm, namely, BIN algorithm [38], DMPC algorithm [13], and A-RPSO algorithm [10], respectively. In the BIN algorithm, the robot will select the neuron (grid) with the largest activity value of its surrounding neurons as its next movement position. In the DMPC algorithm, each robot determines the next search path through rolling optimization decisions, and there is no decision-making information exchange among robots. As for the A-RPSO algorithm, the potential field function is taken as the fitness function to guide the robot to search the uncovered area.

A. Robot Search Process

In this subsection, we analyze the robot search process by observing the movement of several autonomous robots. Relevant experimental parameters are as follows: the number of the robot is 4 and the task area is divided into 20×20 grids. Besides, 60 static targets are randomly distributed in the task area. Four robots start at R_1(2,8), R_2(2,13), R_3(13,6), and R_4(13,20), respectively.

The activity values of the BIN at the tenth and 45th steps are shown in Fig. 5(a) and (b), respectively. As can be seen from Fig. 5(a) and (b), with the movement of the robot, the activity values of neurons decreased to varying degrees, indicating that areas with high activity values (unexplored area) are attractive to the robot, while areas with low activity values are repulsive to the robot. The autonomous robots' movement trajectories at the tenth and 45th steps are shown in Fig. 5(c) and (d). It can be seen from Fig. 5(c) and (d) that all four robots can evade obstacles in


☆ current position ○ initial position □ searched grid
■ obstacle grid ▨ unsearched grid

(a)

(b)

(c)

(d)

Fig. 6. Robots' motion trajectories of different algorithms in the complex obstacle environment with narrow passages. (a) BIN algorithm [38]. (b) DMPC algorithm [13]. (c) A-RPSO algorithm [10]. (d) Proposed algorithm.

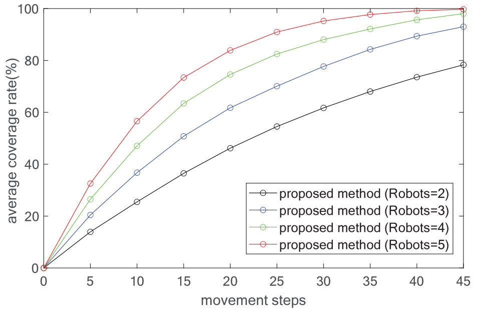


Fig. 7. Average coverage curve in the search process.

TABLE II
PERFORMANCE CRITERIA OF THE FOUR ALGORITHMS CORRESPONDING TO FIG. 6

Criterion / Algorithm	coverage rate (cov)	search efficiency (suc)
BIN algorithm [38]	89.50%	90.00%
DMPC algorithm [13]	88.50%	88.33%
A-RPSO algorithm [10]	90.75%	91.67%
proposed algorithm	100%	100%

real time and search the unexplored areas. The area around the obstacles can be fully searched, which indicates that the designed search objective function (19) is effective. In addition, there are no collisions between the four robots, and the trajectories are rarely repeated, which is the result of collaborative decision-making mechanism among autonomous robots (see Section IV-B). The average decision time of a single robot at each step is 0.0067 s, which has little influence in practical application and can be ignored. When movement steps reach 45, the task area has been basically searched (coverage rate: 97.75%).

To further illustrate the characteristics of the proposed algorithm, we compare the robot motion trajectories of the proposed algorithm and the other three algorithms in the complex obstacle environment with narrow passages, as shown in Fig. 6. The performance criterion of the four algorithms corresponding to Fig. 6 is shown in Table II. When the proposed algorithm has guided the robot to completely search the task area, there are still many unsearched areas for the other three algorithms. Under the same time, the coverage rate (cov) of the four algorithms is 89.50% (BIN algorithm), 88.50% (DMPC algorithm), 90.75% (A-RPSO algorithm), and 100% (proposed algorithm). The above results show that the proposed algorithm can effectively improve the cooperation of the multirobot system and the search ability of the complex obstacle environments.

B. Influence of the Number of Robots

This subsection discusses the influence of the robot's number on search performance. Experimental parameters are as follows: 1) the number of robots is set as 2, 3, 4, and 5, respectively; 2) the task area size is 20×20 grids; and 3) the number of movement steps of the robots is 45.

In order to eliminate the randomness of a single group of starting positions, each corresponding experiment (different robots' number) is repeated 50 times (50 groups different initial positions). Fig. 7 shows the average coverage curve of the proposed algorithm in the search process under different number of robots. It can be seen from Fig. 7 that with the increase in the number of robots, the coverage speed increases significantly. When the robots move 45 steps, the coverage rate reaches 99.76%.

Besides, three algorithms (BIN, DMPC, and A-RPSO) are selected as the compared group, and the corresponding search performance is shown in Table III. It can be seen from Table III that with the increase in the number of robots, the proposed algorithm maintains advantages in both "$\mathrm{Cov}_{\mathrm{ave}}$" and "$\mathrm{Suc}_{\mathrm{ave}}$," and "Std" is also the minimum. With the increase in the number of robots, the average coverage rate of the proposed algorithm is always higher than other three algorithms, indicating that the collaborative decision mechanism proposed in Section IV-B can guarantee the cooperation of the multirobot system. The above experiments show that the proposed algorithm can be applied to cooperative region coverage search of the multirobot system in unknown environment.

C. Influence of Task Area Size

This subsection discusses the influence of task area's size on robots' search performance. Search simulation experiments are carried out in 20×20, 20×40, 30×40, and 40×40 grid areas, respectively. The number of robots is four, and 50 targets are

TABLE III
SEARCH PERFORMANCE (DIFFERENT NUMBERS OF ROBOTS)

Robots	proposed algorithm			BIN algorithm [38]			DMPC algorithm [13]			A-RPSO algorithm [10]		
	Cov$_{ave}$	Std	Suc$_{ave}$	Cov$_{ave}$	Std	Suc$_{ave}$	Cov$_{ave}$	Std	Suc$_{ave}$	Cov$_{ave}$	Std	Suc$_{ave}$
2	78.32%	0.0392	78.10%	73.40%	0.0689	70.65%	71.87%	0.0739	80.95%%	73.65%	0.0596	74.52%
3	93.03%	0.0291	92.75%	86.58%	0.0516	83.28%	82.49%	0.0546	79.20%%	85.76%	0.0550	85.95%
4	98.04%	0.0168	97.45%	93.52%	0.0462	89.38%	89.24%	0.0485	87.01%%	92.46%	0.0374	93.51%
5	99.76%	0.0080	99.50%	96.51%	0.0316	92.47%	92.44%	0.0408	92.48%%	95.92%	0.0286	95.89%

TABLE IV
SEARCH PERFORMANCE (DIFFERENT SIZE OF THE TASK AREA)

Size (squares)	Steps	proposed algorithm			BIN algorithm [38]			DMPC algorithm [13]			A-RPSO algorithm [10]		
		Cov$_{ave}$	Std	Suc$_{ave}$	Cov$_{ave}$	Std	Suc$_{ave}$	Cov$_{ave}$	Std	Suc$_{ave}$	Cov$_{ave}$	Std	Suc$_{ave}$
20*20	45	98.54%	0.0139	97.48%	93.80%	0.0409	92.56%	88.94%	0.0478	87.24%	92.36%	0.0311	93.17%
40*20	90	97.79%	0.0180	97.44%	92.55%	0.0530	92.64%	93.14%	0.0373	93.88%	90.81%	0.0376	91.45%
40*30	135	98.32%	0.0108	98.44%	93.13%	0.0340	92.80%	95.36%	0.0274	95.08%	89.15%	0.0536	89.98%
40*40	180	98.51%	0.0101	98.84%	91.27%	0.0468	91.60%	95.49%	0.0223	95.84%	88.40%	0.0478	89.62%

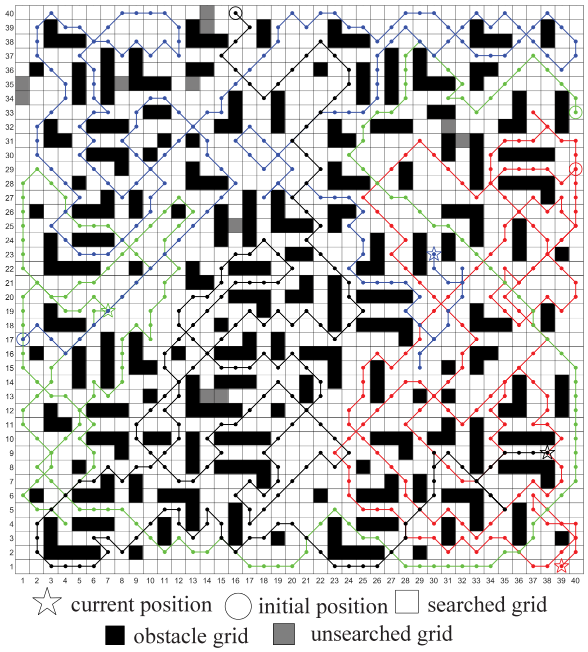


Fig. 8. Motion trajectories of four robots in the 40×40 grid map with complex obstacles.

randomly placed in the area. The movement trajectories of the four robots performing the area search task in the 40×40 grid map are shown in Fig. 8 (movement steps: 180). It can be seen from Fig. 8 that the four robots can basically complete regional coverage through cooperation (coverage rate: 99.31%). Each robot can avoid obstacles during the search process, and the motion trajectories between robots are rarely repeated.

Since different starting positions of the robots in the task area may obtain different search performance, in order to eliminate the randomness of a single group of starting positions, experiments are repeated 50 times in each area (50 groups of different initial positions of the robots are randomly selected). The algorithms in [10], [13], and [38] are selected as the compared group, and the corresponding search performance is shown in Table IV ("Step" represents the movement steps of the robots). Table IV shows that with the expansion of the area's size, the proposed algorithm achieves similar performance under the corresponding number of moving steps. This shows that the proposed algorithm can adapt to the variation of area's size. Compared with the other three algorithms, the proposed algorithm has an improvement in both "Cov$_{\text{ave}}$" and "Suc$_{\text{ave}}$," indicating the effectiveness of the proposed algorithm. In addition, the "Std" of the proposed algorithm is also the smallest, which indicates that the proposed algorithm can better adapt to the change of the initial positions of the robots.

VI. CONCLUSION

This article proposed a multirobot cooperative area coverage search approach based on the BIN in unknown environments. The proposed approach consists of two parts: the information model of dynamic search environments and the collaborative decision-making mechanism among robots. We conducted several simulation experiments of coverage search in an unknown obstacle environment and observed the simulation results from two aspects, including different area sizes and the number of robots. The results demonstrate the effectiveness of the proposed algorithm. Compared with the three previous algorithms in this field, the proposed algorithm improves both search efficiency and collaboration performance among robots. Besides, the proposed approach in this article is mainly applicable to the scenarios of small-scale robots. It is still an open issue for large-scale multirobot systems to perform area coverage search tasks in the unknown environments. This will be the topic of our future research.

REFERENCES

[1] J. Huo, M. Liu, K. A. Neusypin, H. Liu, M. Guo, and Y. Xiao, "Autonomous search of radioactive sources through mobile robots," *Sensors*, vol. 20, no. 12, 2020, Art. no. 3461.

[2] H. Choset, "Coverage for robotics—A survey of recent results," *Ann. Math. Artif. Intell.*, vol. 31, no. 1, pp. 113–126, 2001.

[3] M. Atif, R. Ahmad, W. Ahmad, L. Zhao, and J. J. P. C. Rodrigues, "UAV-assisted wireless localization for search and rescue," *IEEE Syst. J.*, vol. 15, no. 3, pp. 3261–3272, Sep. 2021.

[4] L. Li, X. Zhang, W. Yue, and Z. Liu, "Cooperative search for dynamic targets by multiple UAVs with communication data losses," *ISA Trans.*, vol. 114, pp. 230–241, 2021.

[5] N. Kumar, S. Misra, and M. S. Obaidat, "Collaborative learning automata-based routing for rescue operations in dense urban regions using vehicular sensor networks," *IEEE Syst. J.*, vol. 9, no. 3, pp. 1081–1090, Sep. 2015.

[6] I. Martinez-Alpiste, G. Golcarenarenji, Q. Wang, and J. M. Alcaraz-Calero, "Search and rescue operation using UAVs: A case study," *Expert Syst. Appl.*, vol. 178, 2021, Art. no. 114937.

[7] E. U. Acar, H. Choset, Y. Zhang, and M. Schervish, "Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods," *Int. J. Robot. Res.*, vol. 22, no. 7/8, pp. 441–466, 2003.

[8] J. He, Y. Zhou, L. Huang, Y. Kong, and H. Cheng, "Ground and aerial collaborative mapping in urban environments," *IEEE Robot. Autom. Lett.*, vol. 6, no. 1, pp. 95–102, Jan. 2021.

[9] B. Wang, Z. Liu, Q. Li, and A. Prorok, "Mobile robot path planning in dynamic environments through globally guided reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 6932–6939, Oct. 2020.

[10] M. Dadgar, S. Jafari, and A. Hamzeh, "A PSO-based multi-robot cooperation method for target searching in unknown environments," *Neurocomputing*, vol. 177, pp. 62–74, 2016.

[11] J. Ni, L. Yang, P. Shi, and C. Luo, "An improved DSA-based approach for multi-AUV cooperative search," *Comput. Intell. Neurosci.*, vol. 2018, 2018, Art. no. 2186574.

[12] Y. Zhou, H. Hu, Y. Liu, S.-W. Lin, and Z. Ding, "A distributed approach to robust control of multi-robot systems," *Automatica*, vol. 98, pp. 1–13, 2018.

[13] F. Zhang, B. Chen, and X. Ban, "Multi-robot cooperative search algorithm based on bio-inspired neural network and DMPC," *Control Decis.*, vol. 36, no. 11, pp. 2699–2706, 2021.

[14] P. DeLima, G. York, and D. Pack, "Localization of ground targets using a flying sensor network," in *Proc. IEEE Int. Conf. Sens. Netw., Ubiquitous, Trustworthy Comput.*, 2006, vol. 1, pp. 194–199.

[15] X. Fu, G. Wei, and X. Gao, "Cooperative area search algorithm for multi-UAVs in uncertainty environment," *Syst. Eng. Electron.*, vol. 38, pp. 821–827, 2016.

[16] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robot. Auton. Syst.*, vol. 61, no. 12, pp. 1258–1276, 2013.

[17] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1985, vol. 2, pp. 116–121.

[18] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE J. Robot. Autom.*, vol. RA-3, no. 3, pp. 249–265, Jun. 1987.

[19] X. Cao and C. Sun, "Cooperative target search of multi-robot in grid map," *Control Theory Appl.*, vol. 35, no. 3, pp. 273–282, 2018.

[20] Y. Hou, X. Liang, L. He, and L. Liu, "Cooperative area search algorithm for UAV swarm in unknown environment," *J. Beijing Univ. Aeronaut. Astronaut.*, vol. 45, no. 2, pp. 347–356, 2019.

[21] E. Rolf, D. Fridovich-Keil, M. Simchowitz, B. Recht, and C. Tomlin, "A successive-elimination approach to adaptive robotic source seeking," *IEEE Trans. Robot.*, vol. 37, no. 1, pp. 34–47, Feb. 2021.

[22] Y. Kantaros, B. Schlotfeldt, N. Atanasov, and G. J. Pappas, "Sampling-based planning for non-myopic multi-robot information gathering," *Auton. Robots*, vol. 45, no. 7, pp. 1029–1046, 2021.

[23] X. Zheng, S. Jain, S. Koenig, and D. Kempe, "Multi-robot forest coverage," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 3852–3857.

[24] N. Hazon, F. Mieli, and G. A. Kaminka, "Towards robust on-line multi-robot coverage," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 1710–1715.

[25] G.-Q. Gao and B. Xin, "A-STC: Auction-based spanning tree coverage algorithm formotion planning of cooperative robots," *Front. Inf. Technol. Electron. Eng.*, vol. 20, no. 1, pp. 18–31, 2019.

[26] W. Dong, S. Liu, Y. Ding, X. Sheng, and X. Zhu, "An artificially weighted spanning tree coverage algorithm for decentralized flying robots," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 4, pp. 1689–1698, Oct. 2020.

[27] V. G. Nair and K. Guruprasad, "MR-SimExCoverage: Multi-robot simultaneous exploration and coverage," *Comput. Elect. Eng.*, vol. 85, 2020, Art. no. 106680.

[28] W. Zhang, J. Liu, L.-b. Fan, Y.-h. Liu, and D. Ma, "Control strategy PSO," *Appl. Soft Comput.*, vol. 38, pp. 75–86, 2016.

[29] T. A. Runkler, "Wasp swarm optimization of the c-means clustering model," *Int. J. Intell. Syst.*, vol. 23, no. 3, pp. 269–285, 2008.

[30] M. Amir, S. Bedra, S. Benkouda, and T. Fortaki, "Bacterial foraging optimisation and method of moments for modelling and optimisation of microstrip antennas," *IET Microw., Antennas Propag.*, vol. 8, no. 4, pp. 295–300, 2014.

[31] X.-B. Meng, X. Z. Gao, Y. Liu, and H. Zhang, "A novel bat algorithm with habitat selection and Doppler effect in echoes for optimization," *Expert Syst. Appl.*, vol. 42, no. 17/18, pp. 6350–6364, 2015.

[32] G. Zhang and J. Zeng, "Area coverage algorithm in swarm robotics based on wasp swarm algorithm," *Pattern Recognit. Artif. Intell.*, vol. 24, pp. 431–437, 2011.

[33] Y. Cai and S. X. Yang, "An improved PSO-based approach with dynamic parameter tuning for cooperative multi-robot target searching in complex unknown environments," *Int. J. Control*, vol. 86, no. 10, pp. 1720–1732, 2013.

[34] B. Yang, Y. Ding, Y. Jin, and K. Hao, "Self-organized swarm robot for target search and trapping inspired by bacterial chemotaxis," *Robot. Auton. Syst.*, vol. 72, pp. 83–92, 2015.

[35] H. Tang, W. Sun, H. Yu, A. Lin, and M. Xue, "A multirobot target searching method based on bat algorithm in unknown environments," *Expert Syst. Appl.*, vol. 141, 2020, Art. no. 112945.

[36] J. Ni and S. X. Yang, "Bioinspired neural network for real-time cooperative hunting by multirobots in unknown environments," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2062–2077, Dec. 2011.

[37] X. Cao, D. Zhu, and S. X. Yang, "Multi-AUV target search based on bioinspired neurodynamics model in 3-D underwater environments," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2364–2374, Nov. 2016.

[38] C. Luo, S. X. Yang, X. Li, and M. Q.-H. Meng, "Neural-dynamics-driven complete area coverage navigation through cooperation of multiple mobile robots," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 750–760, Jan. 2017.

[39] C. Luo and S. X. Yang, "A real-time cooperative sweeping strategy for multiple cleaning robots," in *Proc. IEEE Int. Symp. Intell. Control*, 2002, pp. 660–665.

[40] C. Luo, S. X. Yang, and D. A. Stacey, "Real-time path planning with deadlock avoidance of multiple cleaning robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, vol. 3, pp. 4080–4085.

[41] Z. Huang, D. Zhu, and B. Sun, "A multi-AUV cooperative hunting method in 3-D underwater environment with obstacle," *Eng. Appl. Artif. Intell.*, vol. 50, pp. 192–200, 2016.

[42] X. Cao, H. Sun, and G. E. Jan, "Multi-AUV cooperative target search and tracking in unknown underwater environment," *Ocean Eng.*, vol. 150, pp. 1–11, 2018.

[43] E. Fabrizi and A. Saffiotti, "Augmenting topology-based maps with geometric information," *Robot. Auton. Syst.*, vol. 40, no. 2/3, pp. 91–97, 2002.

[44] N. Tomatis, I. Nourbakhsh, and R. Siegwart, "Simultaneous localization and map building: A global topological model with local metric maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Expanding Societal Role Robot. Next Millennium*, 2001, vol. 1, pp. 421–426.

[45] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte, "Information based adaptive robotic exploration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2002, vol. 1, pp. 540–545.

[46] S. X. Yang and M.-H. Meng, "Real-time collision-free motion planning of a mobile robot using a neural dynamics-based approach," *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1541–1552, Nov. 2003.

[47] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architectures," *Neural Netw.*, vol. 1, no. 1, pp. 17–61, 1988.

[48] E. F. Asadi and A. Richards, "Scalable distributed model predictive control for constrained systems," *Automatica*, vol. 93, pp. 407–414, 2018.

Bo Chen received the B.S. degree in automation from the East China University of Technology, Nanchang, China, in 2019. He is currently working toward the M.S. degree with the School of Electrical Engineering, Zhengzhou University, Zhengzhou, China.

His research interests include multirobot system, area coverage search, and path planning.

Wenli Zhang received the B.S. degree in rail transit signaling and control from the Linyi University of China, Linyi, China, in 2019. She is currently working toward the M.S. degree with the School of Electrical Engineering, Zhengzhou University, Zhengzhou, China.

Her research interests include multiagent, multirobot formation, and obstacle avoidance.

Fangfang Zhang received the B.E. and M.E. degrees in applied mathematics in 2008 and 2011, respectively, and the Ph.D. degree in control science and engineering in 2015, all from Shandong University, Jinan, China.

He is currently an Associate Professor with Zhengzhou University, Zhengzhou, China. His research interests include optimal control of multiagent systems, multirobot formation, and machine vision.

Yanhong Liu (Member, IEEE) received the B.E. degree from the Zhengzhou University of Light Industry, Zhengzhou, China, in 1992, and the M.E. and Ph.D. degrees in control science and engineering from Tsinghua University, Beijing, China, in 2002 and 2006, respectively.

From 2012 to 2013, she was a Visiting Scholar with the University of California at San Diego, La Jolla, CA, USA. She is currently a Professor with the School of Electrical Engineering, Zhengzhou University, Zhengzhou. Her research interests include nonlinear system modeling and control, robotic control, and human–robot interactions and collaborations.

Hongnian Yu (Senior Member, IEEE) received the B.Eng. degree in electrical and electronic engineering from the Harbin Institute of Technology, Harbin, China, the M.Sc. degree in control engineering from Northeast Heavy Machinery Institute, Heilongjiang, China, and the Ph.D. degree in robotics from King's College London, London, U.K.

He is a Professor and Head of Research with the School of Engineering and the Built Environment, Edinburgh Napier University, Edinburgh, U.K. He has held academic positions with University of Sussex, Brighton, U.K.; Liverpool John Moores University, Liverpool, U.K.; University of Exeter, Exeter, U.K.; University of Bradford, Bradford, U.K.; Staffordshire University, Stoke-on-Trent, U.K.; and Bournemouth University, Poole, U.K. He has extensive research experience in mobile computing, computer networks, control of robots, and neural networks and computing. He has authored or coauthored more than 200 journal and conference research papers.