The Application of the Multi-Agent Coverage and Self-Healing Control Based on a Swarm Intelligence SONM and Potential Function Approach

ZHENUA PAN\textsuperscript{a}, (Member, IEEE), LING SHU\textsuperscript{b}, HONGBIN DENG\textsuperscript{b}, (Member, IEEE), AND DONGFANG LI\textsuperscript{a}

Department of Electromechanical Engineering, Beijing Institute of Technology, Beijing 100081, China

Corresponding author: Hongbin Deng (denghongbin@bit.edu.cn)

This work was supported by the Joint Ph.D. Program of the Beijing Institute of Technology.

ABSTRACT Self-organizing network will probably be a priority choice for the multi-agent systems because of its good features of configuration, management, optimization and self-healing. In this paper, a swarm intelligence coverage and self-healing method based on the Self-Organizing Network Model (SONM) and potential function is proposed for our multi-agent system. The SNOM is presented with a hierarchical network structure, which mainly includes the link layer design, route establishment and route maintenance. Based on the SNOM, the multi-agent coverage and self-healing control law is devised, and the potential function is designed to realize the collision avoidance, coverage and self-healing control for the multi-agent systems. The proposed approaches are implemented and evaluated by extensive simulations and experiments, and the results confirm that it is feasibility and efficiency in multi-agent coverage and self-healing control.

INDEX TERMS Multi-agent system, self-organizing network model, potential function, self-healing control.

I. INTRODUCTION

With the development of the net-working technologies, more and more intelligent network models and structures are put forward [1]. Considering the multi-agent systems add more applications to coordinate their operations through communication networks, which are mainly used for searching and recovering operations [2], exploration [3], [4], surveillance [5], and environmental monitoring for pollution detection and estimation [6]. With the complexity unknown and dynamic environments and domains demands autonomous operation, the single agent in the multi-agent system is not perfect, various malfunctions usually occur due to their own components or external interference. Due to the coordination and interactions in a multi-agent system, the resulting susceptibility to failures is even worse, a single individual failure often causes the entire system to be paralyzed. Therefore, it becomes a significant and challenging task to establish, manage and maintain such a large-scale, mixed, dynamic network. The Self-Organizing Network Model (SONM) will most probably be a priority choice because of its principles, which can be described as the ability to arrange itself autonomously and spontaneously for a given task in the multi-agent systems [7], [8]. Furthermore, the SONM has the good performance of internal interactions, self-healing and structural organization [9].

Many researchers have attempted to employ self-organization network to cope with the multi-agent path planning [10], [11], formation control [12], coordinated control [13], [14], self-reconfigurable [15], [16] and the other complexity and unpredictability systems. They usually focus on how to create an efficient task distribution system for a given multi-agent system, ignore the key challenges of robustness and self-healing. Therefore, the coverage and self-healing control are becoming another important part of swarm intelligence system. The self-healing control technology research is to find a suitable method to kick the faulty agents out of the network, and the remaining agents can reorganize the network to form a new multi-agent system, which can continue to perform unfinished tasks. Multi-agent coverage control refers to design an algorithm that covers the multi-agent...
system as evenly as possible in the characteristic working area.

Kim et al. [17] proposed a SHAGE (self-healing, adaptive) framework to ensure that the agents can manage themselves. This method mainly including two parts. The first part contains several sets of components for system management, and the second part contains internal and external knowledge bases for storing architecture configuration instructions and components. In the process of intelligent work, these two parts cooperate with each other to observe the environmental conditions and continuously adjust the appropriate structure. However, this structure is not suitable for multi-agent systems. A structure-based adaptive method is proposed by Garlan et al. [18], the reusable infrastructure and mechanism tailored is provided to the needs of the domain. This specialization allows developers to customize aspects of the system, such as monitoring goals or adapting to conditions and actions. The adaptation is done through a statically associated action rules for each identified adaptation cause. The collection of information and the implementation of adaptation depend on the support of the target system. However, the architecture is sensitive to single agent failure and has limited use of multi-robot systems. Focuses on multi-robot systems, Parker and Kann proposed the LeaF (learning based fault diagnosis) [19], an adaptive causal model for fault diagnosis and recovery. Causal models represent expected failures and were initially created by developers at design time. Case-based reasoning can handle unexpected errors at runtime, the model can represent failures that occur only during the interaction between agents. However, this method does not deal with malfunction detection, nor does it explain how to integrate LeaF into the real system. In order to improve the reliability and practicability of multi-agent systems, Kirchner et al. [20] proposed a self-healing architecture system RoSHA for multi-agents. RoSHA is based on the established agent operating system ROS fault diagnosis, it contains five modules: condition monitoring, fault diagnosis, recovery planning, maintenance execution and knowledge base. Adopted abstract system models and Bayesian networks as knowledge bases to determine the reason of detected failures. However, this method cannot be used for network self-repair for multi-agent systems.

Currently, multi-agent coverage and self-healing control are rarely studied. As for the self-healing control, mainly including three typical categories. The first one is direct self-healing method [21], which assigns an individual to repair network topology. However, this method did not consider dynamic changes of the network topology in self-repair process. The second one is density self-healing method, which is widely used in multi-agent coverage tasks. Derbakova et al. [22] proposed a decentralized self-repair to maintain connectivity method for multi-agent coverage tasks in an interested region. Unfortunately, with the increase of the number of agents, this method will cause a large increase in communication traffic and lower self-healing efficiency. The third one is recursive self-healing method, which achieves the self-healing behavior by specified switched topology rules. Younis et al. [23] presented the Recovery through Inward Motion (RIM), a distributed algorithm to efficiently restore network connectivity after a node failure, and the entire self-healing process without external supervision. However, this method is limited to a large number of agents system. For multi-agent formation self-healing control, Ju et al. [24] proposed a switched topology control and distributed negotiation method, which can also be implemented in coverage and exploration tasks. However, the communication delays in this method is not ignored. Zhang and Chen [25] proposed a recursive and distributed topology for self-healing when mobile agents failure, an interaction dynamics model is established to illustrate the logical and physical topologies of the network. Moreover, a metric of the topology structure is designed to evaluate the self-healing performance. Unfortunately, this method had no consideration of collisions between agents. To overcome the problems of the swarm behavior associated with failed robots, Timmis et al. [26] utilized an immune-inspired swarm aggregation algorithm for the multi-agent self-healing control, by developing an immune inspired method to permit the system recovery from certain failure modes. Shen et al. [27] proposed the Digital Hormone Model (DHM) as a bio-inspired distributed control method for the multi-agent self-organization, which has demonstrated the implementation in simulation for swarm actions such as searching and seizing targets, distributing and forming sensor networks, self-repairing, and avoiding pitfalls by detouring. However, further research still needed for these methods to apply to practice.

In terms of multi-agent coverage control, based on honeybee colony behavior, a bee pheromone signaling approach has been applied to the multi-agent coverage issues [28], which had realized the robots to maximise cover the total working area. By analyzing the problems of Voronoi coverage with non-convex environments, Breitenmoser et al. [29] designed a navigation algorithm based on the Lloyd and the Tangent Bug algorithm, and guaranteed a team of robots to converge to a local area. However, they have not considered the self-healing of the system. When some agent nodes fail, the entire system will not work properly.

In this paper, a swarm intelligence coverage and self-healing control method based on the Self-Organizing Network Model (SONM) and potential function is proposed and explored, it can enable a group of robots to set up a network quickly in the working environment. The tasks of this paper mainly including three aspects. (a). Designed an efficient self-organizing network model, which can implement network self-repairing and fast self-organizing. (b). Proposed a coverage and self-healing control method based on the potential function to guarantee the collision avoidance for the swarm robots. (c). The proposed method was verified on both the simulation and real experimental platform, and both achieved good performance. Compared with the existing literature on coverage and self-healing control for the multi-agent systems, the main contributions of our work are...
summarized as follows. (1). We solved the coverage control with the shortest path for the multi-agent systems, and if some robots fail, instead of controlling the specific robots to replace the disappearing ones [25], the system can fast self-organizing the network and manage the remaining robots to maximize distribute in the working area to finish the self-healing, which including on the network and physical. (2). We designed the hierarchical structure self-organizing network model, which has good scalability, and the amount of routing control information is greatly reduced in the network. Moreover, the hybrid multiple access strategy combining Carrier Sense Multiple Access (CSMA) with dynamic reservation Time Division Multiple Address (TDMA) will improve the efficiency of network channel access and satisfy the real-time transmission requirements. (3). Based on the potential function and SONM, we designed a coverage and self-healing control method, which can organize randomly distributed agents to cover the entire interest area with the shortest path. When an agent node disappears due to failure of critical nodes, hostile environmental conditions, hardware malfunction, or battery failure. All remaining agents can readjust their positions according to the coverage and self-healing algorithm to maximize cover the interest area again. The coverage of the work area can achieve repeated self-healing repeatedly multiple times. Moreover, this method can also avoid the collisions between robots.

In the next section, the SONM is designed, the route algorithm layout, route establishment and route maintenance are detailed. Afterwards, the coverage and self-healing control law is proposed in Section III. The following section shows the results obtained both through simulations and hardware experiments, as well as a discussion of the results to demonstrate the performance of the proposed method. Finally, the article ends with conclusions and open issues for future research.

II. SELF-ORGANIZING NETWORK MODEL

A. THE NETWORK STRUCTURE

A reasonable group architecture can enable fast self-organization, effective cooperation, better fault tolerance, flexibility and adaptability for the multi-agent systems [30]. At present, the network structures of the multi-agent systems are mainly divided into the centralized and decentralized. The decentralized ones can be further divided into hierarchical and distributed. In this paper, a hierarchical structure is adopted to form a network between the monitor station, cluster nodes and slave nodes to complete the positioning and communication between the robots, as shown in Fig.1, the cluster nodes are with dual wireless channel, and they have the ability to communicate with the monitor station, other cluster nodes and their slave nodes, all of the cluster nodes and the monitor station will form the primary network. While the slave nodes just have the ability to communicate with their cluster nodes, and when the slave nodes received the build chain command from their cluster nodes, they will construct into a secondary network with their cluster nodes,
and each subnet is linked to the monitor station by its cluster node. Thus, the network construction of the multi-agent system is realized. This hierarchical network structure has several advantages as follows: (a). It has good scalability, and the size of network is unlimited. (b). The amount of routing control information is greatly reduced in the network, and it is not only suitable for the large scale network, but also can provide a certain service quality assurance system.

B. THE LINK LAYER

In practical use, in order to provide reliable data transfer services for the multi-agent system, each robot is equipped with GPS positioning and 2.4GHz and 433M RF wireless communication modules, which is widely used in various industries. Thus, every robot has the function of self-localization and ranging. In order to improve the efficiency of network channel access and accuracy of positioning, and for the mobile self-organized network to satisfy the real-time and non real-time transmission requirements, this paper adopted the hybrid multiple access strategy combining CSMA with dynamic reservation TDMA. The CSMA mechanism to ensure fair efficient access to each node in the network channel. While the TDMA dynamic reservation mechanism to ensure real-time business nodes to access channel on time without any collisions.

1) ROUTE ESTABLISHMENT

After all of the robots start up, the initial state is networking, the monitor station initiates the networking request instruction to the cluster nodes periodically, and then listens for the channel, when the cluster nodes receive the instruction, they will establish communication with the monitor station, until all the cluster nodes have built the connection, and the primary network is finished. Afterwards, the cluster nodes will initiate the networking request instructions to the slave nodes periodically, when the slave nodes receive the instruction, they will accept the network and send the current coordinates to the cluster nodes, if the cluster nodes have reached the maximum number of networking, it will stop the networking request instruction and no longer establish a connection with other slave nodes, then the secondary network between the cluster nodes and their slave nodes are built, the current coordinates of all the robots are sent to the monitor station directly or indirectly, and the monitor station will build the network information table and sent it to each individual. If some robot nodes fail during the normal actions, the system will organize the network to restore normal structure and functions.

Remark 1: Slave node loss, if the monitor station cannot receive coordinate information from a slave node 10 times, we will determine that the node is lost and remove it from the network.

Remark 2: Cluster node loss, in this case, the monitor station will does not receive coordinate information of the cluster node and its slave nodes, then the monitor station will start the network repair command, send a cluster node activation instruction to a slave node, then a new cluster node will be generated, which will built the network connection with the other slave nodes.

III. THE COVERAGE AND SELF-HEALING CONTROL DESIGN

After the multi-agent systems have finished networking, the robots are supposed to be uniform distributed in the working environment, if there are some robots lost, the system will self-healing and try to evenly coverage the working area.

A. THE COVERAGE CONTROL METHOD

The concept of coverage is a metric for evaluating robotic systems, first introduced by Gage [31], he defines the basis types of blanket coverage, where is objective to achieve the robots maximise coverage the working environment. At first, the robots are randomly distributed in a certain area, as shown in Fig.2, we can calculate the optimal uniformly distributed coordinates for the robots, then these coordinates are sent

![FIGURE 2. The randomly distributed robots in a certain area.](image-url)
to every robot, all of the robots are supposed to reach the nearest targets. The potential function method is used to solve the collisions between the robots. The force analysis of the robots is shown in Fig. 3, robot \( R_i \) is attracted by \( F_{att} \), which is produced by the targets and direct to target, when the robot \( R_i \) approaches \( R_j \), the distance \( d < r_{rep} \), robot \( R_i \) will suffer the repulsive force \( F_{ij} \) from \( R_j \), the direction is away from \( R_j \). \( F_a \) is the resultant force of \( F_{att} \) and \( F_{ij} \), and the robots are supposed to move toward the direction of \( F_a \) until reached the targets.

![FIGURE 3. The stress analysis of the robots.](image)

where we assume \( \mathbf{q} \) is the coordinate of robot \( (x, y)^T \), \( r_{rep} \) represents the maximum value of the repulsive potential influence range, the commonly potential function field can be expressed as

\[
U(\mathbf{q}) = U_{att}(\mathbf{q}) + \sum_{j=1}^{n} U_{ij}(\mathbf{q})
\]

where \( U_{att}(\mathbf{q}) \) represents the attractive potential, \( U_{ij}(\mathbf{q}) \) denotes the repulsive potential. \( n \) is the number of the other robots within the scope of influence. Consequently, the negative gradient of the potential function field is defined as the potential force

\[
F(\mathbf{q}) = -\nabla U(\mathbf{q})
\]

\[
= -\nabla U_{att}(\mathbf{q}) - \nabla \sum_{j=1}^{n} U_{ij}(\mathbf{q})
\]

\[
= F_{att}(\mathbf{q}) + \sum_{j=1}^{n} F_{ij}(\mathbf{q})
\]

where \( F_{att}(\mathbf{q}) \) represents the attractive force, \( F_{rep}(\mathbf{q}) \) denotes the repulsive force.

The attractive potential is given as

\[
U_{att}(\mathbf{q}) = \frac{1}{2} k_t (\mathbf{q} - \mathbf{q}_t)^2
\]

where \( k_t \) is a positive coefficient, \( \mathbf{q}_t = (x_t, y_t)^T \) is the location vector of target.

The attractive force towards zero as the robots approach the targets which can be expressed as

\[
F_{att}(\mathbf{q}) = -\nabla U_{att}(\mathbf{q}) = -k_t (\mathbf{q} - \mathbf{q}_t)
\]

The components of \( F_{att}(\mathbf{q}) \) can be written as presented below

\[
\begin{align*}
F_{att}(x) &= k_f (x - x_t) \\
F_{att}(y) &= k_f (y - y_t)
\end{align*}
\]

where \( F_{att}(x) \), \( F_{att}(y) \) are the attractive forces of \( x \) and \( y \) axis direction respectively.

When the robot \( R_i \) approaches to \( R_j \), the repulsive potential will be generated to avoid collisions, which is designed as

\[
U_{ij}(q_{ij}) = \begin{cases} 
-k_{rep} \left( \ln \left( q_{ij} - r \right) - \frac{q_{ij}}{r_{rep} - r} \right), & q_{ij} \leq r_{rep} \cap q_{ij} \neq r \\
0, & q_{ij} > r_{rep}
\end{cases}
\]

where the \( k_{rep} \) is a positive coefficient, \( q_{ij} \) is the distance of \( R_i \) and \( R_j \), it can be represented as \( q_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \), \( r \) represents as the safe distance between the robots.

From the (6), the repulsive force can be expressed as

\[
F_{ij}(q_{ij}) = -\nabla U_{ij}(q_{ij}) = \begin{cases} 
k_{rep} \left( \frac{1}{q_{ij} - r} - \frac{1}{r_{rep} - r} \right), & q_{ij} \leq r_{rep} \cap q_{ij} \neq r \\
0, & q_{ij} > r_{rep}
\end{cases}
\]

Similar to the attractive force, \( F_{rep} \) can be written as

\[
F_{ij}(x) = \begin{cases} 
k_{rep} \left( \frac{1}{q_{ij} - r} - \frac{1}{r_{rep} - r} \right) \cos \theta_{ij}, & q_{ij} \leq r_{rep} \cap q_{ij} \neq r \\
0, & q_{ij} > r_{rep}
\end{cases}
\]

\[
F_{ij}(y) = \begin{cases} 
k_{rep} \left( \frac{1}{q_{ij} - r} - \frac{1}{r_{rep} - r} \right) \sin \theta_{ij}, & q_{ij} \leq r_{rep} \cap q_{ij} \neq r \\
0, & q_{ij} > r_{rep}
\end{cases}
\]

where \( \theta_{ij} \) is the angle of the repulsive force \( F_{ij} \)

\[
\theta_{ij} = \begin{cases} 
\arcsin \left( \frac{y_i - y_j}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \right), & y_i > y_j \\
2\pi - \arcsin \left( \frac{y_i - y_j}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} \right), & y_i < y_j
\end{cases}
\]

From the (6) and (7), The potential and force between the two agents are shown in Fig.4.

Remark 3: As shown in Fig.4, it clearly shows that the value of repulsive force is extremely large when the distance between two robots approach to the safe distance \( r \). Therefore, the distance between the two robots can not be less than the safe distance theoretically, and the robots have no chance to collide.

Based on the potential function, the coverage control method is designed as shown in algorithm1.
where the \( d_b \) is the distance of the robots and the area boundaries.

\begin{align*}
U_{br} (d_b) &= \begin{cases} 
-k_b \log \left( \frac{d_b}{r_{rep}} \right) + k_b d_b, & 0 < d_b < r_{rep} \\
0, & d_b \geq r_{rep}
\end{cases} \tag{11}
\end{align*}

where the \( k_b \) is a positive coefficient.

**Algorithm 1 The Coverage Control Algorithm**

1: Initialize the robots \( R_n \), the target coordinates \( T_n \);
2: for each target \( T_i \), \( i \in [1, n] \) do
3: \hspace{1em} for each robot \( R_j \), \( j \in [1, n] \) do
4: \hspace{2em} Calculate the distance \( d_{ij} \) between the target \( T_i \) and every robot \( R_j \) respectively, \( \min(d_{ij}) \), and assign the target \( T_i \) to the nearest robot
5: \hspace{1em} end for
6: end for
7: for each robot \( R_j \), \( j \in [1, n] \) do
8: \hspace{1em} Calculate the \( F_{att} \) and \( F_{ij} \) based the Eqs.(4) and (7) for the robot \( R_j \).
9: \hspace{1em} The resultant force \( F_j = F_{att} + F_{ij} \)
10: \hspace{1em} Calculate the distance \( d_{ij} \) between the target \( T_i \) and every robot \( R_j \)
11: \hspace{1em} while \((F_j = 0)\) and \((d_{ij} = 0)\) do
12: \hspace{2em} Update the coordinates of \( R_j \)
13: \hspace{1em} end while
14: end for

The resultant force \( F_{br} \) from the boundaries are the negative gradient of the resultant potential \( U_{br} \)

\begin{align*}
F_{br} (d_b) &= -\nabla U_{br} (d_b) \\
&= \begin{cases} 
-k_b r_{rep}, & 0 < d_b < r_{rep} \\
0, & d_b \geq r_{rep}
\end{cases} \tag{12}
\end{align*}

where the direction of \( F_{br} \) is inward perpendicular to the boundaries. From the (11) and (12), the potential and force between the robots and working area boundaries are shown in Fig.7.

**Remark 4:** From the Fig.7(b), it is obvious that the value of repulsive force is extremely large when the robots approach to the area boundaries. Therefore, the robots can not be rejected outside the coverage area. While the repulsive force between the robots can be obtained based the Eq.8, and to ensure that the robots can cover the working area as evenly as possible, the repulsive potential maximum influence range \( r_{rep} \) and safe distance \( r \) of the robots can be adjusted as we need.

We assume that the robots can maximise cover the working area and reach a new balance, and each robot will occupy the same private area, which can be treated as a circular. Therefore, the multi-agent systems working area can be expressed as

\begin{align*}
S_w \approx n \pi r_{rep}^2 \tag{13}
\end{align*}

where \( n \) represents the current number of robots in the working area.

Thus, the value of the repulsive potential maximum influence range \( r_{rep} \) of the robots can be set as

\begin{align*}
r_{rep} = \sqrt{\frac{S_w}{n \pi}} \tag{14}
\end{align*}

Fig.8 shows the self-healing control flowchart for the multi-agent systems, when the system has finished the initialization and networking, the monitor station will collect
the status information of the robots through the wireless communication module. If one or more robot nodes fail, the system will start the network repairing command to restore the normal network structure. Otherwise, every robot will update the status information and send to the monitor, and the virtual resultant force $F_i$ of each robot is calculated, every robot will move toward the direction of $F_i$, until $F_i$ tends to 0.
IV. THE SIMULATIONS AND EXPERIMENTS

To evaluate the performance of the proposed strategy and algorithms, several simulations and experiments have been implemented for multi-agent coverage and self-healing control. Our multi-agent experimental platform consists of 20 two-wheeled agents, as shown in Fig.9.

A. THE MULTI-AGENT COVERAGE CONTROL

This section makes an analysis for multi-agent coverage control both through the simulations and real experiments. At the beginning, a group of robots are distributed in the working area randomly, when the multi-agent system completes the initialization and networking, every robot is supposed to reach its nearest target without any collisions with the other robots. In the simulation, the corresponding parameters of the potential field is: $k_t = 15$, $k_{rep} = 60$, the safe distance between the agents is $r = 2$, the maximum influence range of repulsive potential $r_{rep} = 3$. As shown in Fig.10(a), 20 robots are random distributed in the working area $S_1 \{(x, y) | 0 \leq x \leq 100, 0 \leq y \leq 30\}$, and 20 targets can be calculated for the 20 robots, all the robots are supposed to reach the nearest targets based on the coverage control algorithm. From the Fig.10(a), all robots can reach their targets without any collisions. Fig.10(b) shows the regional coverage of 50 robots. At the beginning, the 50 robots are distributed in a working area $S_2 \{(x, y) | 0 \leq x \leq 100, 0 \leq y \leq 60\}$ randomly, and 50 targets was calculated based on the coverage control algorithm, then each robot will move toward its nearest target without any collisions. At last, all robots stopped at their targets, and it shows that the group of robots have a good performance in regional coverage.

We also did the experiment to prove the reliability and effectiveness of the proposed method, as shown in Fig.11(a). The corresponding parameters of the potential field is:$k_t = 5$, $k_{rep} = 40$, the safe distance between the agents is $r = 0.5m$, the maximum influence range of repulsive potential $r_{rep} = 2m$. At first, 11 robots are distributed in the working area randomly, and they are supposed to evenly distributed with a line (12m) as the simulation in Fig.10(a). After the networking, every robot moves toward its nearest target without any collisions, as shown in Fig.11(b)- Fig.11(h). Finally, all the robots evenly distributed with a line and terminate there, as shown in Fig.11(i). In the experiment, there was one robot do not move, because this robot failed to network blame on not powered on.

Fig.12(a)- Fig.12(i) shows the results of regional coverage with 12 robots. The group of robots start up with a linear distribution, and we want the robots to maximize cover the working area (length:12m, width:5m) as in Fig.10(b).
FIGURE 11. Real experiment 1: The multi-agent distributed with a line.

When the system finished networking, each robot moves toward its nearest target with the shortest path, and stops after reaching its target. Finally, it shows that the 12 robots have evenly distributed in the working area.

B. MULTI-AGENT SELF-HEALING

In this section, we will analyze the proposed self-healing algorithm from both simulation and practical experiment. When the multi-agent system has finished networking and evenly distributed in the working environment, and if some robot nodes fail, the multi-agent systems are supposed to self-organizing the network and self-healing the coverage formation. Fig.13 shows the process of multi-agent linear self-healing control, where $J$ is the number of simulation steps. When 4 robots fail after the coverage control, the system needs self-organizing the network based on the SONM, and the robots need to adjust their current positions to restore the linear coverage formation based on the proposed self-healing method. From the Fig.13, it shows that the robots move toward the empty area as the number of simulation steps increasing, until all the robots reach a new balance and end. In the simulation, the corresponding parameters of the potential field is: $k_t = 0$, $k_{rep} = 60$, the safe distance between the agents is $r = 2$, the maximum influence range of repulsive potential $r_{rep} = 6$.

Fig.14(a)-Fig.14(i) show the real experiment of the multi-agent self-healing with 16 robots. At the beginning, the robots evenly distributed in the working environment with a line (Fig.14(a)), if four of the robots are turned off and taken away (Fig.14(b) and Fig.14(c)). During the maintenance of the route, the remaining robots will find that 4 robots disappeared, then the system will self-organize the network. Afterwards, all the robots move toward the empty area based the potential function method, until the robots reach a new balance (Fig.14(i)), complete the self-healing and end. In the experiment, the corresponding parameters of the potential field is: $k_t = 0$, $k_{rep} = 40$, the safe distance between the agents is $r = 0.5m$, the maximum influence range of repulsive potential $r_{rep} = 1m$. 

FIGURE 12. Real experiment 2: The results of a regional coverage.
FIGURE 13. The simulation of multi-agent linear self-healing.

FIGURE 14. Real experiment 3: The linear self-healing of the multi-agent.

FIGURE 15. The simulation of multi-agent self-healing in a regional environment.

Fig. 15 shows the simulation of regional self-healing with 50 robots, when the robots evenly distributed in the working area, then 12 robots fail, and the remaining robots are supposed to start self-healing and evenly distributed in the working area. Fig. 15(b) shows the performance of the regional self-healing control, each robot calculates its way-point with the potential function method until the system reaches a new balance and end. In the simulation, the corresponding parameters of the potential field is: $k_t = 0$, $k_{rep} = 60$, the safe distance between the agents is $r = 2$, the maximum influence range of repulsive potential $r_{rep} = 7$.

Fig. 16(a)-Fig. 16(i) show the real experiment of the multi-agent self-healing with 20 robots. At the beginning, the robots evenly distributed in the working area, then 4 robots were removed as shown in Fig. 16(b), the system self-organized the network and started the self-healing algorithm quickly, and the remaining robots move toward the empty area (Fig. 16(c)-Fig. 16(e)), until all the robots reach a new balance and stopped. To test the reliability and stability of the proposed method, we removed another 4 robots, as shown in Fig. 16(f), the system self-organized the network and started the self-healing again, the remaining robots try to cover the working area as much as possible. In the experiment, the corresponding parameters of the potential field is: $k_t = 0$, $k_{rep} = 40$, the safe distance between the agents is $r = 0.5m$, the maximum influence range of repulsive potential $r_{rep}$ is calculate by formula (14). From the Fig. 16, the performance of the self-healing is not very well, there are three main reasons why there is no uniform coverage as in the simulation. (1) The experimental site is small, and less robots. (2) The positioning accuracy of the robot is not enough. (3) The speed control of the robots is unstable. It will be better as the improvement of the hardware and experiment environment.
Remark 5: Compare with the method in [22], the proposed method in this paper did not need the map of the working area and the positions of the other agents. The agents just adjust the safe distance between the agents according to the size of the working area and the number of agents in the system. Then, the agent automatically adjusts its position through the self-healing algorithm to complete the uniform coverage of the entire target area. Furthermore, the proposed method can ensure that the coverage of the work area can achieve repeated self-healing multiple times, will improve the robustness of the system. We also consider the collision avoidance between the agents, to ensure the system has strong environmental adaptability. Compare with the method of literature [26], we have applied the multi-agent self-healing control method to practice successfully, and the methods proposed in the literature [26] have yet to be verified in practice. In addition, their method cannot be applied to multi-agent coverage control, and the collision avoidance is ignored.

V. CONCLUSION

This paper has presented a multi-agent coverage and self-healing method based on a swarm intelligence SONM and the potential function. To ensure the multi-agent system could fast self-positioning and self-organizing, the hybrid multiple access strategy combining CSMA with dynamic reservation TDMA, the route control algorithm has been designed for improving the efficiency and self-healing of the network. We have designed the potential function for the multi-agent motion control, collisions avoidance, coverage and self-healing control. To verify the reliability and validity of the proposed method, we have did some simulations in matlab and experiments on our multi-agent platform, and the performance of results demonstrate the effectiveness of the proposed approaches. Furthermore, we plan to optimize and extend the algorithm to ensure the positioning accuracy of the robots, and apply this method to the cooperative control for the multi-agent systems.

REFERENCES

[1] B. S. Northcote and D. E. Smith, “Service control point overload rules to protect intelligent network services,” IEEE/ACM Trans. Netw., vol. 6, no. 1, pp. 71–81, Feb. 1998.
[2] J. L. Baxter, E. Burke, J. M. Garibaldi, and M. Norman, “Multi-robot search and rescue: A potential field based approach,” in Autonomous Robots and Agents. Berlin, Germany: Springer, 2007, pp. 9–16.
[3] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, “Coordinated multi-robot exploration,” IEEE Trans. Robot., vol. 21, no. 3, pp. 376–386, May 2005.
[4] W. Sheng, Q. Yang, J. Tan, and N. Xi, “Distributed multi-robot coordination in area exploration,” Robot. Auto. Syst., vol. 54, no. 12, pp. 945–955, Dec. 2006.
[5] M. Saska, V. Vonásek, J. Chudoba, J. Thomas, G. Loianno, and V. Kumar, “Swarm distribution and deployment for cooperative surveillance by micro-aerial vehicles,” J. Intell. Robotic Syst., vol. 84, nos. 1–4, pp. 469–492, Dec. 2016.
[6] B. Bayat, N. Crasta, A. Cresspi, A. M. Pascoal, and A. Jipsaert, “Environmental monitoring using autonomous vehicles: A survey of recent searching techniques,” Current Opinion Biotechnol., vol. 45, pp. 76–84, Jun. 2017.
[7] T. Zhao, P. Li, and J. Cao, “Soft sensor modeling of chemical process based on self-organizing recurrent interval type-2 fuzzy neural network,” ISA Trans., vol. 84, pp. 237–246, Jan. 2019.
[8] W. R. Ashby, “Principles of the self-organizing system,” in Facets of Systems Science (International Federation for Systems Research International Series on Systems Science and Engineering), vol. 7. Boston, MA, USA: Springer, 1991.
[9] S. Bosse and A. Lechleiter, “A hybrid approach for structural monitoring with self-organizing multi-agent systems and inverse numerical methods in material-embedded sensor networks,” Mechatronics, vol. 34, pp. 12–37, Mar. 2016.
[10] D. Zhu, H. Huang, and S. X. Yang, “Dynamic task assignment and path planning of multi-AUV system based on an improved self-organizing map and velocity synthesis method in three-dimensional underwater workspace,” IEEE Trans. Cybern., vol. 43, no. 2, pp. 504–514, Apr. 2013.
[11] Y. Dai, Y. Kim, S. Wee, D. Lee, and S. Lee, “A switching formation strategy for obstacle avoidance of a multi-robot system based on robot priority model,” ISA Trans., vol. 56, pp. 123–134, May 2015.
[12] Y. Dai, Y. Kim, S. Wee, D. Lee, and S. Lee, “Symmetric caging formation for convex polygonal object transportation by multiple mobile robots based on fuzzy sliding mode control,” ISA Trans., vol. 60, pp. 321–332, Jan. 2016.
[13] M. Ma, L. Shao, and X. Liu, “Coordinated control of micro-grid based on distributed moving horizon control,” ISA Trans., vol. 76, pp. 216–223, May 2018.
[14] G. Lozenguez, L. Adouane, A. Beynier, A.-I. Mouaddib, and P. Martinet, “Punctual versus continuous auction coordination for multi-robot and multi-task topological navigation,” Auto. Robots, vol. 40, no. 4, pp. 599–613, Apr. 2016.

[15] M. Nilsson, “Connectors for self-reconfiguring robots,” IEEE/ASME Trans. Mechatronics, vol. 7, no. 4, pp. 473–474, Dec. 2002.

[16] B. Jakimovski, B. Meyer, and E. Maehle, “Swarm intelligence for self-reconfiguring walking robot,” in Proc. IEEE Swarm Intell. Symp., Sep. 2008, pp. 1–8.

[17] D. Kim, S. Lee, S. Park, Y. Jin, H. Chang, Y.-S. Park, I.-Y. Ko, K. Lee, J. Lee, and Y.-C. Park, “SHAGE: A framework for self-managed robot software,” in Proc. Int. Workshop Self-Adaptation Self-Managing Syst. (SEAMS), Shanghai, China: ACM Press, 2006, pp. 79–85.

[18] D. Garlan, S.-W. Cheng, A.-C. Huang, B. Schmerl, and P. Steenkiste, “Rainbow: Architecture-based self-adaptation with reusable infrastructure,” Computer, vol. 37, no. 10, pp. 46–54, Oct. 2004.

[19] L. Parker and B. Kannan, “Adaptive causal models for fault diagnosis and recovery in multi-robot teams,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Oct. 2006, pp. 2703–2710.

[20] D. Kirchner, S. Niemczyk, and K. Geihs, RoSHA: A Multi-Robot Self-Healing Architecture. Berlin, Germany: Springer-Verlag, Jan. 2014, pp. 304–315.

[21] K. Tomita, S. Murata, H. Kurokawa, E. Yoshida, and S. Kokaji, “Self-assembly and self-repair algorithm for a distributed mechanical system,” IEEE Trans. Robot. Autom., vol. 33, no. 5, pp. 1035–1045, Jan. 2000.

[22] A. Derbakova, N. Correll, and D. Rus, “Decentralized self-repair to maintain connectivity and coverage in networked multi-robot systems,” in Proc. IEEE Int. Conf. Robot. Autom., May 2011, pp. 3863–3868.

[23] M. Younis, S. Lee, S. Gupta, and K. Fisher, “A localized self-healing algorithm for networks of moveable sensor nodes,” in Proc. IEEE Global Telecommun. Conf. (GLOBECOM), Nov. 2008, pp. 1–5.

[24] J. Ju, Z. Liu, and W. Chen, “Switched topology control and negotiation of distributed self-healing for mobile robot formation,” Comput. Intell., Netw., vol. 462, pp. 562–574, Sep. 2014.

[25] F. Zhang and W. Chen, “Self-healing for mobile robot networks with motion synchronization,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Oct. 2007, pp. 3107–3112.

[26] J. Timmis, A. R. Ismail, J. D. Bjerke, and A. F. T. Winfield, “An immune-inspired swarm aggregation algorithm for self-healing swarm robotic systems,” Biosystems, vol. 146, pp. 60–76, Aug. 2016.

[27] W.-M. Shen, P. Will, A. Galstyan, and C.-M. Chuong, “Hormone-inspired self-organization and distributed control of robotic swarms,” Auto. Robots, vol. 17, no. 1, pp. 93–105, Jul. 2004.

[28] I. Caliskanelli, B. Broecker, and K. Tuyls, “Multi-robot coverage: A bee pheromone signalling approach,” in Proc. Artif. Life Intell. Agents Symp. Cham, Switzerland: Springer, 2014, pp. 124–140.

[29] A. Breitenmoser, M. Schwager, J.-C. Metzger, R. Siegwart, and D. Rus, “Voronoi coverage of non-convex environments with a group of networked robots,” in Proc. IEEE Int. Conf. Robot. Autom., May 2010, pp. 4982–4989.

[30] Y. Kantaros and M. M. Zavlanos, “Global planning for multi-robot communication networks in complex environments,” IEEE Trans. Robot., vol. 32, no. 5, pp. 1045–1061, Oct. 2016.

[31] D. W. Gage, “Command control for many-robot systems,” Naval Command Control Ocean Survell. Center Rdt E Div, San Diego, CA, USA, Tech. Rep. ADA422540, 1992.