Chapter 1

Biologically Inspired Intelligence with Applications on Robot Navigation

Chaomin Luo, Gene En Jan, Zhenzhong Chu and
Xinde Li

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.75692

Abstract

Biologically inspired intelligence technique, an important embranchment of series on computational intelligence, plays a crucial role for robotics. The autonomous robot and vehicle industry has had an immense impact on our economy and society and this trend will continue with biologically inspired neural network techniques. In this chapter, multiple robots cooperate to achieve a common coverage goal efficiently, which can improve the work capacity, share the coverage tasks, and reduce the completion time by a biologically inspired intelligence technique, is addressed. In many real-world applications, the coverage task has to be completed without any prior knowledge of the environment. In this chapter, a neural dynamics approach is proposed for complete area coverage by multiple robots. A bio-inspired neural network is designed to model the dynamic environment and to guide a team of robots for the coverage task. The dynamics of each neuron in the topologically organized neural network is characterized by a shunting neural equation. Each mobile robot treats the other robots as moving obstacles. Each robot path is autonomously generated from the dynamic activity landscape of the neural network and the previous robot position. The proposed model algorithm is computationally simple. The feasibility is validated by four simulation studies.

Keywords: biologically inspired intelligence, real-time motion planning, navigation and mapping

1. Introduction

Biologically inspired intelligence technique, an important embranchment of series on computational intelligence, plays a crucial role for robotics. The autonomous robot and vehicle industry has had an immense impact on our economy and society and this trend will continue with biologically inspired neural network techniques. Biologically inspired intelligence, such as biologically inspired neural networks (BNNs), is about learning from nature, which can be applied to the real world robot and vehicle systems. Recently, the research and development of bio-inspired systems for robotic applications is increasingly expanding worldwide. Biologically inspired algorithms contain emerging subtopics such as bio-inspired neural network algorithms, brain-inspired neural networks, swam intelligence with BNN, ant colony optimization algorithms (ACO) with BNN, bee colony optimization algorithms (BCO), particle swarm optimization with BNN, immune systems with BNN, and biologically inspired evolutionary optimization and algorithms. Additionally, it is decomposed of computational aspects of bio-inspired systems such as machine vision, pattern recognition for robot and vehicle systems, motion control, motion planning, movement control, sensor-motor coordination, and learning in biological systems for robot and vehicle systems.

One of the applications of biologically inspired intelligence techniques on the robot navigation is complete area coverage navigation of autonomous mobile robots. Complete area coverage (CAC) is an essential issue in mobile robots, which requires the robot path to pass through every area in the workspace. Many robotic applications require CAC, e.g., cleaning robots [1–6], vacuum robots [7], painter robots, autonomous underwater covering vehicles, de-mining robots [8], land mine detectors [9], lawn mowers, automated harvesters [10], agricultural crop harvesting equipment [11], and window cleaners [12]. CAC can be completed by a single robot or multiple robots.

Nowadays, cooperative coverage by a multiple robot system is becoming increasingly important. The cooperative area coverage by multiple robots can improve the efficiency and complete the work more quickly than a single robot. These robots may share the coverage tasks and thus reduce the time to complete the coverage task. Additionally, if one of the robots fails, the rest will fulfill the missions, therefore, the coverage by robots is able to improve reliability and robustness. For instance, in de-mining applications, coverage reliability, an important factor, is enhanced by using cooperative multirobots. In some cleaning applications, the workspace (e.g., a stadium) needs to be cleaned in a limited amount of time. Thus, it requires multiple robots to work in a cooperative manner.

Multi-robot coverage has been extensively studied using various models. Depending on whether a map is required for the multirobots, the coverage models may be categorized as off-line and on-line algorithms [13]. Off-line algorithms require a map of workspace for robots (e.g. [14–16]), while on-line algorithms do not need an environmental map (e.g. [17–21]). Previous research on area coverage may be classified into cell-decomposition-based model (e.g. [15, 16, 21–24]), spanning-tree-based approach (e.g. [14, 18, 23–25]), behaviour-based model (e.g. [26–30]), graph-based model (e.g. [20, 31, 32]), depth first search approach (e.g. [18, 19]), Frontier-based model (e.g. [33–37]), and others (e.g. [38–41]).

Many multi-robot coverage algorithms are based on cell decomposition. Cell decomposition methods break continuous space into a finite set of cells. After this decomposition, a connectivity graph is constructed according to the adjacency relationships between the cells. From this connectivity graph, a continuous path can be determined by simply following adjacent free cells from the initial point to the goal point. Oh et al. [6] developed a triangular cell decomposition method for unknown environments for CAC. This method combines triangular cell decomposition, a template-based approach, and a wall following navigation algorithm for CAC. It can only deal with a single robot CAC. Wagner et al. [21] proposed multi-robot coverage algorithms to explore and cover an unknown environment by approximate cell decomposition approach. The group of multiple robots has limited sensors and no explicit communication. Kurabayashi et al. [15, 16] proposed an exact cell decomposition off-line coverage algorithm for multiple cleaning robots using a Voronoi diagram and boustrophedon approach, where a cost function is defined to obtain a near-optimal solution of the collective coverage task. The approach can avoid overlaps of sweeping areas. The algorithm needs to consider in advance the knowledge of the workspace with known obstacles.

Gabriely and Rimon [42] suggested a spanning tree coverage approach with single robot. It divides up the workspace into discrete cells and generates spanning tree of graph induced by the cells. The robot is able to cover every point precisely once and travel an optimal path in a grid-like representation in the workspace that achieves complete area coverage. Hazon and Kaminka [14] developed a complete and robust multi-robot spanning-tree coverage (MSTC) algorithm based on approximate cell decomposition. Afterwards, the coverage efficiency was improved by a multi-robot forest coverage (MFC) algorithm of approximate cell decomposition proposed by Zheng et al. [23]. Hazon et al. [18] successfully extended the spanning tree work with single robot of Gabriely and Rimon [42] and their off-line work [14] into on-line multi-robot coverage and improved the coverage efficiency.

Behaviour-based strategy for a multi-robot system employs relatively little internal variable states to model the environment and makes fewer assumptions about their environment, thus it is more robust. The cooperative coverage by multirobots with the feature of reactive planning model is implemented by designing individual and team behaviours. Jung et al. [29] combined the advantages of spatial and topological map representations of the environment in a behaviour-based framework for cooperative cleaning of multiple robots. The cooperative multi-robot coverage missions can be accomplished by unifying paths in navigation, cooperation, communication and reactive behaviours [43]. The more detailed explanation of the implementation of the architecture for behaviour-based agents for cooperative multiple cleaning robots are given in [30]. Balch and Arkin [26] proposed a behaviour-based multi-robot approach for coverage, in which the robots are developed with various goal-oriented behaviours for navigation. Recently, Fang et al. [27] used a behaviour-based coverage approach for multirobots to efficiently define the region in which an optimal solution can be found in unknown environments. Most recently, an idea of a leader robot and other follower robots for planning path and controlling robots was proposed using the behaviour-based model [44].

The idea of building up a graph of environment is used for multi-robot coverage. One workspace is decomposed into subregions called cells and therefore a graph may be constructed. The underlying idea of graph-based approach is that multiple robots traverse every edge of the graph to achieve the cooperative coverage. Wagner et al. [20, 31] proposed an approximate cellular decomposition approach for multi-robot coverage to decompose the environment. They employ a dirt grid on the floor for communication among robots. The robots communicate with each other by leaving traces. A graph is built up for representation of the workspace to be covered. Each edge is assigned to two "smell labels". If an edge is traversed, it is marked by a fresh trace of odour. Recently, to benefit from the graph-based approach, Williams and Burdick [32] constructed a graph for multi-robot navigation. An improved graph representation of the task is applied for boundary coverage problem and a graph algorithm is developed for the boundary coverage problem.

A robot can obtain updated knowledge for its environmental map if it moves to a frontier since frontiers are this type of areas that are on the boundary between open space and uncovered space. With the movement of single robot or multiple robots to successive frontiers, the robots can obtain sufficient information to build up and update the maps for coverage mission. Yamauchi [36, 37] adopted a Frontier-based coverage approach, which leads each robot to the closest unknown region, represented by frontier between the free and unknown workspace to produce a robust autonomous cooperative coverage strategy. The technique builds a global map of the environment, which is analysed to locate the frontiers around the robot and environments. Recently, Burgard et al. [33] and Ko et al. [35] developed algorithms to compute utilities of frontier cells to cover different areas of the unknown environments.

Some previous work combines several approaches to take an advantage of different benefits. For instance, Rekleitis et al. [19] split a terrain by an exact cell decomposition method and a tree was built with each subregion as a node. A centralized depth first search (DFS) algorithm is employed for robots to traverse the unknown region and thus the entire areas are explored and covered. Hazon et al. [18] recently extended the spanning tree work with single robot [42] and their own off-line work [14] into on-line multi-robot coverage. Their spanning trees are constructed by a DFS-like procedure. The effective, robust, and complete multi-robot coverage is implemented. Gossage et al. [34] combined the advantages, in order to obtain robust cooperation, of local Frontier-based approach and global graph-based representation of unknown environments for the cooperation of multirobot. Most recently, Zavlanos and Pappas [45] combined a distributed multi-destination potential field approach and a dynamic assignment algorithm for coverage motion planning of multiple robots.

In some other methods, multirobots collect the incoming sensor information of every single robot in a team to cooperatively perform coverage in unknown environments. In most cases, cell decomposition method is used to split terrain and ensures complete coverage [46]. Butler [38, 39] proposed a distributed cooperative coverage algorithm for multirobot, which performs independently on each robot in a team with a rectilinear environment. The algorithm employs only intrinsic contact sensing knowledge to determine the boundaries of the environment. Recently, Boonpinon and Sudsang [47] developed a multi-robot mapping and area coverage approach using a centroidal Voronoi diagram where a team of robots exchange limited sensory information by explicit communication. Latimer et al. [40] and Rekleitis et al. [41] proposed a coordinated multi-robot approach for a coverage mission while the workspace is typically broken down into distinct regions by Boustrophedon decomposition and different region is covered by robots with back-and-forth motions. Most recently, Schwager et al. [48] suggested a near optimal sensing configuration for coverage by a group of robots by learning the distribution of the sensory information in environments.

Although there have been many studies on multi-robot coverage and most attempt to improve completeness, very few existing coverage algorithms focus on robustness. The MSTC algorithm proposed by Hazon and Kaminka [14] is robust and complete. The robots with this algorithm would cover cells more than once. Neural network methods have been broadly applied to robot motion planning, control and coverage (e.g. [2], [49, 50]). However, most of them deal with single robot for coverage (e.g. [2, 5, 6]). Some neural network models require learning procedures (e.g. [51, 52], which are computationally expensive and difficult to achieve CAC in real time.

In this chapter, a neural dynamics approach is proposed for multi-robot area coverage applications. Mobile robots have no collisions among themselves and can avoid obstacles and cooperatively work together to improve cleaning productivity effectively. The proposed approach is capable of performing CAC for multirobots, autonomously without any human intervention. Each robot treats the other robots as moving obstacles. The neural activity landscape of each robot is able to guide the robot to follow a reasonable path and to cooperate with other robots. In this paper, the real-time path is generated by employing a neural network algorithm, without either any prior knowledge of the environment or any pre-defined template. No learning procedures are required in the proposed algorithm. The advantage of such CAC strategy using the proposed neural networks is that the robots do not repeat the previous covered locations. The simulation studies demonstrate that the robustness and fault-tolerant can be ensured if one of the robots fails. It is computationally simple and flexible to implement the proposed algorithm on autonomous CAC as no learning procedures and no templates are required. The dynamics of each neuron is characterized by a shunting equation or an additive equation derived from the membrane model for a biological neural system [53]. There are only local lateral connections among neurons. Thus, the computational complexity depends on the neural network size. The varying environment is represented by the dynamic activity landscape of the neural network. Multiple robots share the environmental information, which is collected from the sensors mounted in the workspace, and all the sensors on individual robots.

The effective, complete, and robust cooperate area coverage is achieved by the proposed neural dynamics model. The term "cooperate" is in the sense that multiple robots can work together to achieve a common coverage mission more efficiently and more quickly. "Robust" is in the sense that the multi-robot system does not fatally failed or is not wholly affected by a single robot failure. In this chapter, cleaning robot is used as an example, but the method is applicable for any CAC applications.

The rest of the chapter is organized as follows. In Section 2, the biological inspiration, model algorithm, and stability analysis of this neural dynamics-based approach to real-time collision-free CAC by multirobot are addressed. Several simulation studies aimed to demonstrate the completeness, robustness, and effectiveness of the proposed model for CAC are performed and described in Section 3. Finally, several important properties of the proposed model with CAC are concluded in Section 4.

2. The proposed model

In this section, the originality of the proposed neural network approach to real-time CAC for multiple mobile robots will be briefly introduced. Then, the fundamental concept and model algorithm of the proposed approach will be presented.

2.1. Biological inspiration

In 1952, Hodgkin and Huxley [53] proposed a computational model for a patch of membrane in a biological neural system using electrical circuit elements. In this model, the dynamics of voltage across the membrane, V_m, is described using the state equation technique as

$$\begin{aligned} C_m \frac{dV_m}{dt} = & -\left(E_p + V_m\right)g_p + (E_{Na} - V_m)g_{Na} \\ & -(E_K + V_m)g_K \end{aligned} \tag{1}$$

where C_m is the membrane capacitance, E_K, E_{Na}, and E_p are the Nernst potentials (saturation potentials) for potassium ions, sodium ions, and the passive leak current in the membrane, respectively. Parameters g_K, g_{Na} and g_p represent the conductances of potassium, sodium, and passive channels, respectively. This model provides the foundation of the shunting model and leads to a number of model variations and applications [54].

By setting $C_m = 1$ and substituting $x_i = E_p + V_m, A = g_p, B = E_{Na} + E_p, D = E_k - E_p, S_i^e = g_{Na}$ and $S_i^i = g_K$ in Eq. (1), a shunting equation is obtained

$$\frac{dx_i}{dt} = -Ax_i + (B - x_i)S_i^e(t) - (D + x_i)S_i^i(t) \tag{2}$$

where x_i is the neural activity (membrane potential) of the ith neuron. Parameters A, B, and D are nonnegative constants representing the passive decay rate, the upper and lower bounds of the neural activity, respectively. Variables S_i^e and S_i^i are the excitatory and inhibitory inputs to the neuron. This shunting model was first proposed by Grossberg to understand the real-time adaptive behaviour of individuals to complex and dynamic environmental contingencies and has many applications in visual perception, sensory motor control, and many other areas [54]. Research on biologically inspired robots has currently received attention [24, 55, 56].

2.2. Model algorithm

The fundamental concept of the proposed model is to develop a neural network architecture, whose dynamic neural activity landscape represents the dynamically varying environment. By properly defining the external inputs from the varying environment and internal neural connections, the uncovered areas and obstacles are guaranteed to stay at the peak and the valley of the activity landscape of the neural network, respectively. The uncovered areas globally attract the robot in the whole state space through neural activity propagation, while the obstacles have only local effect in a small region to avoid collisions. The real-time collision-free robot area coverage is accomplished based on the dynamic activity landscape of the neural network, the previous robot position and the other robot positions, to guarantee all locations to be covered and the robots to travel along smooth, continuous paths with less turning.

The proposed topologically organized model is expressed in a 2D Cartesian workspace $\mathcal{W}$ of the cleaning robots. The position of the ith neuron in the state space S of the neural network, denoted by a vector $q_i \in R^2$, uniquely represents a position in $\mathcal{W}$. In the proposed model, the excitatory input results from the uncovered locations and the lateral neural connections, while the inhibitory input results from the obstacles only. Each neuron has local lateral connections to its neighbouring neurons that constitute a subset R$_i$ in $\mathcal{S}$. The subset R$_i$ is called the receptive field of the ith neuron in neurophysiology. The neuron responds only to the stimulus within its receptive field. Thus, the dynamics of the ith neuron in the neural network is characterized by a shunting equation as

$$\frac{dx_i}{dt} = -Ax_i + (B - x_i)\left([I_i]^+ + \sum_{j=1}^{k} w_{ij}[x_j]^+\right) - (D + x_i)[I_i]^- \tag{3}$$

where k is the number of neural connections of the ith neuron to its neighbouring neurons within the receptive field R$_i$. The external input I_i to the ith neuron at Position (m,n) is defined as:

$$I_i(m,n) = \begin{cases} E & \text{if it is uncovered} \\ 0 & \text{if it is covered} \\ -E & \text{if it is obstacle (another robot} \end{cases} \tag{4}$$

where $E \gg B$ is a very large positive constant. The terms $[I_i]^+ + \sum_{j=1}^{n} w_{ij}[x_j]^+$ and $[I_i]^-$ are the excitatory and inhibitory inputs, S_i^e and S_i^i in Eq. (2), respectively. Function $[a]^+$ is a linear-above-threshold function defined as $[a]^+ = \max\{a, 0\}$ and the nonlinear function $[a]^-$ is defined as $[a]^- = \max\{-a, 0\}$. The connection weight w_{ij} from the ith neuron to the jth neuron is given by $w_{ij} = f\left(|q_i - q_j|\right)$, where $|q_i - q_j|$ represents the Euclidean distance between vectors q_i and q_j in the state space, and $f(a)$ is a monotonically decreasing function, such as a function defined as $f(a) = \mu/a$, if $0 \le a < r_0$; $f(a) = 0$, if $a \ge r_0$, where μ and r_0 are positive constants. Therefore, each neuron has only local lateral connections in a small region $[0, r_0]$. It is obvious that the weight w_{ij} is symmetric, i.e., $w_{ij} = w_{ji}$. A schematic diagram of the neural network in 2D with three-layer (r = 1, 2, and 3) neighbouring neurons with regard to the central neuron C(m,n) is shown in **Figure 1**, where r_0 is chosen as $r_0 = 2$ and r is the number of circles enclosing the central neuron C(m,n). The receptive field of the ith neuron is represented by a circle with a radius of r_0. The ith neuron has only eight lateral connections to its neighbouring neurons that are within its receptive field. The 2D Cartesian workspace in the proposed approach is discretized into squares. The diagonal length of each discrete area is equal to the robot coverage radius that is the size of robot effector or footprint [2]. Each position (grid) uses a number to represent its environmental information. The neurons are placed uniformly on the space to represent covered positions, uncovered positions, and obstacles. In this algorithm, it is necessary to have a flag, denoted by $I_i(m,n)$ in Eq. (4), for a neuron at Position (m,n) to indicate its status that is uncovered, covered, obstacle, or moving obstacle (another robot is regarded as a moving obstacle). This flag may be technically obtained by the sensor information from the current map. A topologically organized discrete map is employed to represent the workspace; each grid uses a number (flag) to represent its environmental information

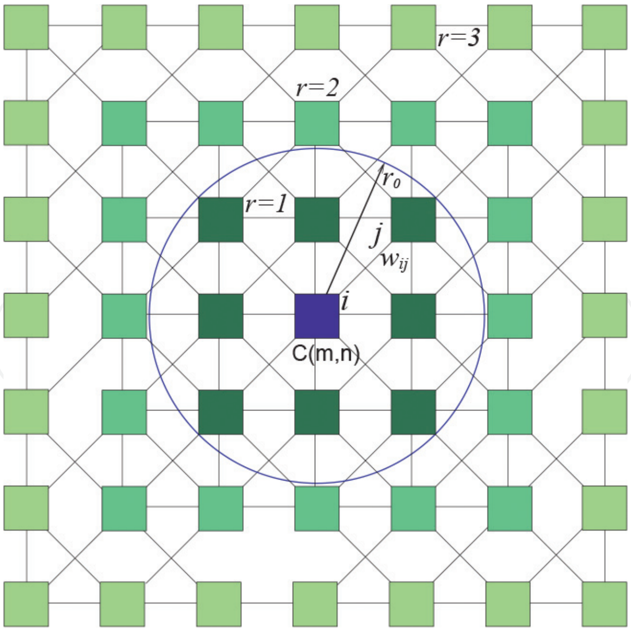

Figure 1. The architecture of a 2D neural network with three-layer (r = 1, 2, and 3) neighbouring neurons with regard to the central neuron C(m,n). The *i*th neuron has only eight lateral connections to its neighbouring neurons that are within its receptive field.

(state). This approach only needs current map, instead of the prior map information. The boundary of the workspace is assumed to be known that can be obtained by wall-following algorithm [57].

The proposed network characterized by Eq. (3) ensures that the positive neural activity can propagate to all the state space, but the negative activity only stays locally. Therefore, the uncovered areas globally attract the robot, while the obstacles only locally prevent the robot from collisions. The positions of the uncovered areas and obstacles may vary over time, e.g., there are moving obstacles (other robots); the covered areas become uncovered again. The activity landscape of the neural network dynamically changes due to the varying external inputs from the uncovered areas and obstacles and the internal activity propagation among neurons. For energy and time efficiency, the robot should travel a shortest path (with least re-visited locations) and make least turning of moving directions. The robot path is generated from the dynamic activity landscape and the previous robot position to avoid least navigation direction changes. For a given current robot position in S (i.e., a position in $\mathcal{W}$), denoted by p_c, the next robot position p_n (also called "command position") is obtained by

$$p_n \quad \Leftarrow \quad x_{p_n} = \max\left\{x_j + cy_j, j = 1, 2, \cdots, k\right\} \tag{5}$$

where c is a positive constant, k is the number of neighbouring neurons of the p_cth neuron, i.e., all the possible next positions of the current position p_c. Variable x_j is the neural activity of the jth neuron, y_j is a monotonically increasing function of the difference between the current to next robot moving directions, which can be defined as a function of the previous position p_p, the current position p_c, and the possible next position p_j, e.g., a function defined as

$$y_j = 1 - \frac{\Delta\theta_j}{\pi} \tag{6}$$

where $\Delta\theta_j \in [0, \pi]$ is the turning angle between the current moving direction and next moving direction, e.g., if the robot moves straight, $\Delta\theta_j = 0$; if goes backward, $\Delta\theta_j = \pi$. Thus, $\Delta\theta_j$ can be given as: $\Delta\theta_j = |\theta_j - \theta_c| = |\text{atan2}\left(y_{p_j} - y_{p_c}, x_{p_j} - x_{p_c}\right) - \text{atan2}\left(y_{p_c} - y_{p_c}, x_{p_p} - x_{p_p}\right)|$. After the current position reaches its next position, the next position becomes a new current position (if the found next position is the same as the current position, the robot stays there without any movement). The current robot position adaptively changes according to the varying environment.

In a multi-robot system, if there exist two robots that work together to sweep in a workspace, it may be viewed as a multiple neural network system [58]. Each robot needs one neural network and all robots share the same workspace information. Each robot treats the other robots as moving obstacles recognized by the sensor information so that they can avoid collisions and cooperatively work together. Each robot has its own neural network which is updated dynamically on the positions of the robot. The environmental knowledge is also dynamically updated, which is sensed by robots via the sensor information. Every position is flagged by a number in Eq. (4). Once one of the multiple robots moves to Position (m, n), the position should be marked as the external input $I_i(m, n) = -E$.

The proposed neural network is a stable system. The neural activity x_i is bounded in the finite interval $[-D, B]$ [54]. The stability and convergence of the present shunting neural network model can also be rigorously proved using a Lyapunov stability theory. By introducing new variables and performing variable substitutions, Eqs. (2) or (3) can be written as

$$\frac{dz_i}{dt} = a_i(z_i)\left(b_i(z_i) - \sum_{j=1}^{N} c_{ij} d_j(z_j)\right) \tag{7}$$

which is Grossberg's general form [54]. It can be proved that Eqs. (2) or (3) satisfies all the three stability conditions required by the Grossberg's general form [54]. The rigorous proof of the stability and convergence of Eq. (7) can be found in [59]. The dynamics of the neural network is guaranteed to converge to an equilibrium state of the system. Eq. (3) combined with the previous robot position ensures to generate complete coverage path. At the beginning, when $t = 1$, the neural activity of all neurons is set to zero. The state of the workspace varies in terms of the dynamics of the neural network described by (3) due to the influence of external inputs. The planned motion ends when the network reaches a steady state.

It is inevitable that multiple cleaning robots have to deal with a deadlock situation in real-world applications. When a cleaning robot arrives in a deadlock situation, i.e., all the neighboring positions are either obstacle or covered locations, all the neural activities of its neighboring locations are not larger than the activity at the current location, because its neighboring locations receive either negative external input (obstacles) or no external input (covered locations), and all the covered neighboring locations passed a longer decay time as they were covered earlier than the current location [see Eq. (3)]. In the proposed model, the neural activity at the deadlock location will quickly decay to zero due to the passive decay term $-Ax_i$ in Eq. (3). Meanwhile, due to the lateral excitatory connections among neurons, the positive neural activity from the uncovered locations in the workspace will propagate toward the current robot location through neural activity propagation (please see [60]). Therefore, the robot is able to find a smooth path from the current deadlock location directly to an uncovered location. The robot continues its cleaning task until all the locations in the workspace become covered. Thus, the proposed model is capable of achieving complete coverage path planning with deadlock avoidance. The multiple robots will not be trapped in deadlock situations.

The complexity is squarely proportional to the degree of discretization. There are only local connections among neurons. If the workspace is an $N \times N$ square and the number of neurons is $M = N \times N$, N^2 neurons are required and there are $8N^2$ neural connections. If the workspace is a rectangle, the number of neurons required is equal to $M = N_x \times N_y$, where N_x and N_y are the discretized size of the Cartesian workspace. Each neuron has maximal eight local connections. As a result, the total neural connections are $8M$. The computational complexity of the proposed algorithm is $O\left(N^2\right)$.

2.3. Implementation issues

There have been many studies on the CAC implementation of mobile robots using various approaches [46, 61–63]. The selection of on-board sensors is equally important as the development of CAC navigation algorithm itself. The performance of multirobots will depend on both CAC navigation algorithm and the placement of on-board sensors. Appropriate amount of sensors is key for the multi-robot system. Use of excessive amount of sensors will cause the increase of cost of robots [62, 64]. Each robot in this multi-robot system has the same configuration. The basic configuration of the robot consists of CPU, memory, sensors, DC motors, wheels, brush or dustpan, and an on-board power supply (e.g., rechargeable battery) [61].

- Sensors equipped on the robots are assumed to be imaging sensors (e.g., camera and position sensor) and rang sensors (e.g., infrared sensors, sonar, ultrasonic sensors or small radar). With on-board sensors, mobile robots can construct current environmental map around the robots [2, 65]. Ultrasonic sensors are employed for range measures due to their simplicity, flexibility, adaptability, low cost, and robustness. The interpretation of the sonar readings is helpful to build an environmental map. The small angular resolution of infrared sensors makes it suitable for CAC with back-and-forth motion. Infrared sensors are capable of reducing angular uncertainty caused by accumulated error of dead reckoning [63]. In addition, several cameras are suspended from the ceiling of workspace that provides global environmental information to robots [66].

- Wheels, DC motors, and discretized environments: each robot is driven by two DC geared motors with two wheels installed to the gear axis. The 2D Cartesian workspace is discretized into squares as most other CAC models. The diagonal length of each discrete grid is equal to a robot sweeping radius, which is the size of the robot effector or footprint (**Figure 2**). The robot in this paper is assumed to be round in shape and a square is embedded in this round as robot's body. A robot sweeping range, which is the size of the

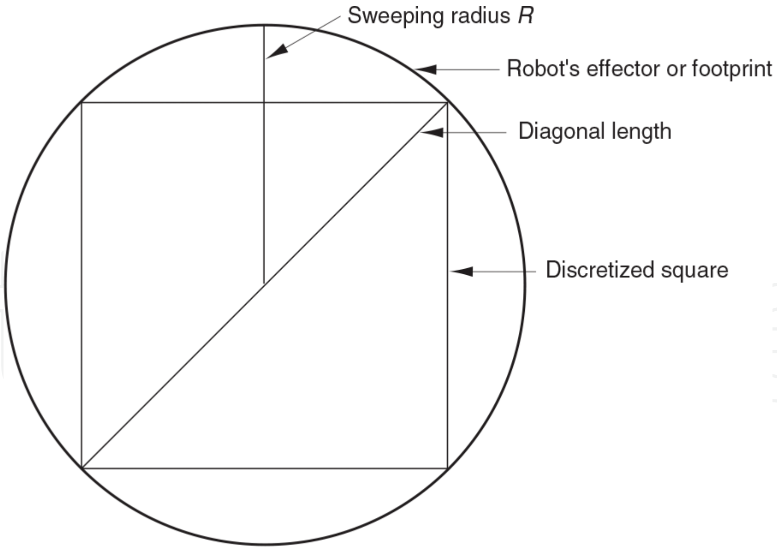

Figure 2. Discretized square enclosed by sweeping area.

robot effector, is proportional to the size of each square. The size of a robot is slightly larger than that of each square. Two wheels driven by DC motors are mounted on shafts with brush or dustpan. The wheels are axle mounted and supported using two sealed ball bearings [see **Figure 3(a)**]. Note that ball bearings do not appear in **Figure 3**. New design is assumed that the wheels have capability to rotate at any directions on the floor make the robots flexibly turn in the workspace. For example, **Figure 3(b)** shows that the robot turns clockwise and counterclockwise 45°. Therefore, sweeping an area can be achieved by traversing the center of that area represented by a rectangular cell. It is assumed that a discrete location represented by that squared cell is regarded to be covered once a robot visits the discrete cell. If a cleaning robot covers every discrete cell, the robot path is considered as a CAC in the workspace.

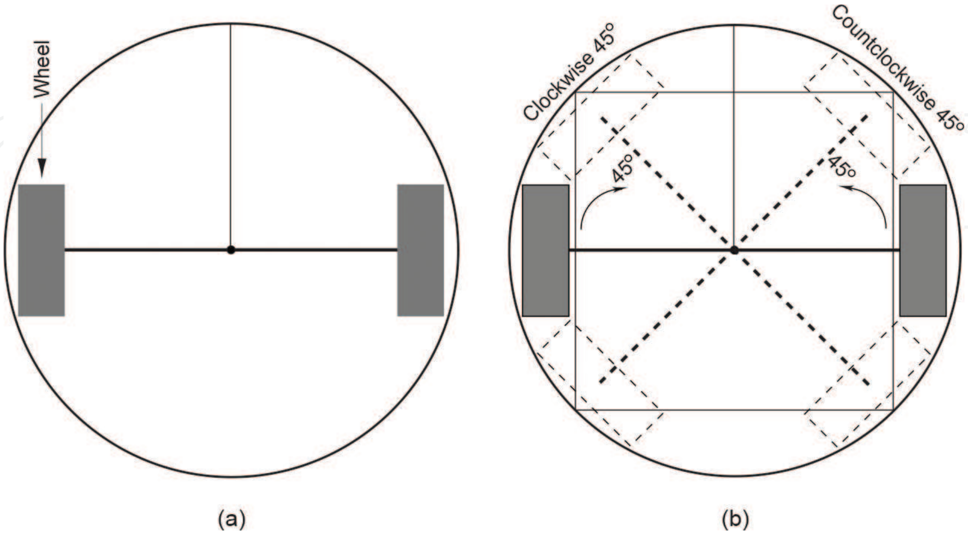

Figure 3. Two wheels driven by DC motors are mounted to the robot. (a) The robot; (b) the robot rotates clockwise and counter-clockwise 45°.

3. Simulation and comparison experiments

The proposed model for CAC is performed by simulation experiments in C++ in this section. The approach is capable of performing CAC for multiple cleaning robots, autonomously without human operation. Two or more mobile robots can cooperatively sweep in indoor environments so as to improve the work productivity. Each robot is required to not only clean its own floor but also cooperate with others to perform the cleaning tasks. In this section, the proposed approach is first applied to multiple robots for cooperative coverage in a corridor-like environment. Then, cooperative coverage in an indoor room environment is studied. Next, it is applied to cooperative coverage in a warehouse environment. Finally, four cleaning robots working together to sweep a sport field is simulated.

3.1. Cooperative coverage in a corridor-like environment

To illustrate the cooperative coverage by a multi-robot system, the proposed model is applied to a cooperative coverage with obstacle avoidance in a corridor-like environment. In the simulation for the multi-robot system, there are two neural network systems for these two robots that share mutual external input signals from the sensor information representing environmental knowledge. Each neural network has 33 × 28 topologically organized neurons with zero initial neural activities. The model parameters are set as $A = 10$, $B = 1$, and $D = 1$ for the shunting equation; $\mu = 0.7$ and $r_0 = 2$ for the lateral connections; and $E = 100$ for the external inputs (on parameter sensitivity, see [2]). In **Figure 4(a)**, one robot, called Robot 1, represented using solid circle starts to sweep from the lower left corner $S_1(1, 1)$; the other, Robot 2, exhibited by a hollow circle covers from the upper right corner $S_2(31, 26)$. After two robots sweep four columns in their own regions, they encounter walls and then enter to clean in a narrow corridor-like workspace [see **Figure 4(a)**]. Two robots move along smooth zigzag

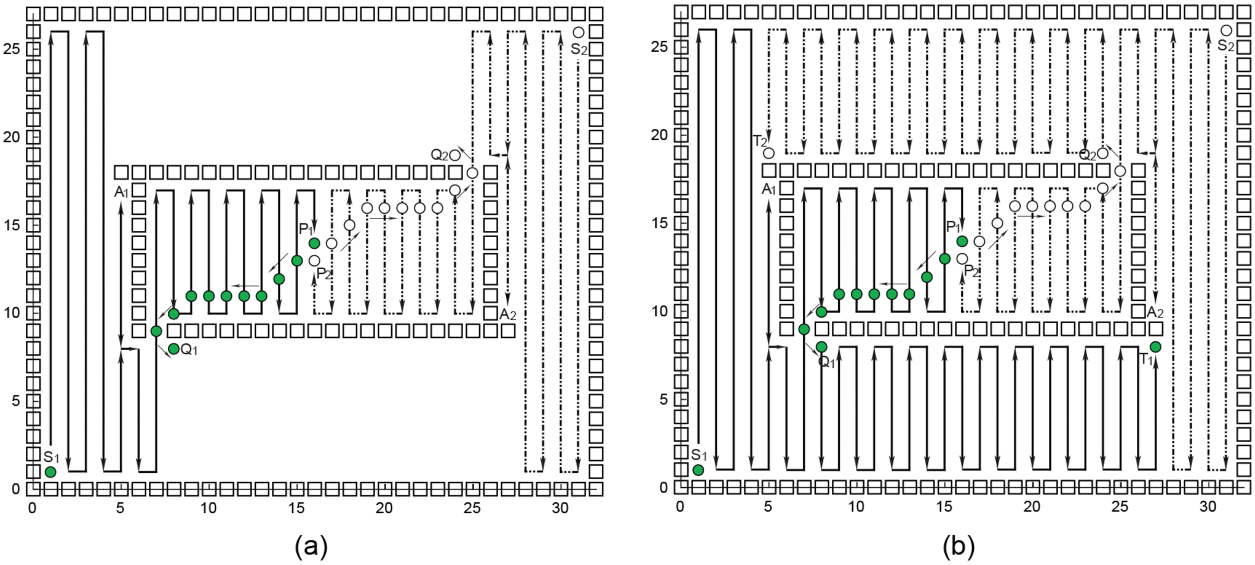

Figure 4. Cooperative coverage in a corridor-like environment. (a) When Robot 1 reaches Position $B_1(8, 8)$ and Robot 2 Position $B_2(24, 19)$; (b) when two robots fulfill the coverage task.

paths by performing back-and-forth motions. Initially, when Robot 1 reaches Position $A_1(5, 17)$ and Robot 2 Position $A_2(27, 10)$, the neural activity landscape of the neural networks is shown in **Figure 5(a)**. In this case, two robots have two neural networks and the neural activities of two neural networks are, respectively, computed through Eq. (3). Consequently, their neural activities may be plotted in a figure as a neural activity landscape of neural network.

These two robots meet at the middle section of the corridor at Positions $P_1(16, 14)$ (flag $I_i(16, 14) = -E$) and $P_2(16, 13)$ (flag $I_i(16, 13) = -E$), respectively. Obviously, Robot 1 and Robot 2 reach the central corridor simultaneously without collision in the central area as shown in **Figure 4(a)**. From the central area, these two robots are able to search point-to-point paths to move to uncovered areas since uncovered areas globally attract them (see [60]). These two robots can follow continuous and smooth paths to achieve $Q_1(8, 8)$ and $Q_2(24, 19)$, respectively. They generate CAC in these two uncovered areas and sweep them as shown in **Figure 4(b)** until they fulfill all the coverage mission at Positions $T_1(27, 8)$ and $T_2(5, 19)$. In fact, the procedure of these two robots cooperatively cover the workspace can be illustrated by the neural activity landscape of the neural networks in **Figure 5**. It demonstrates that after these two robots sweep their own areas, they clean the public aisle areas cooperatively and then move to sweep their own areas again. **Figure 5** shows the neural activity landscape when these two robots move to Positions $A_1(5, 17)$, $A_2(27, 10)$; $B_1(12, 13)$, $B_2(20, 14)$; $C_1(15, 15)$, $C_2(17, 12)$; $D_1(12, 4)$, $D_2(20, 23)$; $E_1(17, 6)$, $E_2(15, 21)$; and $F_1(25, 6)$, $F_2(7, 21)$.

3.2. Cooperative coverage in a warehouse environment

To investigate the flexibility and adaptability of cooperative coverage by multiple robots, the proposed model is applied to a warehouse environment with wall-like obstacles placed in different positions (**Figure 6**). For each robot, the neural network has 32 × 32 topologically organized neurons with zero initial neural activities and the model chooses the same parameters as the case above.

Robots 1 and 2, respectively, work in the lower half and upper half sections of the workspace. However, they can also assist each other, in other words, one robot can aid to cover the other areas if it has already covered its own column. In this simulation, they start from the same side in the workspace. Robot 1 represented by a solid dot starts from the lower left corner S_1(1,1), whereas Robot 2 by an empty circle sweeps from the upper left corner S_2(1,30).

The wall-like obstacles have influence over the cleaning assignments of two robots. The robot paths that these two robots meet at the middle of the workspace are shown in **Figure 6(a)** when they reach Positions A_1(1,15) and A_2(1,16), where these two robots equally accomplish the coverage tasks. The neural activity landscape of the neural networks when Robots 1 and 2 meet at Positions A_1(1,15) and A_2(1,16) can be found in **Figure 7(a)**.

These two robots are responsible for the different areas in the warehouse and sweep their own regions. Due to the placement of obstacles, these two robots get through different amounts of sweeping assignments. For instance, when Robot 1 arrives at Position B_1(7,5), Robot 2 only reaches Position B_2(4,23). The neural activity landscape of the neural networks is illustrated in **Figure 7(b)**. Similarly, when Robot 1 arrives at Position C_1(12,15), Robot 2 only reaches Position

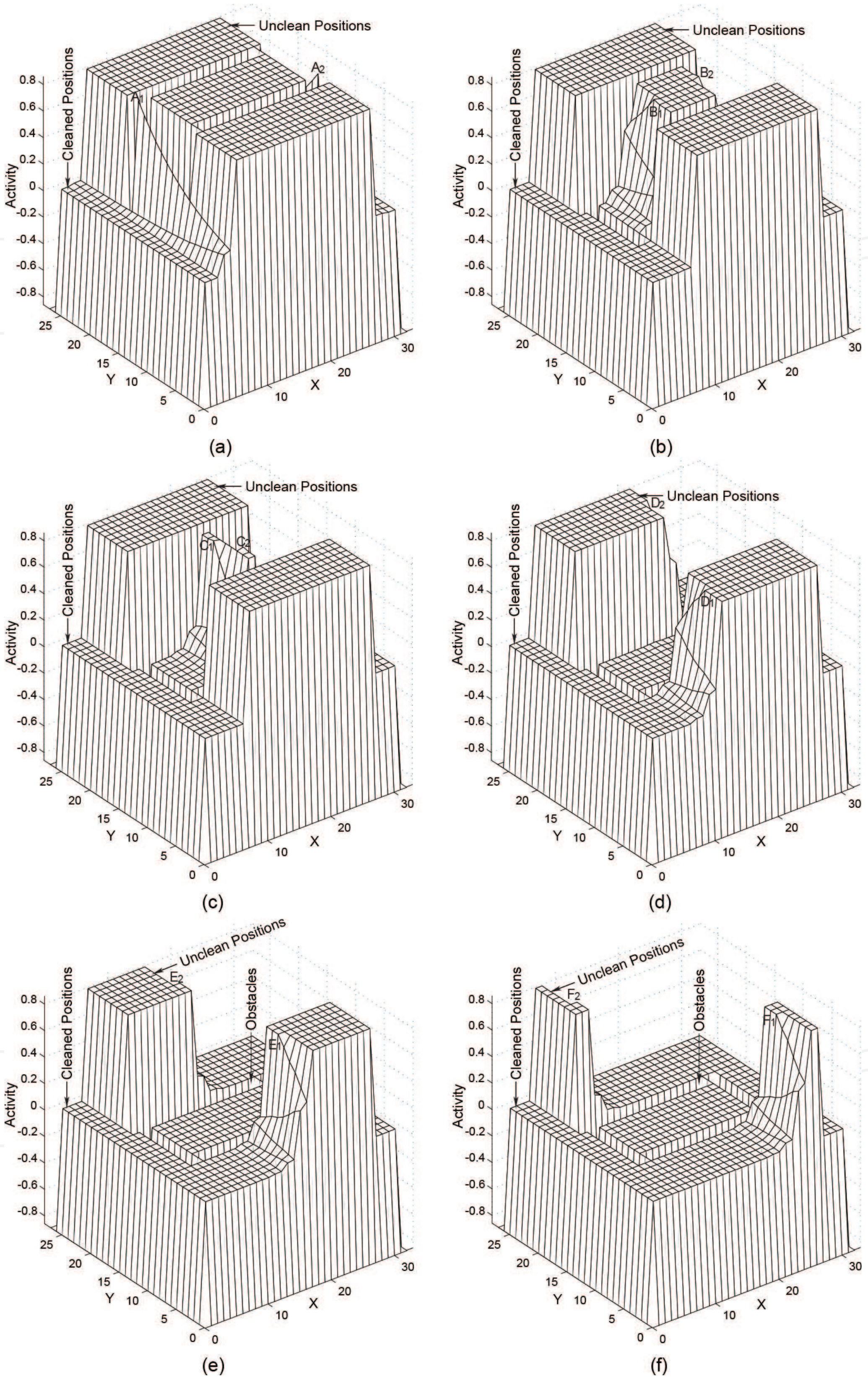

Figure 5. The neural activity landscape of the neural networks for the corridor-like environment case when Robots 1 and 2 reach (a) Positions $A_1(5, 17)$, $A_2(27, 10)$; (b) $B_1(12, 13)$, $B_2(20, 14)$; (c) $C_1(15, 15)$, $C_2(17, 12)$; (d) $D_1(12, 4)$, $D_2(20, 23)$; (e) $E_1(17, 6)$, $E_2(15, 21)$; and (f) $F_1(25, 6)$, $F_2(7, 21)$.

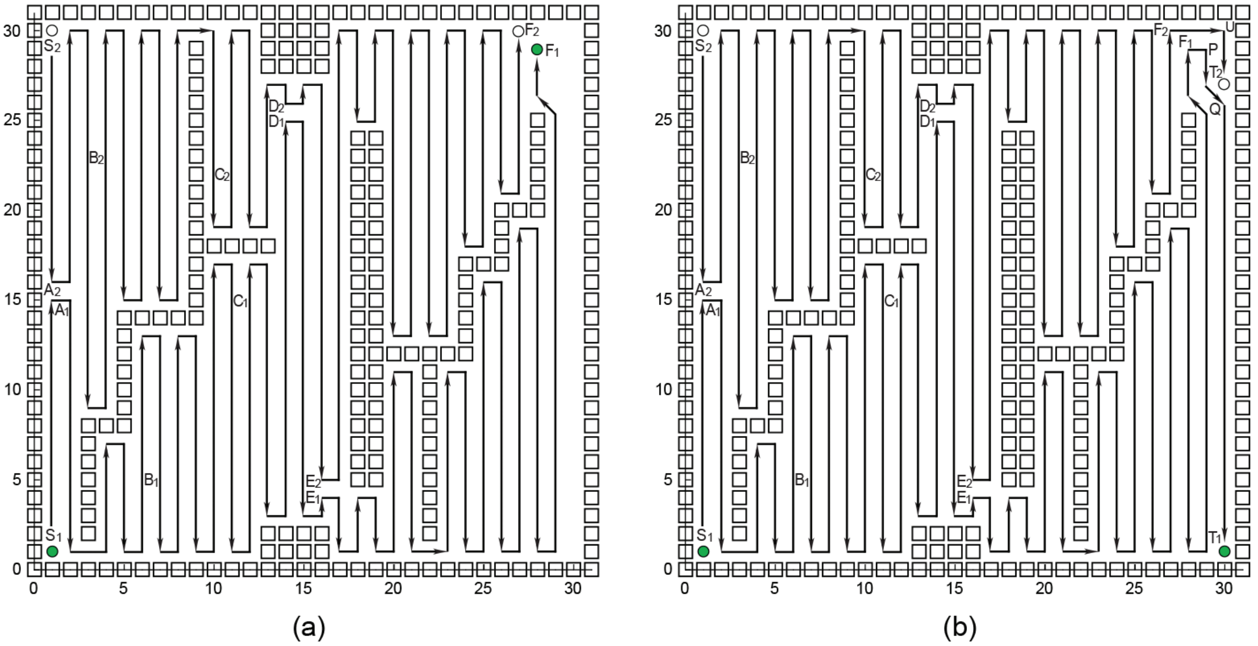

Figure 6. Cooperative coverage in the warehouse environment. (a) Robots 1 and 2 reach Positions $F_1(28,29)$ and $F_2(27,30)$; (b) two robots work cooperatively.

$C_2(10,22)$, in which the corresponding neural activity landscape of the neural networks is exhibited in **Figure 7(c)**. Obviously, when these two robots meet at Positions $D_1(14,25)$ and $D_2(14,26)$, Robot 1 is assisting Robot 2 for coverage work, where the corresponding neural activity landscape of the neural networks is shown in **Figure 7(d)**. Conversely, Robot 2 aids Robot 1 to do coverage work when they arrive at Positions $E_1(16,4)$ and $E_2(16,5)$. They can cooperatively work together without collisions to improve the cleaning productivity efficiently. The neural activity landscape of the neural networks is illustrated in **Figure 7(e)**. Finally, when Robot 1 arrives at $F_1(28,29)$, Robot 2 reaches $F_2(27,30)$, which shows that the Robot 1 has assisted to sweep one column area for Robot 2 as shown in **Figure 6(a)**. These two robots continue to sweep the rest of the workspace. Robot 1 passes Positions $P(29,29)$ and $Q(30,26)$ and reaches the final position $T_1(30,1)$, while Robot 2 attains the final position $T_2(30,27)$ via $U(30,30)$ [see **Figure 6(b)**]. The neural activity landscape of the neural networks is illustrated in **Figure 7(f)**. Ultimately, they reaches Positions $T_1(30,1)$ and $T_2(30,27)$ asynchronously. Therefore, the neural network is able to guide these two robots to complete the coverage task.

3.3. Cooperative coverage by four cleaning robots in a sports field environment

The proposed model is further applied to cooperative coverage in a sports field environment by four cleaning robots, where there exist four neural network systems and the four cleaning robots share mutual external input signals from sensory data representing the environmental information. Each neural network has 20 × 20 topologically organized neurons with zero initial neural activities and the same model parameters as the above case. In **Figure 8(a)**, the use of various lines is to distinguish the generated paths by the robots. Robot 1 whose paths are represented by solid lines starts to move from the lower left corner $S_1(1,1)$. Robot 2 whose

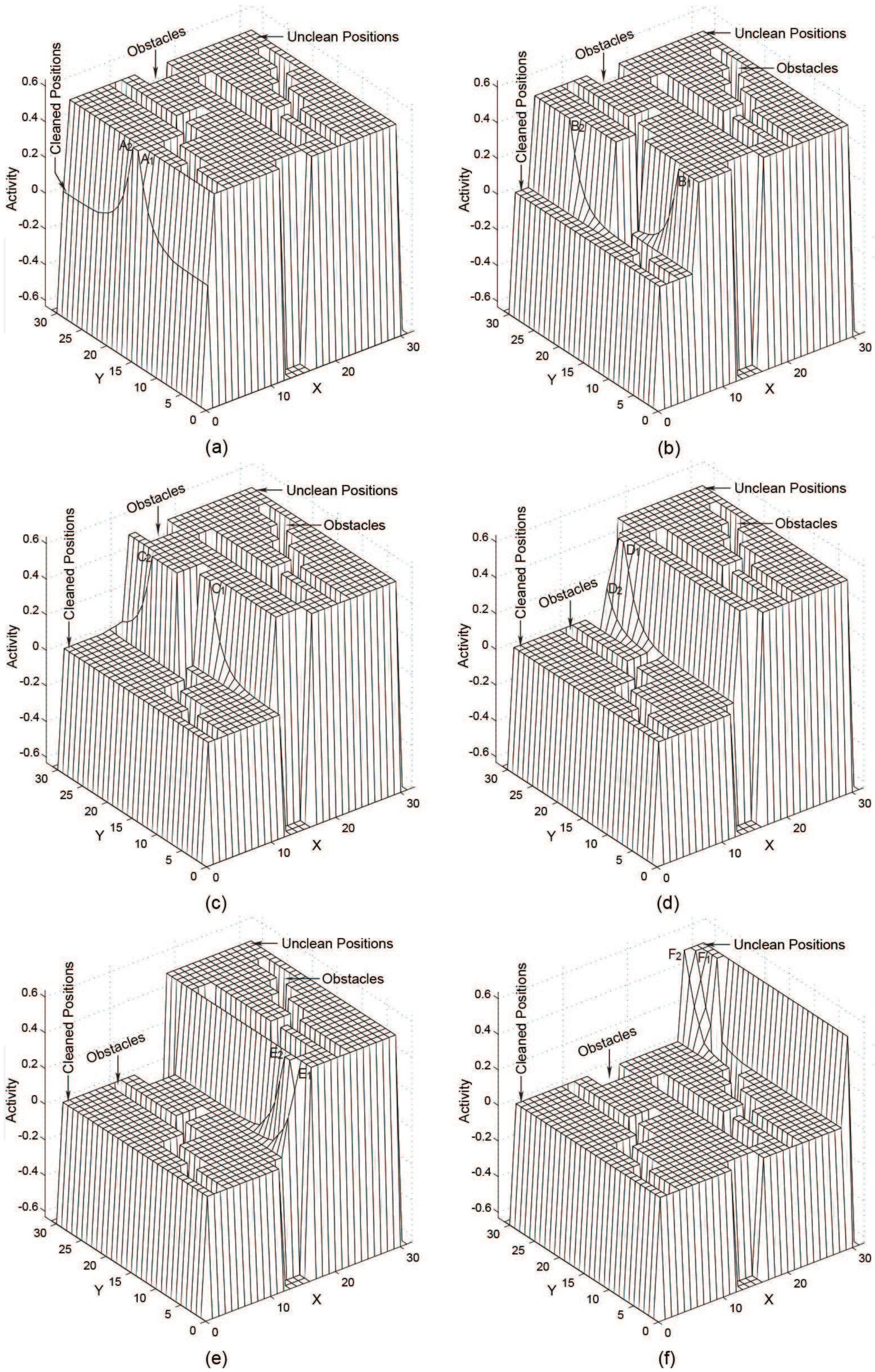

Figure 7. The neural activity landscape of the neural networks for the warehouse environment when Robots 1 and 2 reach (a) Positions $A_1(1,15)$ and $A_2(1,16)$; (b) $B_1(7,5)$ and $B_2(4,23)$; (c) $C_1(12,15)$ and $C_2(10,22)$; (d) $D_1(14,25)$ and $D_2(14,26)$; (e) $E_1(16,4)$ and $E_2(16,5)$; and (f) $F_1(28,29)$ and $F_2(27,30)$.

paths are represented by dashed lines moves from the upper left corner $S_2(1,18)$. Robot 3 whose paths are represented by dash-dotted lines starts from the upper right corner $S_3(18,18)$. Robot 4 whose paths are represented by dash-dot-dot lines sweeps from the lower right corner $S_4(18,1)$. In the simulation, the planned robot paths are shown in **Figure 8(a)**, where the four robots search snake-trail CAC paths and meet at the central area. Because for each neural network, the positive neural activity can propagate to the whole state space of the neural network, each robot can achieve CAC. If one grid is covered by one robot, it will be marked covered by external input signal ($I_i = 0$), another robot will know that it has been covered. When the four robots eventually meet, they have no collision. It shows that these four robots are able to autonomously sweep the whole workspace. They can sweep along zigzag coverage paths and avoid collisions with each other.

After they meet at the central area, where there is a deadlock situation [2], i.e., Robot 1 is at F_1 (9,9); Robot 2 at F_2 (9,10); Robot 3 at F_3 (10,10); and Robot 4 at F_4 (10,9), the robots are able to search point-to-point paths to move to any pre-defined targets. In this simulation as shown in **Figure 8(b)**, Robot 1 goes back to its initial point $G_1(1,1)$. Robot 2, 3, and 4 move back to their initial points $G_2(1,18)$, $G_3(18,18)$, and $G_4(18,1)$, respectively. They may be pre-defined to move to any points based on the demand (see [2]). The targets can globally attract the robots in the whole workspace through neural activity propagation. This case has potential applications in sport fields such as basketball or volleyball match. The four mobile robots can be assigned to clean fields together (e.g., mop sweat on the floor during the volleyball match) and then move back to their starting points during sports contest break, without any human intervention.

3.4. Cooperative coverage in an indoor environment with a robot failure

To verify the robustness and effectiveness of the proposed model, it is applied to a complicated case of cooperative coverage in an indoor environment, where there exist seven sets of

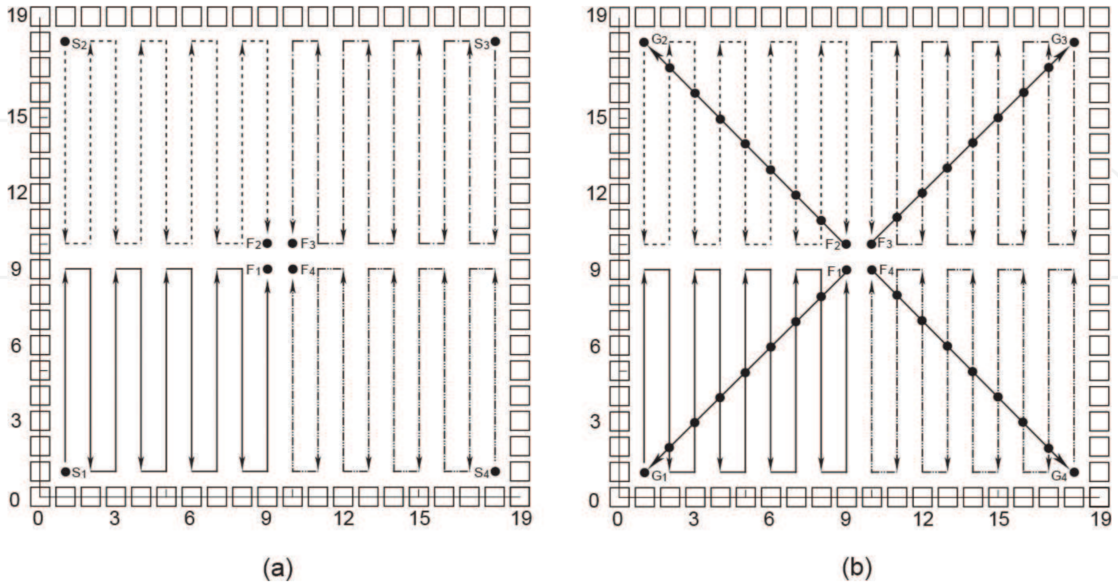

Figure 8. CAC of four mobile robots in a sport field environment. (a) The robot paths when four meet at the centre; (b) the entire robot paths after each robot returns its home position.

obstacles with different sizes and shapes in the workspace as shown in **Figure 9**. Each neural network has 33×28 topologically organized neurons with zero initial neural activities. The model parameters are chosen as $A = 50$, $B = 1$ and $D = 1$ for the shunting equation; $\mu = 0.7$ and $r_0 = 2$ for the lateral connections; and $E = 100$ for the external inputs. In this section, two simulations are performed as shown in **Figure 9**. First, these two robots work cooperatively

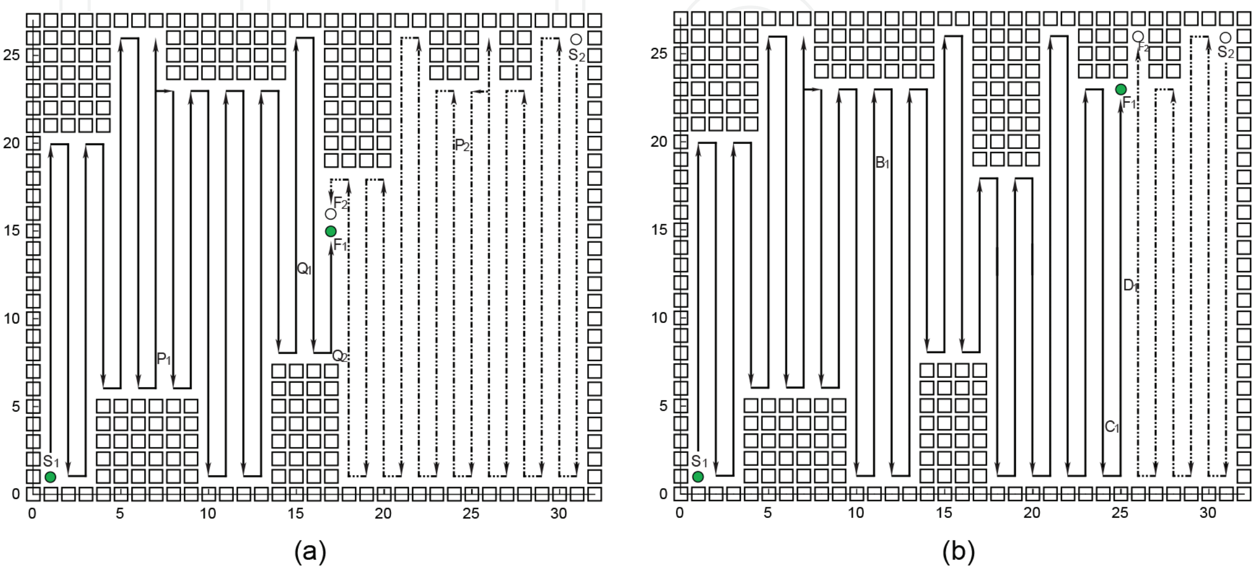

Figure 9. Complete cooperative coverage in an indoor environment with (a) both robots function properly; (b) a robot failure at Position $F_2(26, 26)$.

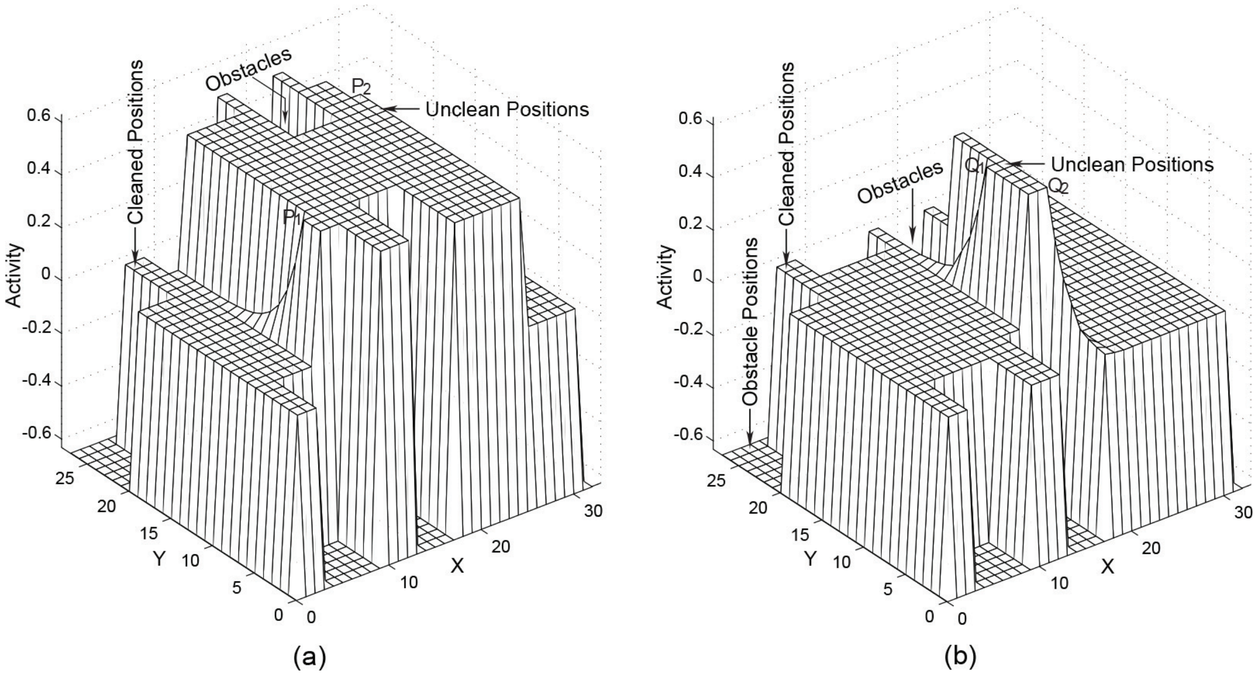

Figure 10. The neural activity landscape of the neural networks for the unstructured environment case when Robot 1 reaches (a) Position $P_1(8, 8)$; (b) Position $Q_1(16, 13)$.

[shown in **Figure 9(a)**]. Second, when Robot 2 fails at one position, Robot 1 is demonstrated to perform the rest of the work.

Robot 1 symbolized by a solid circle starts to move from the lower left corner S_1(1,1) and Robot 2 by an empty circle sweeps from the upper right corner S_2(31,26) as illustrated in **Figure 9**. The real-time collision-free robot paths are shown in **Figure 9(a)**, where the solid lines represent paths of Robot 1 and the dashed lines stand for those of Robot 2. These two robots cover the workspace cooperatively. Robot 1 reaches Position $P_1(8,8)$, while Robot 2 attains Position $P_2(25,20)$ as shown in **Figure 10(a)**. Continuing the work cooperatively, two robots can

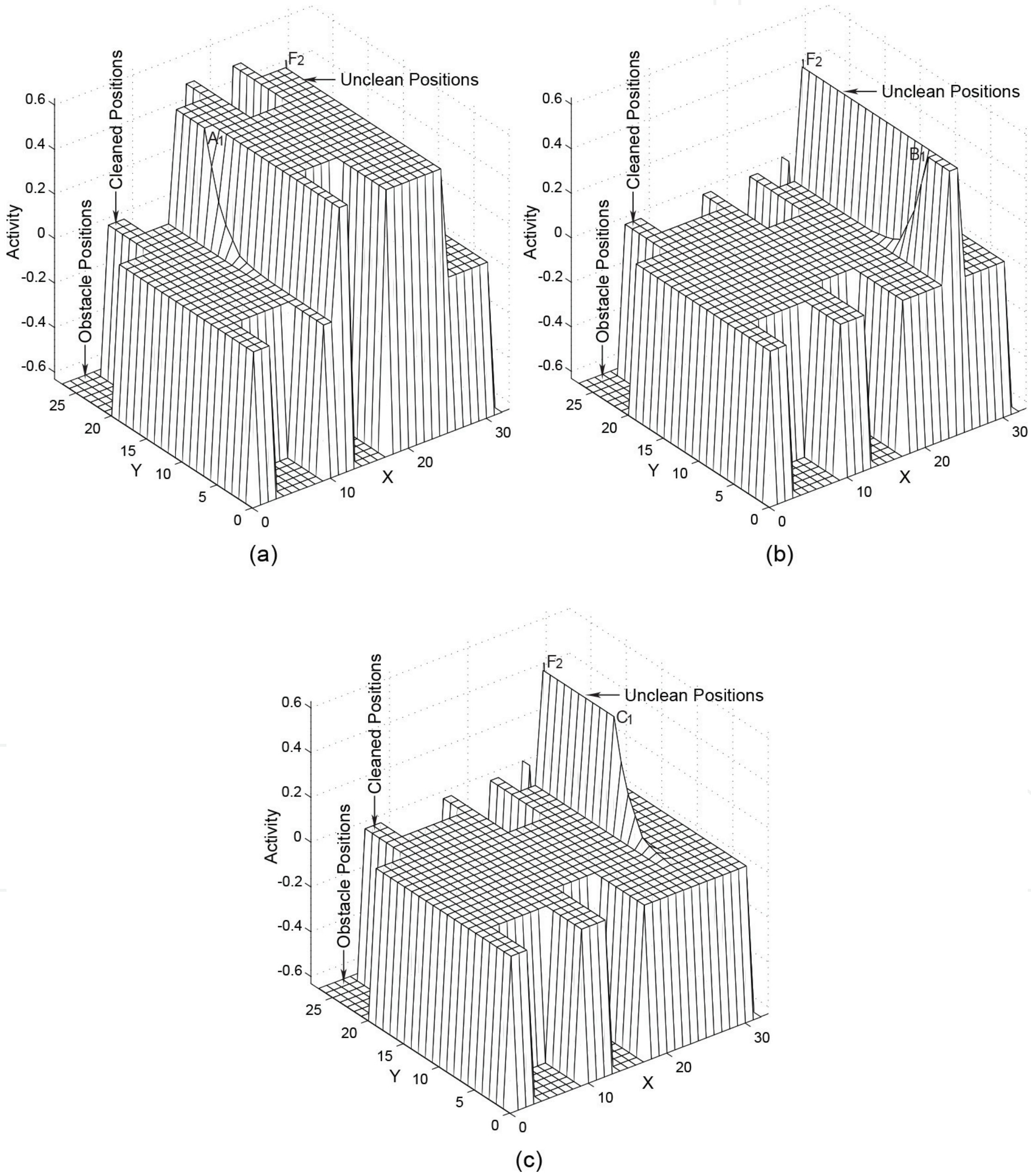

Figure 11. The neural activity landscape of the neural networks for the indoor environment case when Robot 2 fails at $F_2(26,26)$ and Robot 1 reaches (a) Position $A_1(11,19)$; (b) $B_1(24,4)$; (c) $C_1(25,13)$.

approach to achieve complete coverage of the entire workspace as uncovered areas globally attract them to visit. The neural activity landscape of the neural networks illustrates that these robots march on to achieve CAC when Robot 1 arrives at Position $Q_1(16, 13)$ and Robot 2 Position $Q_2(18, 8)$ in **Figure 10(b)**. It shows that these two cleaning robots are capable of autonomously and cooperatively sweeping the whole workspace. Not only can they sweep along a curve path to avoid the irregularly shaped obstacles but also are able to avoid collisions with each other.

Now a simulation is performed to demonstrate the robustness of the proposed model. It is assumed that Robot 2 fails to work at Position $F_2(26, 26)$ as shown in **Figure 9(b)**. Robot 1 can continue to complete the rest of coverage mission even though its partner, Robot 2, was stuck at Position $F_2(26, 26)$ [see **Figure 11(b)**]. The neural activity landscape of the neural networks when Robot 1 reaches Positions $A_1(11, 19)$ and $B_1(24, 4)$ are illustrated in **Figure 11(a)** and **(b)**, respectively. Eventually, when Robot 1 arrives at Position $C_1(25, 13)$ as shown in **Figure 11(c)**, the CAC approaches to an end. In fact, compared **Figure 9(a)** with **Figure 9(b)**, Robot 1 assists Robot 2 for eight columns of sweeping work. The neural activity landscape of the neural networks for the indoor environment case when Robot 2 fails at $F_2(26, 26)$ illustrates the area coverage progress of these two robots (**Figure 11**). Although Robot 2 fails at Position $F_2(26, 26)$, Robot 1 can still be responsible for the rest coverage work. This simulation shows that the proposed approach is robust. The complete coverage can be achieved as long as at least one single robot is able to work.

4. Conclusion

Multiple robots have the capacity for covering the areas more efficiently than a single robot. In this chapter, a biologically motivated neural network approach to cooperative area coverage by a multi-robot system is proposed, which is capable of autonomously accomplishing collision-free cooperative coverage in CAC environments. The effectiveness of the presented paradigm has been discussed and demonstrated through case studies. Multiple robots can work together to achieve a common coverage goal efficiently and robustly.

It is practical to implement the proposed approach in autonomous area coverage as no learning and no templates are required. The robustness and fault-tolerant can be ensured if some robots fail. The model algorithm is computationally simple. The robot path is generated without explicitly searching over the global free workspace or the collision paths, without explicitly optimizing any global cost functions, without any prior knowledge of the dynamic environment, without any templates, and without any learning procedures.

In the future, some research work will be carried out. First, energy-driven multirobot algorithms associated with deep reinforcement learning will be further studied to explore the minimum-energy cleaning robots. Second, the task allocation and impact of number of robots for the cleaning mission will be addressed. Third, the algorithm will be considered to be implemented on an FPGA-based platform. Finally, SLAM and robot vision will be carried out to make the cleaning algorithms more accurate.

Author details

Chaomin Luo[1]*, Gene En Jan[2,4], Zhenzhong Chu[3] and Xinde Li[5]

*Address all correspondence to: luoch@udmercy.edu

1 Department of Electrical and Computer Engineering, University of Detroit Mercy, Michigan, USA

2 Graduate Institute of Animation and Film Art, Tainan National University of the Arts, and Department of Electrical Engineering, National Taipei University, Taiwan

3 College of Information Engineering, Maritime University, Shanghai, China

4 Department of Electrical Engineering, National Taipei University, Taiwan

5 School of Automation, Southeast University, Nanjing, China

References

[1] Lawitzky G. A navigation system for cleaning robots. Autonomous Robots. 2000;**9**:255-260

[2] Luo C, Yang SX. A bioinspired neural network for real-time concurrent map building and complete coverage robot navigation in unknown environments. IEEE Transactions on Neural Networks and Learning Systems. 2008;**19**(7):1279-1298

[3] Ortiz F, Pastor JA, Alvarez B, Iborra A, Ortega N, Rodriguez D, Concsa C. Robots for hull ship cleaning. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2007. pp. 2077-2082

[4] Wagner IA, Altshuler Y, Yanovski V, Bruckstein AM. Cooperative cleaners: A study in ant robotics. The International Journal of Robotics Research. 2008;**27**(1):127-151

[5] Oh JS, Park JB, Choi YH. Complete coverage navigation of clean robot based on triangular cell map, In: Proceedings of IEEE International Symposium on Industrial Electronics; Pusan. South Korea; 2001. pp. 2089-2093

[6] Oh JS, Choi YH, Park JB, Zheng YF. Complete coverage navigation of cleaning robots using triangular-cell-based map. IEEE Transactions on Industrial Electronics. 2004;**51**(3): 718-726

[7] Yasutomi F, Takaoka D, Yamada M, Tsukamoto K. Cleaning robot control. In: Proceedings of the IEEE International Conference on Robotics and Automation; Philadelphia. USA; 1988. pp. 1839-1841

[8] Gage DW. Randomized search strategies with imperfect sensors. In: Proceedings of SPIE, Mobile Robots VIII—The International Society for Optical Engineering; Boston. USA; 1994. pp. 270-279

[9] Najjaran H, Kircanski N. Path planning for a terrain scanner robot. In: Proceedings of the 31st International Symposium on Robotics; Montreal. Canada; 2000. pp. 132-137

[10] Ollis M, Stentz A. Vision-based perception for an automated harvester. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems; Grenoble. France; 1997. pp. 1838-1844

[11] Ollis M, Stentz A. First results in vision-based crop line tracking. In: Proceedings of IEEE International Conference on Robotics and Automation; Minneapolis. USA; 1996. pp. 951-956

[12] Farsi M, Ratcliff K, Johnson PJ, Allen CR, Karam KZ, Pawson R. Robot control system for window cleaning. In: Proceedings of 11th International Symposium on Automation and Robotics in Construction; Brighton. UK; 1994. pp. 617-623

[13] Choset H. Coverage for robotivs - a survey of recent results. Annals of Mathematics and Artificial Intelligence. 2001;**31**:113-126

[14] Hazon N, Kaminka GA. Redundancy, efficiency, and robustness in multi-robot coverage. In: Proceedings of IEEE International Conference on Robotics and Automation; Barcelona. Spain; 2005. pp. 735-741

[15] Kurabayashi D, Ota J, Arai T, Ichikawa S, Koga S, Asama H, Endo I. Cooperative sweeping by multiple mobile robots with relocating portable obstacles. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems; Osaka. Japan; 1996. pp. 1472-1477

[16] Kurabayashi D, Ota J, Arai T, Yoshida E. Cooperative sweeping by multiple mobile robots. In: Proceedings of IEEE International Conference on Robotics and Automation; Minneapolis. USA; 1996. pp. 1744-1749

[17] Ferranti E, Trigoni N, Levene M. Brick & Mortar: An online multi-agent exploration algorithm. In: Proceedings of IEEE International Conference on Robotics and Automation; Roma. Italy, 2007. pp. 761-767

[18] Hazon M, Mieli F, Kaminka GA. Towards robust on-line multi-robot coverage. In: Proceedings of IEEE International Conference on Robotics and Automation; Orlando. USA; 2006. pp. 1710-1715

[19] Rekleitis IM, Dudek D, Milios EE. Multi-robot exploration of an unknown environment, efficiently reducing the odometry error. In: Proceedings of the 15th IEEE International Joint Conference on Artificial Intelligence; Nagoya. Japan; 1997. pp. 1340-1345

[20] Wagner IA, Lindenbaum M, Bruckstein AM. Distributed covering by ant-robots using evaporating traces. IEEE Transactions on Robotics and Automation. 1999;**15**(5):918-933

[21] Wagner IA, Lindenbaum M, Bruckstein AM. MAC versus PC: determinism and randomness as complementary approaches to robotic exploration of continuous unknown domains. The International Journal of Robotics Research. 2000;**19**(1):12-31

[22] Kong CS, New AP, Rekleitis I. Distributed coverage with multi-robot system. In: Proceedings of IEEE International Conference on Robotics and Automation; Orlando, USA. 2006. pp. 2423-2429

[23] Zheng X, Jain S, Koenig S, Kempe D. Multi-robot forest coverage. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems; Edmonton, Canada. 2005. pp. 3852-3857

[24] Zheng X, Koenig S. Robot coverage of terrain with nonuniform traversability. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems; San Diego, USA. 2007. pp. 3757-3764

[25] Agmon N, Hazon N, Kaminka GA. Constructing spanning trees for efficient multi-robot coverage. In: Proceedings of IEEE International Conference on Robotics and Automation; Orlando, USA. 2006. pp. 1698-1703

[26] Balch T, Arkin R. Behavior-based formation control for multirobot teams. IEEE Transactions on Robotics and Automation. 1998;**14**(6):926-939

[27] Fang G, Dissanayake G, Lau H. A behaviour-based optimisation strategy for multi-robot exploration. In: Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics; Singapore. 2004. pp. 875-879

[28] Jung D, Cheng G, Zelinsky A. Experimentals in realising cooperation between autonomous mobile robots. In: Experiments Robotics V—The Fifth International Symposium on Experimental Robotics; Barcelona, Spain. 1997. pp. 609-620

[29] Jung D, Cheng G, Zelinsky A. Robot cleaning: An application of distributed planning and real-time vision. In: Zelinsky A, editor. Filed and Service Robotics. New York, USA: Springer; 1998. pp. 187-194

[30] Jung D, Zelinsky A. An architecture for distributed cooperative-planning in a behaviour-based multi-robot system. Robotics and Autonomous Systems. 1999;**26**:149-174

[31] Wagner IA, Lindenbaum M, Bruckstein AM. Smell as a computational resource—A lesson we can learn from the ant. In: Proceedings of the Fourth Israel Symposium on Theory of Computing and Systems; Jerusalem, Israel. 1996. pp. 219-230

[32] Williams K, Burdick J. Multi-robot boundary coverage with plan revision. In: Proceedings of IEEE International Conference on Robotics and Automation; Orlando, USA. 2006. pp. 1716-1723

[33] Burgard W, Moors M, Stachniss C, Schneider FE. Coordinated multi-robot exploration. IEEE Transactions on Robotics. 2005;**21**(3):376-386

[34] Gossage M, New AP, Cheng CK. Frontier-graph exploration for multi-robot systems in an unknown indoor environment. In: Gini M, Voyles R, editors. Distributed Autonomous Robotic Systems. Japan: Springer; 2006. pp. 51-60

[35] Ko J, Stewart B, Fox D, Konolige K, Limketkai B. A practical, decision-theoretic approach to multi-robot mapping and exploration. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems; Las Vegas, USA: 2003. pp. 3232-3238

[36] B. Yamauchi. Frontier-based exploration using multiple robots. In: Proceedings of the 2nd International Conference on Autonomous Agents; Minneapolis, USA: 1998. pp. 47-53

[37] Yamauchi B. Decentralized coordination for multirobot exploration. Robotics and Autonomous Systems. 1999;**29**(2–3):111-118

[38] Butler Z. Distributed coverage of rectilinear environments [PhD dissertation]. Carnegie Mellon University; 2000

[39] Butler ZJ, Rizzi AA, Hollis RL. Cooperative coverage of rectilinear environments. In: Proceedings of IEEE International Conference on Robotics and Automation; San Francisco, USA. 2000. pp. 2722-2727

[40] Latimer-IV D, Srinivasa S, Lee-Shue V, Sonne S, Choset H, Hurst A. Toward sensor based coverage with robot teams. In: Proceedings of IEEE International Conference on Robotics and Automation; Washington, DC, USA: 2002. pp. 961-967

[41] Rekleitis I, Lee-Shue V, New AP, Choset H. Limited communication, multi-robot team based coverage. In: Proceedings of IEEE International Conference on Robotics and Automation; New Orleans, USA: 2004. pp. 3462-3468

[42] Gabriely Y, Rimon E. Spanning-tree based coverage of continuous areas by a mobile robot. Annals of Mathematics and Artificial Intelligence. 2001;**31**(1–4):77-98

[43] Chen J, Li LR. Path planning protocol for collaborative multi-robot systems. In: Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation; 2005. pp. 721-726

[44] Chueh M, Yeung YLWA, Lei K-PC, Joshi SS. Following controller for autonomous mobile robots using behavioral cues. IEEE Transactions on Industrial Electronics. 2008;**55**(8):3124-3132

[45] Zavlanos MM, Pappas GJ. Dynamic assignment in distributed motion planning with local coordination. IEEE Transactions on Robotics. 2008;**24**(1):232-242

[46] Ribas L, Izquierdo M, Mujal J, Ramon E. A prototype of a low-cost, downsizeable, dynamically reconfigurable unit for robot swarms. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2007. pp. 2167-2172

[47] Boonpinon N, Sudsang A. Constrained coverage for heterogeneous multi-robot team. In: Proceedings of IEEE International Conference on Robotics and Biomimetics; Sanya, China. 2007. pp. 799-804

[48] Schwager M, Slotine JJ, Rus D. Consensus learning for distributed coverage control. In: Proceedings of IEEE International Conference on Robotics and Automation; Pasadena, USA. 2008. pp. 1042-1048

[49] Huang S-J, Lian R-J. A hybrid fuzzy logic and neural network algorithm for robot motion control. IEEE Transactions on Industrial Electronics. 1997;**44**(3):408-417

[50] Su Z, Zeng B, Liu G, Ye F, Xu M. Application of fuzzy neural network in parameter optimization of mobile robot path planning using potential field. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2007. pp. 2125-2128

[51] Arleo A, Smeraldi F, Gerstner W. Cognitive navigation based on nonuniform gabor space sampling, unsupervised growing networks, and reinforcement learning. IEEE Transactions on Neural Networks and Learning Systems. 2004;**15**(3):639-652

[52] Er MJ, Deng C. Obstacle avoidance of a mobile robot using hybrid learning approach. IEEE Transactions on Industrial Electronics. 2005;**52**(3):898-905

[53] Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. Journal of Physiology (London). 1952;**117**:500-544

[54] Grossberg S. Nonlinear neural networks: principles, mechanisms, and architecture. Neural Networks. 1988;**1**:17-61

[55] Svennebring J, Koenig S. Building terrain-covering ant robots. Autonomous Robots. 2004; **16**(3):313-332

[56] Xu WL, Pap J-S, Bronlund J. Design of a biologically inspired parallel robot for foods chewing. IEEE Transactions on Industrial Electronics. 2008;**55**(2):832-841

[57] Domenech JE, Regueiro CV, Gamallo C, Quintia P. Learning wall following behaviour in robotics through reinforcement and image-based states. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2007. pp. 2101-2106

[58] Lee C-Y, Lee J-J. Multiple neuro-adaptive control of robot manipulators using visual cues. IEEE Transactions on Industrial Electronics. 2005;**52**(1):320-326

[59] Grossberg S. Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. IEEE Transactions on Systems, Man, and Cybernetics. 1983;**13**(5):815-926

[60] Yang SX, Luo C. A neural network approach to complete coverage path planning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 2004;**34**(1):718-725

[61] Gutierrez-Osuna R, Janet JA, Luo RC. Modeling of ultrasonic range sensors for localization of autonomous mobile robots. IEEE Transactions on Industrial Electronics. 1998;**45**(4): 654-662

[62] Ulusar UD, Akin HL. Design and implementation of a real time planning system for autonomous robots. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2006. pp. 74-79

[63] Fong T, Thorpe C, Baur C. Multi-robot remote driving with collaborative control. IEEE Transactions on Industrial Electronics. 2003;**50**(4):699-704

[64] Luo RC, Su KL. Multilevel multisensor-based intelligent recharging system for mobile robot. IEEE Transactions on Industrial Electronics. 2008;**55**(1):270-279

[65] Won WJ, Ban SW, Lee M. Real time implementation of a selective attention model for the intelligent robot with autonomous mental development. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2005. pp. 1309-1314

[66] Perez DP, Gomez ES, Quintas MM. Simultaneous localization and structure reconstruction of mobile robots with external cameras. In: Proceedings of IEEE International Symposium on Industrial Electronics; 2005. pp. 1321-1326