



# Multi-robot target entrapment using cooperative hierarchical gene regulatory network<sup>☆</sup>

Meng Wu<sup>a</sup>, Xiaomin Zhu<sup>b,a,\*</sup>, Li Ma<sup>b,a</sup>, Weidong Bao<sup>a</sup>, Zhun Fan<sup>c</sup>, Yaochu Jin<sup>d</sup>

<sup>a</sup> College of Systems Engineer, National University of Defense Technology, Changsha, 410073, Hunan, PR China

<sup>b</sup> Strategic Assessments and Consultation Institute, Academy of Military Sciences, Beijing 100091, PR China

<sup>c</sup> Guangdong Provincial Key Laboratory of Digital Signal and Image Processing, College of Engineering, Shantou University, Shantou 515063, PR China

<sup>d</sup> Department of Computer Science, University of Surrey, Guildford, Surrey, GU2 7XH, UK

## ARTICLE INFO

### Classification codes:

C10

### Keywords:

Gene regulatory network

Target entrapment

Cooperation

Kilobot

## ABSTRACT

For accomplishing a variety of challenging tasks, multi-robot systems perform better than single robots because they have certain properties that a single robot lacks. Target entrapment is one such task; its challenges include finding ways to adapt to different environments to improve entrapment performance. This paper proposes a cooperative hierarchical gene regulatory network (CH-GRN) with the aim of enhancing mutual cooperation between robots neighbours and the utilisation of obstacles to achieve more effective and efficient entrapment. A target-neighbour-obstacle (TNO) pattern generation method is proposed in the upper layer of the CH-GRN design; it integrates the information on targets, neighbours, and obstacles in order to generate more accurate patterns for surrounding the targets. A concentration-vector method is applied in the lower layer of the CH-GRN to enable the robots to adapt quickly to the pattern and thereby complete the entrapment task. At the same time, a proposed obstacle avoidance method is incorporated, which leads to more timely obstacle avoidance. Several simulation experiments are conducted to quantitatively analyse CH-GRN's performance on the target entrapment task in a variety of environments consisting of different types of obstacles. In addition, experiments with Kilobots are conducted to further evaluate CH-GRN's effectiveness. The results show that the proposed model can guide a robot swarm to perform target entrapment tasks in challenging environments with a variety of obstacles, such as various shapes obstacles, narrow channel obstacles, and dynamic obstacles.

## 1. Introduction

Multi-robot systems (MRSs), which consist of a large number of small but autonomous robots [1], are widely applied in the performance of various tasks. The applications of such robot swarms include flocking [2], forming shapes [3], transporting a large object to a goal [4], exploration of unknown environments [5], and autonomous sequencing of several specific tasks [6]. The interactions of the robots with each other and with the environment engender many desirable properties: scalability for different tasks [7,8], adaptability to harsh environments [9], and robustness to partial damage [10–12]. Collectively, swarm robots exhibit abilities that a single robot lacks [13]. Because of their attractive characteristics and their rapid development,

swarm robots have been receiving increasing attention. Many theories about swarm robots have been proposed, which further promote the development of MRSs [14]. Among their applications, the use of swarm robots to entrap targets is an emerging focus of research [15]. MRSs can be more suitable than humans for deployment to achieve target entrapment in dangerous environments [16,17], especially in applications such as those related to anti-terrorism, the marking of hazardous goods in densely populated areas, and isolation of dangerous targets.

With the development of robot technology and increasing user demands, swarm robot systems are applied to complete more complex tasks [18]. An example is target entrapment, a challenging task in which swarm robots are required to surround a target and maintain

**Abbreviations:** C-GRN, cooperative gene regulatory network; CH-GRN, cooperative hierarchical gene regulatory network; GRN, gene regulatory network; H-GRN, hierarchical gene regulatory network; MRS, multi-robot system; TNO, target-neighbour-obstacle

<sup>☆</sup> This work is supported in part by the National Natural Science Foundation of China under Grants 61872378, by the Science and Technology Planning Project of Guangdong Province of China under grant (180917144960530,2019A050519008, 2019A050520001), by the Project of Educational Commission of Guangdong Province of China under grant 2017KZDXM032, and by the State Key Lab of Digital Manufacturing Equipment & Technology under grant DMETKF2019020.

\* Corresponding author.

**E-mail addresses:** [wumeng15@nudt.edu.cn](mailto:wumeng15@nudt.edu.cn) (M. Wu), [xmzhu@nudt.edu.cn](mailto:xmzhu@nudt.edu.cn) (X. Zhu), [mali10@nudt.edu.cn](mailto:mali10@nudt.edu.cn) (L. Ma), [wdbao@nudt.edu.cn](mailto:wdbao@nudt.edu.cn) (W. Bao), [zfanz@stu.edu.cn](mailto:zfanz@stu.edu.cn) (Z. Fan), [yaochu.jin@surrey.ac.uk](mailto:yaochu.jin@surrey.ac.uk) (Y. Jin).

<https://doi.org/10.1016/j.swevo.2023.101310>

Received 3 March 2021; Received in revised form 25 February 2023; Accepted 4 April 2023

Available online 29 April 2023

2210-6502/© 2023 Published by Elsevier B.V.

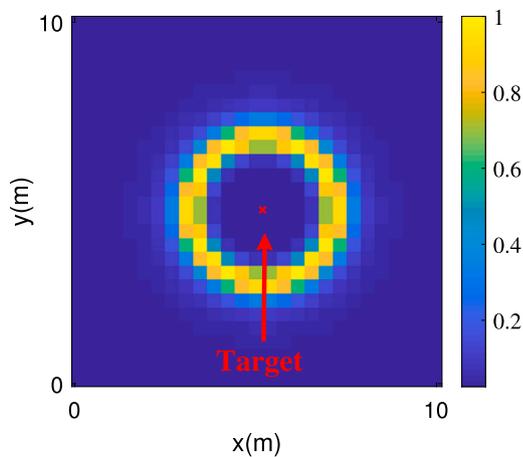


Fig. 1. The pattern of entrapment of a target at  $[x, y]$ .

an appropriate distance from it so that the target cannot escape. The distance is not expected to be too large, because a too large distance will not cause an effective fire threat from the swarm to the target, and the target is easy to escape owing to the difficulty of catching up at a large distance [19]. This distance is also expected not to be too small because some targets are aggressive in reality, and a too small distance will increase the damage possibility of the robot swarm. Therefore, the entrapping robots are expected to be deployed in an annulus with a suitable distance from the target, which is defined as the pattern of entrapment. The pattern of entrapment of a target is shown in Fig. 1. The red “x” represents the target at  $[x, y]$  to be entrapped, and the values around it represent the entrapping requirement. The higher the value, the more effective the entrapment when a robot is in the position. In addition, the appropriate distance between robots is also important, which needs to improve the efficiency of trapping while ensuring communication. The robots in the swarm should pay attention to the cooperation with their neighbours because the task requires many robots rather than a single robot to encircle the target.

Altan et al. [20] designed a single UAV to track the target accurately. However, it should be noted that when the unit used to entrap a target is a swarm, the problem becomes complex. As a key technology, the ability of swarm robots to perform target entrapping tasks directly affects the real-world functions and applications of swarm robot systems. Currently, the target entrapment methods are mainly divided into two categories. One includes improvements to controlling the hardware. The mechanical control of the robot is realised by designing the dynamic equation of the robots’ hardware, for example, to realise the tracking a given path by UAV [21,22]. The other is swarm collaboration, which concerns the swarm robots’ processing of environmental information, to find the best action in their current state. The swarm can be composed of UAVs, unmanned vehicles, or other robots [23], and their movement mode is ignored. The proposed CH-GRN model belongs to the second category. It can be applied to any swarm and has strong practical significance.

For target entrapment, paths are designed for the swarm robots before the task in many studies [24]. However, the swarms in actual engineering application often cannot get sufficient information of the task scene in advance. Inspired by biology, the mechanism in cells has become effective inspire to solve this problem [25,26]. Using the gene regulation network model, the robot can select an action in real-time according to the environmental state, which solves the simple target entrapment problem [27]. However, scenarios of swarm robot tasks trend towards increasing diversification and dynamism, as exemplified by the variety of task scenarios [28] and the motion of task targets [29]. For the target entrapment task, the targets can usually have the ability to move. Therefore, it is difficult to form a complete

and reliable enclosure around the target all the time. In addition, there are various types of obstacles in the task environment, including some obstacles with movement ability, which interfere with the completion of the entrapment task. Among current applications of swarm robots, the problem of target entrapment has been widely studied, but many challenges remain to be overcome:

- The exclusive relies on repulsion to complete the enclosure. In the operation of entrapping a target, the swarm cannot calculate the key areas of the encirclement, and the individuals in the swarm rely solely on repulsion; thus, the encirclement formation is completed slowly and with low efficiency [19].
- The obstacle avoidance strategy is complex and independent. When swarm robots are avoiding obstacles in the environment, they cannot execute the entrapping task at the same time [30].
- Adapt passively to your environment. When obstacles appear in the environment, swarm robots typically only avoid them and do not make effective use of them. When obstacles approaching dynamically, swarm robots are damaged because of untimely obstacle avoidance [29].
- Unable to entrap dynamic targets. Due to the slow speed of robot pattern generation and pattern adaptation, when the target is dynamic, it cannot track and surround the dynamic targets in a short time [31].

In this study, an approach is explored to controlling robot behaviour that aimed to achieve an efficient and robust entrapment. The pattern of entrapment is defined as a disk-like shape around the target, composed of a robot swarm, that forms a robust enclosure. The proposed cooperative hierarchical gene regulatory network (CH-GRN) includes a mechanism for cooperation among neighbours and the utilisation of obstacles for target entrapment. Several simulations are designed to investigate the performance of CH-GRN in terms of the entrapping speed and quality. Furthermore, an experiment using Kilobots is conducted to identify the model’s advantages and disadvantages and to ascertain its potential applicability (in the slightly modified form necessary for working with Kilobots). Theoretically, the proposed method inspires many other tasks improvements of swarms by making use of their task environment. In practice, it effectively improves the performance of the swarm in the task of target entrapping and the success rate in besieging dangerous targets and other entrapping tasks.

Our study on the proposed cooperative hierarchical control model makes several contributions to target entrapment research:

- A target–neighbour–obstacle (TNO) pattern generation method, embodied in the upper layer of the proposed network, takes into account interactions with the environment, including the utilisation of obstacles. By adding the use of obstacles in generating an entrapping pattern, the swarm can purposefully move to the blank entrapment areas, resulting in improved efficiency and better performance.
- When adapting to the generated pattern, obstacles are regarded as part of the enclosure, which reduces the enclosure area that must be filled by the swarm and the number of entrapment resources required.
- A concentration-vector method is applied in the lower layer of the network. From the current states, the method directly obtains the optimal direction in which the robots must move leading to faster pattern adaptation.
- A set of experiments is conducted using Kilobots to determine whether the proposed model is suitable for implementation with robots having very limited individual capabilities.

The rest of this paper is structured as follows. Section 2 reviews related work. Section 3 describes the proposed CH-GRN model (consisting of a model of protein concentration diffusion, an upper layer for pattern generation, and a lower layer for robot guidance). Section 4 presents

the results of several sets of experiments and the performance analysis. Finally, conclusions are summarised and possibilities for future work are discussed in Section 5.

## 2. Related work

Techniques for controlling multi-robot behaviour are inspired primarily by group behaviour in animals. A second source of inspiration is the phenomenon of clustering at the microcellular level. Other sources include the spatial distribution of bacterial colonies [32] and patterns generated in the fur of certain animals [33]. If a robot is regarded as a cell that cooperates with others to form an adaptive pattern, then research on MRSs can also be inspired by the mechanism of biological evolution. Biologists have carried out a substantial amount of research to understand the mechanisms of embryonic development and cell growth [34,35], which provides a theoretical basis for research on swarm robots.

Therefore, the mechanisms of biomode formation, such as morphogen diffusion, chemotaxis, and gene expression and regulation, are studied to help develop techniques for guiding the MRS. In biology, the concentration of a morphogen (signal molecules diffused in cells) decays with the distance from its source, forming a gradient. To address the problem of oversensitivity to the disturbance caused by the exponential expansion of protein concentration, Werfel [36] provided a solution in which another small diffusion source is placed nearby, resulting in a linear gradient diffusion of protein concentration. Slavkov et al. [37] used the diffusion of two morphogens to enable robots to know their roles within the swarm and simulated the pigmentation on animal furs. Chemotaxis is a mechanism that guides cell movement by a process in which some cells release chemicals to the local environment while other cells react by approaching or avoiding the chemicals [38]. Eyiurekli et al. [39] used chemotaxis in a simulation to separate a mixture of two types of agents and cause one type of agent to surround the other. A problem with this method is that the field function cannot always generate the patterns desired. Gene regulatory networks (GRNs) come into play because all of the cells in a multicellular organism share the same genome, but changes in the arrangement of activated and silenced genes distinguish the behaviour and function of each cell and thus regulate gene expression. Just as GRNs control the division and differentiation of groups composed of individual cells to form organisms, they can be constructed to control the behaviour of individual robots in a robot swarm. Applications of the theory of the evolution of organisms have been studied in recent years [40], and this paper investigates the control of the behaviour of individual robots in a swarm by using GRNs.

Wang et al. proposed a robot navigation algorithm by combining a gene regulatory network (GRN) with a finite state machine (FSM) [41]. The robots employ FSM mechanism to make behavioural choices according to their actual environment and current state of the other robots. However, the limited number of states is hard to meet the needs of complex tasks. Guo et al. [42] applied a GRN to the generation of a multi-robot pattern by controlling the  $x$ -direction and  $y$ -direction proteins of the robot, in which the movement of the swarm is more flexible. But it is difficult for a group of physical robots with limited capabilities to access the global coordinate system. To solve this problem, Guo et al. [43] proposed a GRN-based method for multi-robot pattern generation, in which each robot in the swarm finds its direction by selecting a reference robot as the origin and establishing a local coordinate system. Jin et al. [44] introduced a two-layer GRN for multi-robot pattern generation to entrap targets in a dynamic environment. Oh and Jin [19] extended this hierarchical GRN (H-GRN) structure to achieve region coverage rather than boundary coverage. The first layer of the H-GRN generates the entrapment pattern through gene activation and protein concentration regulation. As the input of the lower layer, this pattern regulates the protein concentration, which determines the positions and internal states of each robot. Meng and Guo [45] studied an evolutionary GRN, in which an evolutionary algorithm determines

the coefficient of the GRN by using a frequently repeated regulation pattern called a network pattern. In addition, through proposing a cooperative GRN, the exploration of the problem of target entrapment has been extended to the field of three-dimensional space with GNSS-denied environments [46]. The method greatly expands the application of GRN in three-dimensional space [47]. However, these GRN frameworks neglect the information in the environment that that may play a paramount role in the efficient self-organisation of MRSs. In our previous work [48], cooperation between neighbouring robots is enhanced by the addition of a diffusion source in a cooperation-based GRN (C-GRN), but the obstacles in the environment are not fully utilised. In actual scenarios, neighbours and obstacles can hinder the movement of not only the robots but also the targets. Therefore, the cooperative hierarchical gene regulatory network (CH-GRN) is designed to enable the robot swarm to obtain help from neighbours and take advantage of obstacles, which is not available in H-GRN and C-GRN, thereby performing tasks more efficiently and robustly. CH-GRN also enables the robot swarm to adapt to changes in the environment and the shapes of obstacles.

To date, most studies on entrapment with a group of robots have been theoretical, based on simulations rather than actual robot swarms. Therefore, it is difficult to judge whether the robot control networks proposed thus far can cross the reality gap and be applied to a large group of simple robots. Rubenstein et al. [49] made a breakthrough in the field of swarm robotics by inventing the Kilobot, a minimal, low-cost robot which is applied to the study of morphogenesis to generate desired shapes, such as a starfish shape [50]. A Kilobot is a kind of simple robot; it can move by vibrating two small legs and can communicate and measure distances by infrared reflection. Each robot receives message packets composed of nine bytes broadcast twice per second. However, because of its limited abilities, a Kilobot is not able to self-localise, nor can it obtain the locations of neighbouring robots or obstacles. A Kilobot uses LED light to indicate its current type and state. To date, little work using Kilobots for entrapment has been reported, with a few exceptions [51]. In this study, several experiments are conducted using Kilobots in order to ascertain the applicability of the proposed CH-GRN.

The acronyms and their original phrases are summarised in Table 1.

## 3. Cooperative hierarchical gene regulatory network

Organisms are made up of cells. In the process of biological growth, a single cell can get the protein concentration around itself. Depending on the protein concentration information, cells can judge their position and state to complete their tasks of making the whole organism maintain a stable state. The mechanism is expected to be applied in the distributed multi-robot systems, so that the information can be transmitted between agents through the diffusion of protein concentration. Thus, it is helpful for the swarm to efficiently calculate its proper position in the enclosure, so as to achieve efficient entrapment. In addition, depending on the immune system of organisms, when there is an invasion, cells can locate the invader through the concentration of protein diffused and take countermeasures including tracking, entrapment, and phagocytosis. The mechanism of biological immune systems can be applied to multi-robot systems, which enlightens the method of target entrapment and obstacle avoidance. In an organism, the change in protein concentration depends on the control of gene regulatory networks. Therefore, a gene regulatory network is designed for the swarm to entrap the targets in the paper.

In this section, the two-layer model of the cooperative hierarchical gene regulatory network (CH-GRN) is introduced. The CH-GRN model which is based on the mechanism of cell growth in organisms [52] is shown in Fig. 2. In the biological model, there is a series of proteins in a cell, along with multiple gene segments inside to control its growth. Some factors in the external environment (such as, the protein concentration) will activate certain gene segments, leading to a change

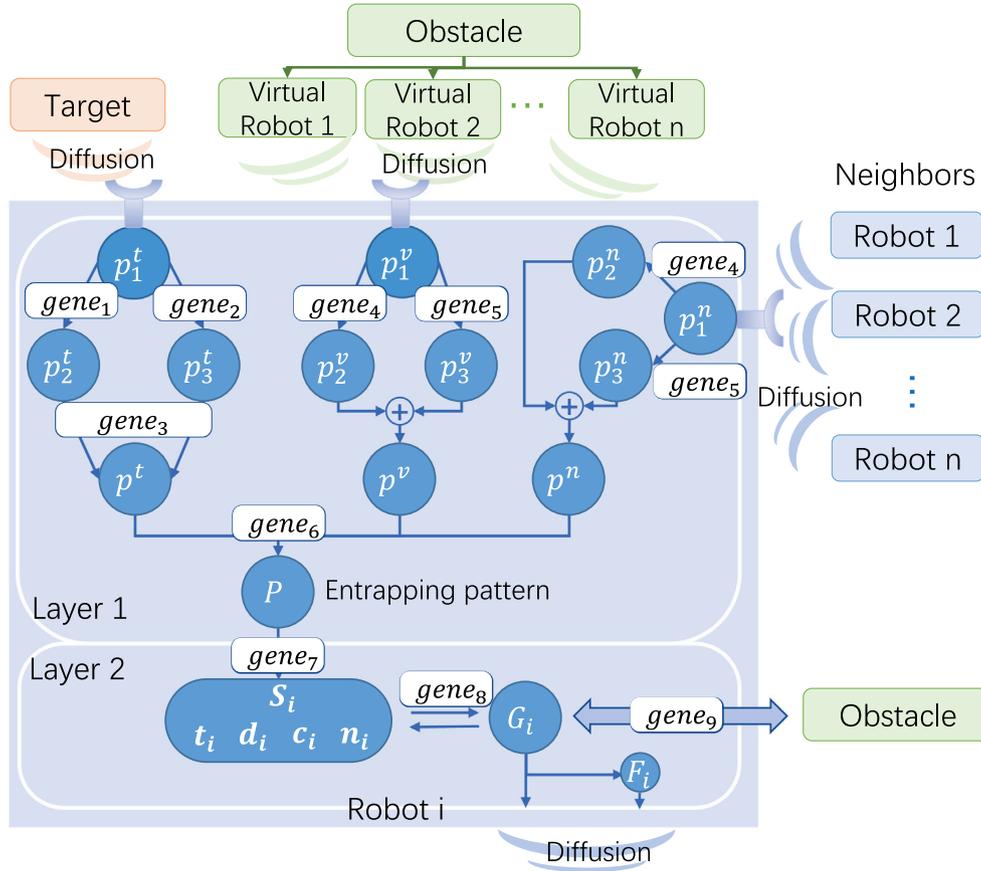


Fig. 2. Cooperative hierarchical gene regulatory network (CH-GRN) model. The internal structure of a robot is represented by a blue rectangle that also represents a cell, in which the dark blue circle represents the protein concentration, the white box is the gene segment, and the ‘Y’ structure at the cell’s edge is a receptor used to detect the protein concentration around the cell.

Table 1  
Acronyms and the original phrases.

Acronyms	The original phrases
CH-GRN	Cooperative hierarchical gene regulatory network
TNO	Target-neighbour-obstacle
MRSs	Multi-robot systems
UAV	Unmanned Aerial Vehicle
GRNs	Gene regulatory networks
C-GRN	Cooperation-based GRN
H-GRN	Hierarchical GRN

in the protein concentration inside the cell and ultimately controlling its growth and development. Correspondingly, there are many ‘gene segments’ in a robot swarm to control changes in its internal ‘protein concentration’ in conjunction with its environment and ultimately control the robots’ entrapment behaviour. A robot swarm is composed of multiple robots with the same structure, and any single robot  $R_i$  in the swarm has nine gene segments (gene 1 to gene 9). The distribution of gene segments is shown in Fig. 2. The internal structure of a robot is represented by a blue rectangle that also represents a cell, in which the dark blue circle represents the protein concentration, the white box is the gene segment, and the ‘Y’ structure at the cell’s edge is a receptor used to detect the protein concentration around the cell. In the following sections, the model’s specifics are given in detail. Firstly, to make the concentration value calculation more accurate and robust, the protein concentration diffusion model in biology is improved in Section 3.1. Then, the details of the upper and lower layers of the proposed gene regulatory network are shown in Sections 3.2 and 3.3, respectively.

### 3.1. Protein concentration diffusion model

CH-GRN is established as an imitation of the structure of a biological cell, and cells communicate with each other through protein concentrations. Analogously, the way for robots to obtain information about the local environment can be seen as the diffusion and perception of protein concentrations in multicellular organisms.

A common phenomenon in biology is that each cell is a source of protein concentration diffusion, and the protein concentration diffuses exponentially, as shown in Fig. 3(a), where the height of the polygon indicates the protein concentration. However, because the low-slope part of the diffusion process is sensitive to noise and interference, the exponential diffusion will affect the accuracy of robot judgement. In our previous work [48], the protein concentration is observed to expand nearly linearly after the addition of a diffusion source. Similarly, by the placement of an auxiliary factor that is allowed to interact with the original protein, the gradient of the protein concentration diffusion will become nearly linear, as shown in Fig. 3(b).

It is assumed that the auxiliary factor  $F_i$  is activated when robot  $R_i$  diffuses the location information via the protein concentration. The concentration of positional protein  $G_i$  traditionally diffuses exponentially, but the auxiliary factor  $F_i$  diffuses non-exponentially. The concentration diffusion modes of  $G_i$  and  $F_i$  are respectively defined by Eqs. (1) and (2), respectively.

$$\frac{dp_{G_i}}{dt} = -p_{G_i} + k_1 e^{-\sigma_i}, \tag{1}$$

$$\frac{dp_{F_i}}{dt} = -p_{F_i} + k_2 [-(1 - \text{sig}(\sigma_i, z_1, \alpha_1)) - \text{sig}(\sigma_i, z_2, \alpha_2)], \tag{2}$$

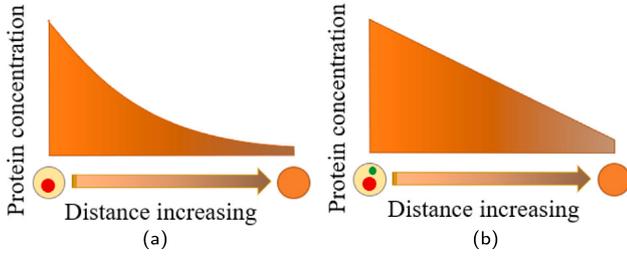


Fig. 3. Comparison of different modes of protein concentration diffusion. The protein concentration decreases as the distance from the diffusion source increases. (a) Exponential concentration gradient. (b) Linear concentration gradient.

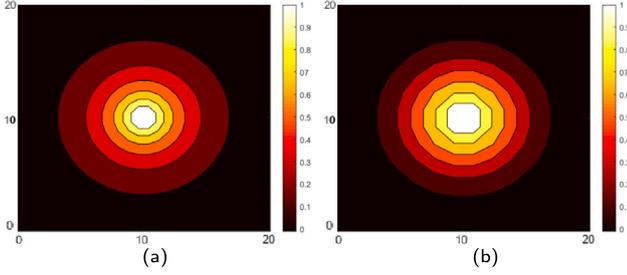


Fig. 4. Two modes of protein concentration diffusion with a robot in the centre position (10,10). (a) Exponential concentration gradient. (b) Linear concentration gradient.

where,

$$\text{sig}(\sigma, z, \alpha) = \frac{1}{1 + e^{-\alpha(\sigma-z)}} \quad (3)$$

and  $p_F$  and  $p_G$  are the concentrations of proteins  $F_i$  and  $G_i$ , respectively, at the location that is a distance of  $\sigma_i$  from the diffusion source. The ratio of the concentrations of proteins  $F_i$  and  $G_i$  is determined by  $k_1$  and  $k_2$ .  $\alpha$  and  $z$  are positive constants which determine the slope and deviation, respectively, of the sigmoid function, which is represented by 'sig' in Eq. (3). As for robot  $R_i$ , the diffusion mode of its protein concentration  $p_i$  after the protein concentration  $G_i$  and auxiliary factor  $F_i$  interact can be expressed as follows:

$$\frac{dp_i}{dt} = -p_i + p_{G_i} + p_{F_i} \quad (4)$$

Through the combination of the two proteins, the protein concentration becomes more robust to environmental changes. In Fig. 4, the concentration diffusions of a robot from the central position (10,10) in two ways are shown. It can be clearly seen that the protein concentration follows almost linear diffusion when the auxiliary factor  $F_i$  is added. The protein concentration diffusion with the robot in the centre changes from exponential diffusion to approximate linear diffusion, which is highly similar to the curve in Fig. 3. The influence of noise and interference on the protein concentration can thus be reduced.

### 3.2. CH-GRN upper layer: the TNO pattern generation method

In the proposed model, each robot in the swarm obtains the relative positions of the targets, its neighbours, and the obstacle boundary through position sensors. In the upper layer of CH-GRN, the entrapment pattern is generated to guide the robots' movements. For improved accuracy over that of the previous generated pattern, a new pattern generation method is proposed, named target-neighbour-obstacle (TNO), which integrates the information on targets, neighbours, and obstacles.

#### 3.2.1. Target

The receptors receive the protein concentration information representing the targets' locations and input it into CH-GRN, where it is represented as the protein concentration  $p'_1$ , calculated as the sum of protein concentrations diffused by targets within the range of detection:

$$\frac{dp_j}{dt} = -p_j + \nabla^2 p_j + \sigma_j^t, \quad (5)$$

$$p'_1 = \sum_{j=1}^{n_r} p_j. \quad (6)$$

In these equations, the target information (obtained by a robot through its sensor or by local simple communication with other robots) is denoted as  $\sigma_j$ , which is a positive constant value when the  $j$ th target exist in the environment.  $p_j$  represents the protein concentration generated by the  $j$ th target, and  $p$  is the sum of the protein concentrations for all of the detected targets.  $\nabla$  is the gradient operator. This integrated protein concentration  $p$  activates the internal genes  $g_1$ - $g_3$  of robot  $R_i$  resulting in changes in protein concentrations  $p'_2$ ,  $p'_3$ , and  $p'$ , where  $p'$  defines the entrapment pattern determined by the information from the targets.  $g_1$ - $g_3$  are calculated as Eq. (7)-(9):  $g_1$  :

$$\frac{dp'_2}{dt} = -p'_2 + \text{sig}(p'_1, z_{t1}, \alpha_{t1}). \quad (7)$$

$g_2$  :

$$\frac{dp'_3}{dt} = -p'_3 + [1 - \text{sig}(p'_1, z_{t2}, \alpha_{t2})]. \quad (8)$$

$g_3$  :

$$\frac{dp'}{dt} = -p' + \text{sig}(p'_2 + p'_3, z_{t3}, \alpha_{t3}). \quad (9)$$

Note that the protein concentration  $p'$  is regulated by both  $p'_2$  and  $p'_3$ , whereas  $p'_2$  and  $p'_3$  are regulated only by  $p'_1$  within the specified concentration range. According to the sigmoid function and the specified thresholds  $z_{t1}$ ,  $z_{t2}$ , and  $z_{t3}$ , the encirclements will be generated around the targets.

#### 3.2.2. Neighbour

The receptor of a robot can receive information on the protein concentration diffused by neighbours. Similarly,  $p^n$  is the sum of the protein concentrations diffused by all neighbours within the range of detection:

$$\frac{dp_j}{dt} = -p_j + \nabla^2 p_j + \sigma_j^n, \quad (10)$$

$$p'_1 = \sum_{j=1}^{n_r} p_j, \quad (11)$$

where  $n_r$  is the number of neighbouring robots within the detection range.  $p'_1$ ,  $p'_2$ , and  $p'_3$  are temporary variables used to calculate protein concentration  $p^n$ . By activating  $g_4$  and  $g_5$ , the protein concentrations  $p'_2$  and  $p'_3$  will be changed. Both of them change the protein concentration  $p^n$ , affected by the locations of the neighbours through simple superposition.  $g_4$  and  $g_5$  are calculated as follows:

$g_4$  :

$$\frac{dp'_2}{dt} = -p'_2 + \text{sig}(p'_1, z_{n1}, \alpha_{n1}). \quad (12)$$

$g_5$  :

$$\frac{dp'_3}{dt} = -p'_3 + [1 - \text{sig}(p'_1, z_{n2}, \alpha_{n2})]. \quad (13)$$

Their superposition affects protein concentration  $p^n$ :

$$\frac{dp^n}{dt} = -p^n + \delta_{n1} p'_2 + \delta_{n2} p'_3. \quad (14)$$

The relationship between the protein concentration  $p^n$  and the distance to the robot's neighbour is shown in Fig. 5. In Fig. 5, the protein

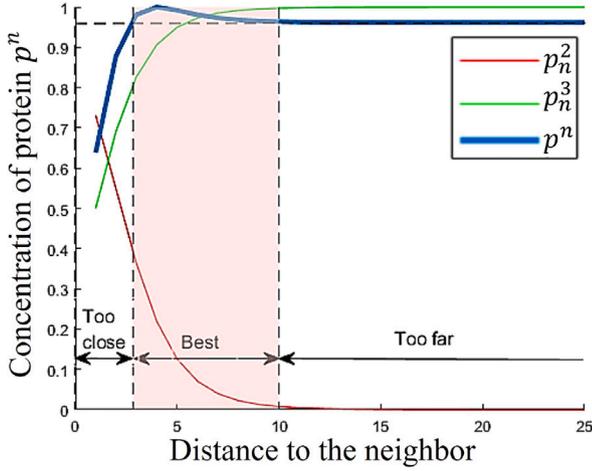


Fig. 5. Relationship between protein concentration  $p^n$  and distance to the robot's neighbour.

concentration  $p^n$  is low near the robot's neighbours. Because the robot can protect its immediate surroundings, the area around the robot does not need reinforcement from other robots. Instead, those robots will be more inclined to move to a position outside the neighbour-protected area to cooperate with their neighbours. Of course, robots are expected to be deployed to a location not too distant from their neighbours; this expectation can be met by adjusting  $z_{n1}$ ,  $z_{n2}$ ,  $\delta_{n1}$ , and  $\delta_{n2}$  are proportional coefficients according to the task and the environment.

### 3.2.3. Obstacle

Leveraging the realisation that obstacles can be another tool for restricting targets when entrapping a target, the proposed CH-GRN model in this study simulates obstacles as a special kind of robot, called virtual robots. With the help of virtual robots, it is possible to achieve firm encirclement of the targets while consuming less resource, for example by using fewer robots.

Because obstacles can have a wide range of shapes and sizes, it is nearly impossible for robots to obtain complete information on the obstacle shapes through sensors alone. It is assumed that the sensor of a robot can only detect the surfaces of obstacles, each of which can be expressed as a one-dimensional curve in two-dimensional coordinates. A circle is used to simulate the collision region of a virtual robot, and the protein diffuses from the centre of the circle. Therefore, a line is drawn in the area with a radius apart from the boundary, and sampling points are taken as the centres of the virtual robots. In this way, an obstacle boundary composed of virtual robots is generated. Fig. 6 shows the arrangements of virtual robots when typical shapes of a detected obstacle surface are given.

A robot obtains the boundary information of obstacles around it and then uses that information as input to generate the pattern of obstacles in its environment:

$$\frac{dp_j}{dt} = -p_j + \nabla^2 p_j + \sigma_j^n, \quad (15)$$

$$p_1^v = \sum_{j=1}^{n_v} p_j, \quad (16)$$

where  $j \in 1, 2, \dots, n_v$  is the sequence number of the virtual robot. By simulating obstacles as a series of virtual robots similar to real robots, obstacles can be used to entrap targets.  $p_1^v$ ,  $p_2^v$ , and  $p_3^v$  are temporary variables used to calculate protein concentration  $p^v$ .  $p^v$  is the sum of the protein concentrations diffused by all virtual robots within the range of detection. Because the virtual robots are similar to real robots in structure,  $g_4$  and  $g_5$  can also affect their concentrations:

$$g_4 : \quad \frac{dp_2^v}{dt} = -p_2^v + \text{sig}(p_1^v, z_{v1}, \alpha_{v1}). \quad (17)$$

$$g_5 : \quad \frac{dp_3^v}{dt} = -p_3^v + [1 - \text{sig}(p_1^v, z_{v2}, \alpha_{v2})]. \quad (18)$$

Their superposition affects protein concentration  $p^v$ :

$$\frac{dp^v}{dt} = -p^v + \delta_{v1}p_2^v + \delta_{v2}p_3^v. \quad (19)$$

Considering the effects of targets, neighbours, and obstacles on pattern generation, the protein concentration  $P$  is obtained after  $g_6$  is activated in the upper layer; it also represents the target entrapment pattern that will be transferred to the lower layer:

$$g_6 : \quad P = p^t + p^n + p^v. \quad (20)$$

Specifically, when a swarm surrounds a target in a scene with obstacles, the target entrapment patterns during and after the target entrapment operation are as shown in the thermodynamic diagram in Fig. 7. The task is to entrap the target next to an obstacle with a swarm of robots. The white dots indicate the swarm, and the white 'x' indicates the entrapped target. The obstacles are shown as green rectangles, and the entrapment pattern generated by the robots is represented by the thermodynamic diagram.

The pattern generation for target entrapment is obviously affected by the existence of obstacles, and the area of incomplete entrapment is represented by a lighter colour. It can be seen that the protein concentration in the area around the obstacle is low. In Fig. 7(a), the entrapment operation is in progress, and the robots are approaching the target near the obstacle. At this time, the high-concentration area is to the right of the target because there are no obstacles or robots there, and the target can easily escape in this direction. In Fig. 7(b), the robots have successfully entrapped the target by taking advantage of the obstacles. It can be seen that because of the restriction imposed by the obstacles on the movement of the target, the number of robots near the obstacles is much less than that on the side away from the obstacles. Therefore, robots will be deployed away from obstacles to strengthen the defence and prevent targets from escaping. The generated pattern can guide the robots to complete the encirclement more quickly.

### 3.3. CH-GRN lower layer: the concentration-vector method

In the proposed CH-GRN model, the upper layer is responsible mainly for generating an adaptive pattern, and the lower layer is responsible for guiding robot behaviour according to the pattern generated by the upper layer. In the lower layer, a concentration-vector method is applied to enable the robot swarm to adapt to the pattern more quickly:

$$g_7 : \quad \frac{dS_i}{dt} = -S_i + r_1 t_i + r_2 d_i + r_3 c_i + r_4 n_i, \quad (21)$$

where  $i \in 1, 2, \dots, N$  is the robot index, and the protein concentration  $S_i$  is the internal position vector of robot  $R_i$ .  $r_1$ ,  $r_2$ ,  $r_3$ , and  $r_4$  are parameters that need to be optimised according to the targets; each represents the weight of the corresponding vector.  $t_i$  is the protein concentration vector of robot  $R_i$ , which depends on its target;  $d_i$  is the sum of the location protein vectors diffused by the neighbours around the robot;  $c_i$  is the core vector of the internal state;  $n_i$  is the sum of direction vectors to avoid collisions between neighbours of robot  $R_i$ .

$t_i$  is a unit vector, which points to the target and guides robot  $R_i$  to approach or avoid the target:

$$t_i = \begin{cases} \frac{G_i - G_{nt}}{|G_i - G_{nt}|}, & |G_i - G_{nt}| \geq R^{\text{target}}, \\ -\frac{G_i - G_{nt}}{|G_i - G_{nt}|}, & |G_i - G_{nt}| < R^{\text{target}}, \end{cases} \quad (22)$$

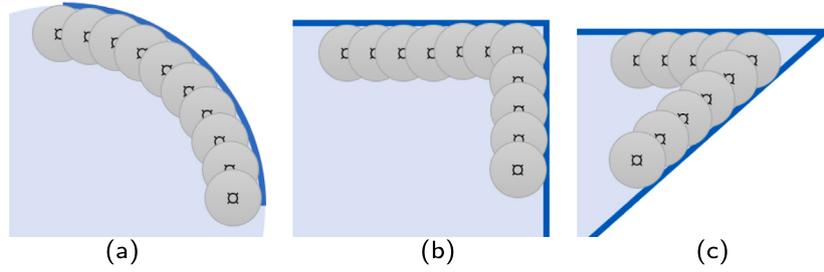


Fig. 6. Arrangements of virtual robots when a detected obstacle surface has a typical shape. (a) Arc. (b) Right angle. (c) Acute angle.

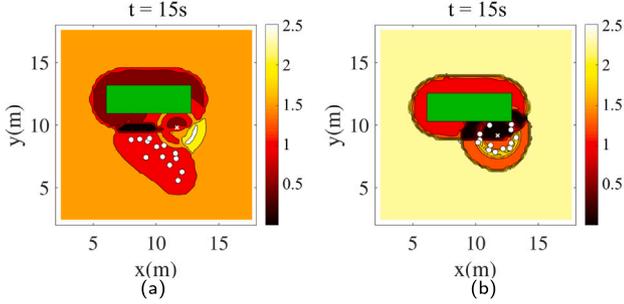


Fig. 7. Target entrapment patterns considering the influence of obstacles. (a) The entrapment task is in progress. (b) The entrapment task is completed.

where  $nt$  is the target of robot  $R_i$ ;  $G_i$  and  $G_{nt}$  are the location proteins of robot  $R_i$  and the task target diffusion, respectively; and  $t_i$  is a unit vector. Generally, if robot  $R_i$  is far from the task targets, then  $t_i$  guides it to move closer to the task targets. However, it is worth noting that if a target is dangerous, the swarm may be damaged if a robot approaches it too closely. As the task of the swarm is to entrap the targets, the robots are required to keep a safe distance  $R^{\text{target}}$  away from any target. When they become too close to a target, they will move in the opposite direction from it.

$d_i$  guides the robot to move to a low-density area of the swarm within its range:

$$d_i = \frac{1}{nb_i} \sum_{j=1}^{nb_i} \frac{G_i - G_j}{|G_i - G_j|}, \quad (23)$$

where  $nb_i$  denotes the number of neighbours of robot  $R_i$ .

Robot  $R_j$  is considered a neighbour of robot  $R_i$  only when the distance between them is shorter than a threshold  $R_i^{\text{probe}}$ .

$c_i$  is determined by the information output by the upper layer of CH-GRN. It guides the robot to the highest concentration point in its domain by activating  $g_8$ , which is calculated as  $g_8$  :

$$c_i = \frac{G^* - G_i}{|G^* - G_i|}, \quad (24)$$

where  $G^*$  is the maximum value of the positional protein in the entrapment pattern. That is,

$$G^* = \operatorname{argmax}(P(G)) \quad G \in \left\{ G \mid |G - G_i| < R_i^{\text{probe}} \right\}, \quad (25)$$

where  $G$  is the collection of protein concentrations at all positions in the domain of robot  $R_i$  that can be detected.

In order to avoid physical collisions between robots, a safe distance  $R^{\text{safety}}$  is set for the robots, whose value is determined by the environment and the size of the entrapping robots. When other robots are less than the safe distance away from robot  $R_i$ , they will immediately move

in the direction away from robot  $R_i$ :

$$n_i = \begin{cases} \frac{1}{n_{si}} \sum_{j=1}^{n_{si}} \frac{G_i - G_j}{|G_i - G_j|}, & n_{si} > 0 \\ \mathbf{0}, & n_{si} = 0 \end{cases}, \quad (26)$$

where  $n_{si}$  is the set of neighbours within the security range of robot  $R_i$ . This process is only activated when the distance from robot  $R_i$  to its nearest neighbour is less than the threshold  $R^{\text{safety}}$ , i.e. when  $n_{si} > 0$ . Otherwise, the vector is set to zero.

The internal state protein  $S_i$  of a robot is the weighted average of the concentrations of these proteins. After the protein concentration and the obstacles interact through the self-organisation mechanism, the location protein  $G_i$  of robot  $R_i$  can be updated by activating  $g_9$  :

$$\frac{dG_i}{dt} = -G_i + r_5 S_i + r_6 z_i, \quad (27)$$

where  $z_i$  is a vector obtained from interacting with obstacles by using an obstacle avoidance mechanism.

In the conventional obstacle avoidance method, robots need information about the boundaries of obstacles in order to avoid them; they then perform some complex processing and calculate the best direction to move in order to avoid the obstacles. However, in a complex and dynamic environment with obstacles, the robots may not be able to obtain the information on the obstacle boundaries in time. Therefore, an obstacle avoidance mechanism is proposed to deal with sudden changes in the obstacle environment and to respond to sudden emergencies and changes in obstacle locations.

The main principle of the obstacle avoidance mechanism is to enable the robot to determine the direction in which it should move next after calculating the robot's current direction of motion based on the existing model. If an obstacle is detected, left and right deflections are used to avoid the obstacle based on the initial direction of motion obtained prior. The next direction of motion is recorded when the direction is detected with no obstacle and with minimal deflection.

In Fig. 8(a), the polygon indicates an obstacle in the environment, the initial direction of motion of the robot is represented as a solid arrow, and the dotted circle around the robot shows the single-step displacement of the robot. It can be seen that according to the predicted direction of the robot's movement, its next position will coincide with that of the obstacle. Therefore, according to the obstacle avoidance mechanism, the robot will test the left and right deflections and choose the smaller one by comparing the two directions. The robot's final direction of motion is shown in Fig. 8(b); in this scene,  $z_i$  is the vector from the initial direction of motion to the final direction of motion of robot  $R_i$ .

#### 4. Experimental results

In order to test the performance of the proposed model, three sets of computer simulations and a set of physical experiments with Kilobots are conducted.

**Table 2**  
Notation used for simulations.

Parameters	Description	Values
$k_1, k_2$	The diffusion strength of $p_G, p_F$	2,4
$\alpha_1, \alpha_2$	The slope of sigmoid function in protein diffusion	20,20
$z_1, z_2$	The bias of sigmoid function in protein diffusion	0,2
$\alpha_{i1}, \alpha_{i2}, \alpha_{i3}$	The slope of sigmoid function in $g_{1-3}$	20,20,20
$z_{i1}, z_{i2}, z_{i3}$	The bias of sigmoid function in $g_{1-3}$	0.25,0.3,1.2
$\alpha_{i4}, \alpha_{i5}$	The slope of sigmoid function in $g_{4-5}$	20,20
$z_{i4}, z_{i5}$	The bias of sigmoid function in $g_{4-5}$	0.95,1
$r_1, r_2, r_3, r_4$	Weight of internal state protein $S_i$	0.4,0.2,0.2,0.2
$R^{\text{target}}$	The safe distance the robot needs to keep from the targets	1
$R^{\text{safety}}$	The safe distance the robot needs to keep from the neighbours	0.15
$R_i^{\text{probe}}$	Detection distance of robot $R_i$	0.5
$r_5, r_6$	Weight of location protein $G_i$	0.5,0.5

#### 4.1. Simulations

##### 4.1.1. Setup and evaluation metrics

In the simulation studies, an agent represents a robot used to entrap targets. The three simulations are used three different types of obstacles in the task scene. In the first simulation, entrapment with single obstacles, a single specific scenario is analysed, in which the swarm size is set to 40 and the agents are required to entrap three targets next to the three single obstacles. In the second simulation, entrapment with special obstacles, two long obstacles forms a narrow channel. In the third simulation, entrapment with dynamic obstacles, the obstacles have the ability to move.

In these simulations, all agents move in a two-dimensional space, and a square with a side of length 20 m is the region under consideration. There are several green polygon obstacles distributed within the region. Agents are represented by blue dots, and they aim to entrap the targets, each represented by a red 'x'. The generated pattern is represented as a closed red shape. The time interval in the simulations is 1 s, and the total duration  $T$  of one simulation run is 50 s. The speed of the agents' motion is 0.25 m/s, and the speed of motion of the targets and obstacles is 0.1 m/s. Other parameter settings are summarised in Table 2.

In addition, two methods are set up to compare with the proposed CH-GRN to demonstrate the advantages of CH-GRN in entrapping performance. It is worth noting that CH-GRN improves pattern generation by the upper layer, pattern adaptation by the lower layer, and the obstacle avoidance mechanism based on H-GRN [44]. The first method for comparison is CH-GRN without the cooperation mechanism. Compared with H-GRN, CH-GRN without the cooperation improves the pattern adaptation and obstacle avoidance mechanism in the lower layer, while the upper structure is the same as H-GRN. The second method for comparison is H-GRN without improvement [44], which is a classic method in this field and can be applied to target entrapment task. In the simulation experiments, the entrapping performances of the three methods are compared.

To evaluate the effectiveness of entrapment by a swarm of agents, two metrics are used: entrapment strength and enclosure occupancy rate.

The first of these, entrapment strength, can be conceptualised for each target as follows. Both the agents and the obstacles can effectively limit the targets' motion. For the agents to entrap the targets, it is necessary for them to maintain an appropriate distance from the targets. Therefore, the area around each target is divided into three levels as shown in Fig. 9. The enclosure around the best entrapment distance to the target is Region I. The two areas that are too close to or too far from the target constitute Region II. Otherwise, when the distance between the agents and the target exceeds a certain threshold, the agents no longer have an entrapment effect on the target; this area is marked as Region III.  $\mu$  is the weight set according to the importance of different

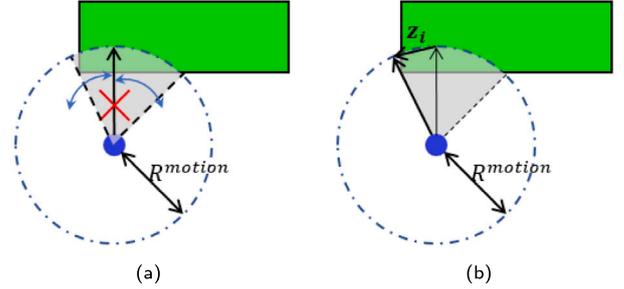


Fig. 8. Obstacle avoidance mechanism for a robot. The polygon indicates an obstacle in the environment, the initial direction of motion of the robot is represented as a solid arrow, and the dotted circle around the robot shows the single-step displacement of the robot. (a) Initial direction of motion. (b) Final direction of motion.

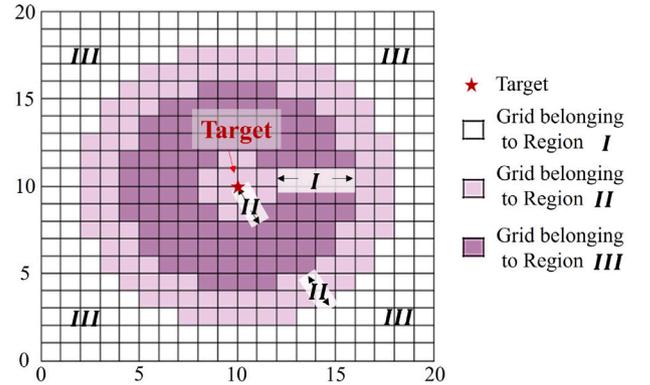


Fig. 9. Division of area around a target.

regions;  $\mu_1$  is the weight of Region 1 and  $\mu_2$  is the weight of Region II, respectively. In the simulation,  $\mu_1 = 2$  and  $\mu_2 = 1$ .

$$\mu = \begin{cases} \mu_1, & p(i, j) \in I \\ \mu_2, & p(i, j) \in II \\ 0, & p(i, j) \in III \end{cases}, (\mu_1 > \mu_2) \quad (28)$$

The calculation of the entrapment strength is based on the grid. According to the location coordinates of the agents and obstacles in the scene, each grid cell has an attribute: obstacle, agent, or blank, which are assigned values of 1.5, 1, and 0, respectively. The obstacle cells are set to a larger value because they are assumed to be more solid than agents.

For each target that needs to be entrapped, the attributes of its surrounding cells and calculate the entrapment strength  $F_s$  can be quantified by using the following equation:

$$F_s = \frac{\sum_i \sum_j \mu * T}{F_{s\max}} \quad (29)$$

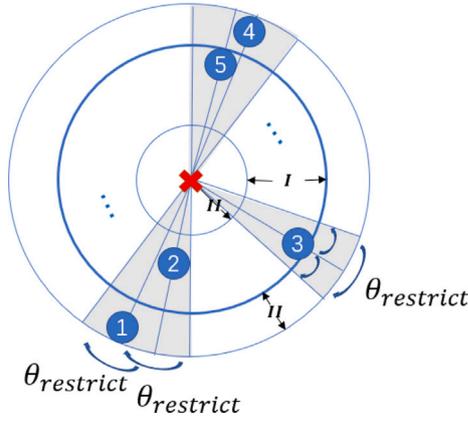


Fig. 10. Area within which the movement of the central target is restricted by five agents.

where  $i$  and  $j$  are the coordinate positions of the grids around the target, respectively. And  $F_{s_{max}}$  is the entrapment strength when  $\mu$  and  $T$  for each grid are the maximum values.

The second evaluation metric is the enclosure occupancy rate; this is used to evaluate the integrity of the target enclosure. Within the threshold of effective entrapment distance, agents will restrict the direction of movement by the targets. This restriction can be measured in angular terms. It is assumed that an agent can restrict the movement of a target within a certain angle determined according to the agent's size and movement capability. As shown in Fig. 10, the area within which the movement of the central target is restricted (marked as shaded) depends on the five agents surrounding it.  $\theta_{restrict}$  is the angle at which the movement of a target is restricted by a single agent. The integrity of the target's enclosure is the ratio of the shaded area to that of the entire enclosure. It should be noted that the overlapping parts of any two agents' restricted areas are counted only once.

Thus, the occupancy rate  $F_o$  of a single target's enclosure can be calculated using the following equation:

$$F_o = \frac{\sum_N \theta_{restrict}}{360} \times 100\%, \quad (30)$$

where  $N$  is the number of agents within the threshold of the effective bounding distance, and  $\theta_{restrict}$  is the angle that the entrapment agents can effectively surround.

#### 4.1.2. Entrapping with single obstacles

In order to assess the effect of the cooperation mechanism in the CH-GRN model on entrapment performance, several simulation experiments are designed with typical scenes. Forty agents are allocated equidistantly on a circle having a diameter of 20 m. Three obstacles of various shapes are placed in the scene, and a target is positioned next to each one. As is often the case in the real world, targets may have some ability to escape from entrapment. Therefore, in the scene, the targets are allowed to move back and forth on the black track lines. The speed of target motion is set to 0.20 m/s, which is slightly less than the speed of agent motion so that the agents would not lose information because of movement by the targets.

In Fig. 11, snapshots of the simulation at times  $t = 1$  s, 30 s, and 50 s are shown, respectively. The results of applying the CH-GRN model, the CH-GRN model without the cooperation mechanism, and the H-GRN model to guide the agents to entrap the targets can be seen on (a)(b)(c) in the first line, (d)(e)(f) in the second line, and (g)(h)(i) in the last line of Fig. 11, respectively.

As shown in the figure, at  $t = 1$  s, the task configurations are identical, including the initial positioning of the agent swarm or the positions of the targets and obstacles. However, with consideration of

the assistance by obstacles in the CH-GRN, the generated patterns differ. Those generated by the model without the cooperation mechanism are a number of circles around the targets because they are determined only by the targets when obstacle information is not considered. This can be seen in Fig. 11(d) and (g), which is the initial state of entrapment under CH-GRN without cooperation and H-GRN. With the full version of CH-GRN, however, the areas covered by obstacles are utilised to restrict target movement; the patterns generated take advantage of the obstacles to encircle the targets. At  $t = 30$  s, the entrapment is in progress. Some details are shown magnified at the top right of Fig. 11(b)(e)(h). Compared with the model without cooperation, the complete CH-GRN model produces a higher degree of completion of target encirclement. It can be seen that the patterns generated by the CH-GRN model evolve with changes in the local environment. As the description in Section 3.2, the patterns consistently highlight the weak part of the enclosure. In contrast, the patterns generated by the model without cooperation mechanism do not change. Compared with (e) and (h), under the control of H-GRN model without the improved of the lower layer, there are significantly fewer robots deployed on the encircles, and the coverage area is also smaller, because the improvement of the lower layer and obstacle avoidance mechanism makes the robots move more accurately and effectively. At  $t = 50$  s, under the H-GRN and the CH-GRN model not making use of obstacle information, the agents are not well deployed to encircle the targets, whereas the entrapment under the complete CH-GRN model avoids this problem. The agents are evenly distributed around the targets, forming stable and solid enclosures with the help of the obstacles. The robots controlled by H-GRN are more likely to drift away due to the loss of target information and are more likely to accumulate in the same area of the encirclement, which is difficult to achieve well-distributed entrapment.

Then, the encirclement effectiveness is evaluated using the proposed two quantitative metrics. The lines in Fig. 12(a) and Fig. 12(b) show the values of the entrapment strength and enclosure occupancy rate for the targets, respectively, over time. There are slight fluctuations in the curves because the agents need to adjust their positions according to the local environment for better entrapment. It can be observed that the three models are able to guide the agents to entrap the targets. In the early stages, there is almost no difference in performance between the three models because they have the same initial positions and distances to the targets. In the later stages, however, the complete version of CH-GRN performs markedly better than the model that does not use the obstacle information, in terms of both entrapment strength and enclosure occupancy rate. Compared with CH-GRN with and without cooperation, H-GRN shows obvious disadvantages. The speed of entrapment under H-GRN is slower, the entrapment strength is lower, and the encirclement is incomplete than under CH-GRN.

#### 4.1.3. Entrapping with special obstacles

Special obstacles are an important part of the task environment of a swarm. As the CH-GRN model is committed to cooperating with obstacles, it is expected to perform better than the model without cooperation in task scenes having special obstacles. To assess this, a scene having a narrow channel composed of two long obstacles is created. The target is initially located in the centre of the narrow channel and then moved vertically downward along a specified trajectory during the simulation, shown by the black dotted line in Fig. 13. At initiation, the swarm agents are randomly deployed in the rear of the target's movement, with the aim of simulating a swarm's pursuit of a target.

An experiment is designed to simulate the entrapment process as the target escapes from the narrow channel. The channel restricts the movement of the swarm and prevents the swarm from reaching the front of the target to complete the entrapment task, but CH-GRN solves this problem well. Fig. 13 shows snapshots of the simulation at  $t = 30$  s, in which the black hollow circles and the black 'x' indicate the initial positions of the swarm agents and target, respectively. At this moment, the ordinate of the target is equal to the bottom of the

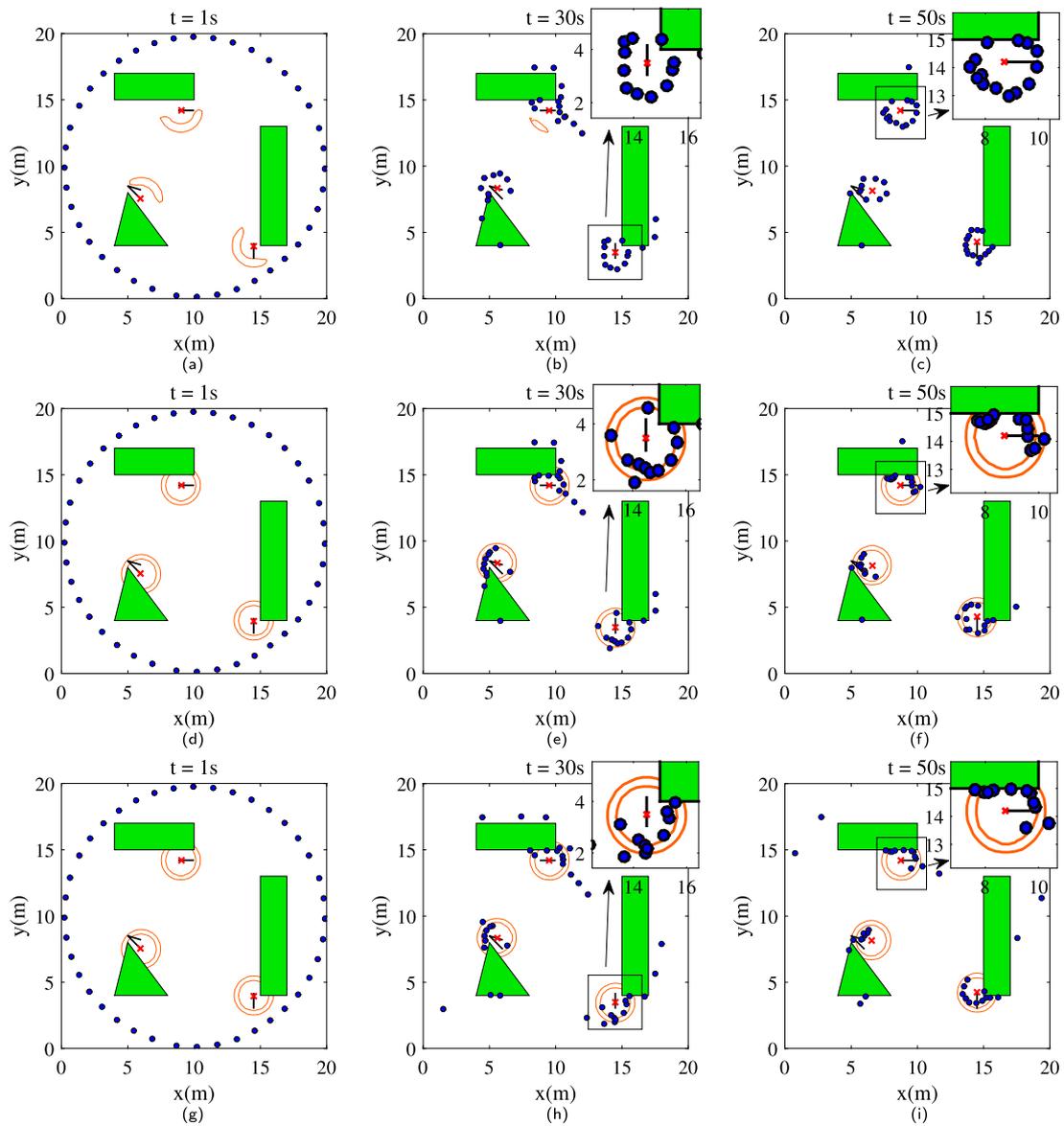


Fig. 11. Snapshots of simulation applying the CH-GRN model with and without the cooperation mechanism and the H-GRN model to guide the agents to entrap the targets. Forty agents in a swarm aim to entrap three targets in the same environment. (a)(b)(c) Entrapment under CH-GRN model at times  $t = 1$  s, 30 s, and 50 s. (d)(e)(f) Entrapment under CH-GRN model without cooperation mechanism at times  $t = 1$  s, 30 s, and 50 s. (g)(h)(i) Entrapment under H-GRN model at times  $t = 1$  s, 30 s, and 50 s.

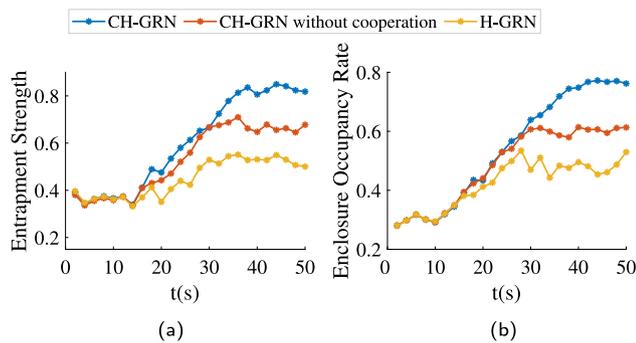
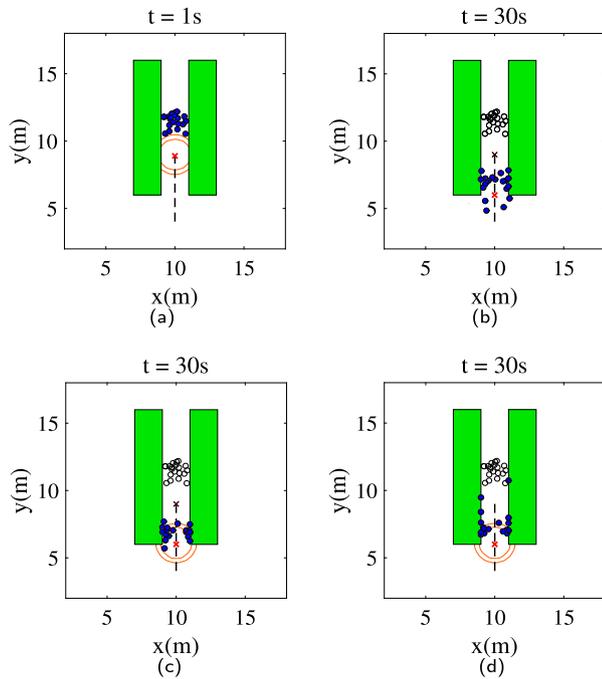


Fig. 12. Effect of cooperation mechanism in CH-GRN when entrapping targets using single obstacles. (a) Entrapment strength. (b) Enclosure occupancy rate.

obstacles, and the target is about to exit the channel. It can be seen that under the control of the model without the cooperation mechanism,

the swarm is concentrated behind the target in spite of its efforts to surround the target. In contrast, under the control of the complete CH-GRN model, the empty space in front of the target is determined as a location needing swarm reinforcement, and so the swarm is more inclined to fortify this empty space to complete the entrapment task. The result is that before the target exits the channel, an almost complete encirclement is formed under the control of the complete CH-GRN model, but not under the version without the cooperation mechanism. The robot under the H-GRN shows the inadaptability of entrapping dynamic targets in the narrow channel. Compared with CH-GRN, robots are hard to catch up with the target and pile up next to the obstacles so that they cannot entrap the dynamic target effectively.

The lines in Fig. 14 show the results of the quantitative evaluation. During the entrapment operation, the target escapes from the inside of the channel to the outside. The initial states under the three methods are the same, and so the main task of the swarm agents in the initial stage is to avoid each other and find suitable positions in the swarm. Therefore, the performance of the three models is similar at this stage. In the stage when the target is in the channel, the agents under the



**Fig. 13.** Snapshots of simulation at  $t = 1$  s and  $t = 30$  s applying the CH-GRN model with and without the cooperation mechanism and the H-GRN model to guide the agents to entrap the target in a channel. (a) Initial setting of entrapment. (b) Entrapment under CH-GRN at  $t = 30$  s. (c) Entrapment under CH-GRN without cooperation at  $t = 30$  s. (d) Entrapment under H-GRN at  $t = 30$  s.

control of the complete CH-GRN model are deployed below the target, whereas the other swarm is blocked above the target. During this stage, therefore, the performance of the model without the cooperation mechanism and H-GRN gradually becomes increasingly lower than that of the complete CH-GRN. At  $t = 30$  s, the target escapes from the channel; the details are shown in Fig. 13. After that point, the target remains outside the channel, and the swarm controlled by the complete CH-GRN continues surrounding the target, whereas the swarm controlled by the model without the cooperation mechanism is just beginning to complete the entrapment task. Throughout the operation, the enclosure occupancy rate under the complete CH-GRN fluctuates around 0.75 because of the completion of the encirclement in the channel. However, because the entrapment strength of the obstacle is set to a value higher than that of the agents, the entrapment strength of the target decreases slightly as the target escapes from within the obstacle. Throughout the operation, the complete CH-GRN model maintains better performance than the version without the cooperation mechanism and H-GRN. The comparison can also be seen from the snapshots at  $t = 20$  s and  $t = 40$  s in Fig. 13. The enclosure integrity of H-GRN is significantly lower than that of the other two models during the entrapment, because they are limited by the channel and difficult to find the optimal motion direction to entrap the target.

#### 4.1.4. Entrapping with dynamic obstacles

To evaluate the adaptability of CH-GRN to a complex environment, this simulation is performed with the assumption that obstacles in the task environment have the ability to move, which leads to continuous change in the motion areas of target and swarm. Fig. 15 shows the changes in the task environment and the swarm under the control of CH-GRN throughout the simulation. In the task environment, obstacles are arranged to form a narrow channel and allowed to move left and right, and the target moves vertically downward following the trajectory.

In Fig. 15(a), the black circles indicate the initial positions of the target, obstacles, and agents at  $t = 1$  s, and the circles are filled with

colour at  $t = 20$  s. As  $t$  changes from 1 to 20, the obstacles move closer to the centre and compress the space available for agents to move. During this time, the target remains in motion, and the agents do their best to surround it. At  $t = 20$  s, the space available for agent motion become very small, but it can be seen that some agents have already reached the space below the target. Fig. 15(b) shows the entrapment task status at  $t = 20$  s and  $t = 50$  s. As  $t$  changes from 20 to 50, the obstacles separate to the different sides, and the agents can return to their normal states with less influence from the obstacles. As a result, the agents complete the entrapment task before  $t = 50$  s.

The lines in Fig. 16 show the results of the quantitative evaluation. During the stage before the change in the direction of obstacle movement, both the enclosure occupancy rate and the entrapment strength are increasing, and after that, they decrease substantially. The reason is that the obstacles can effectively assist in the entrapment of the target. When the direction of the obstacle changes, the obstacles around the target almost fill the whole enclosure, resulting in high enclosure occupancy rate and strength of the encirclement. It can be seen that throughout the operation, the entrapment strength of the agents controlled by the complete CH-GRN is better than that of the model without the cooperation mechanism; this is because the CH-GRN is more effective in driving the agents to reach the space below the target and thereby complete the encirclement. The relative advantage of entrapment strength is smaller than the enclosure occupancy rate because obstacles near the target have a greater effect on the entrapment strength than on the enclosure occupancy rate. In fact, CH-GRN shows its advantages in the later stage of the target entrapment when the obstacles gradually separate. As for CH-GRN without cooperation mechanism, the concentration-vector method realises the accurate motion direction selection of the swarm robots. In addition, the cooperation mechanism allows swarm robots to find the empty area of the enclosure, so as to complete the enclosure faster than under the model without the cooperation mechanism and H-GRN.

The decision-making times of each agent in the swarm within 50 simulation steps are summarised in Fig. 17. It can be seen that the decision times are around 0.64s, and there is a slight fluctuation due to the different amounts of information obtained. Since the computing power of each agent is set to be the same, the fluctuation is slight. In summary, The experimental results on the entrapment strength and enclosure occupancy rate show that the decision-making time is short enough to enable the swarm to complete the entrapping task efficiently. Thus the proposed method is helpful to the real-time decision-making of the swarm to cope with the changing task environment.

## 4.2. Experiments on Kilobots

To validate the results on an actual robotic platform, a set of comparative experiments are designed to test the proposed model with Kilobots. The task area is 65 cm  $\times$  50 cm; a number of robots are placed at the boundary to prevent the entrapping robots from going outside this area. The experimental video showing the entrapment process with Kilobots can be downloaded from OneDrive<sup>1</sup>.

Twelve Kilobots are used for entrapping three targets. The entrapping robots emit blue light, and the target robots emit red light. As a robot's perception of its surroundings is received only via infrared rays, it is unable to perceive ordinary physical obstacles. Thus, in this experiment, five Kilobots are used to simulate a trapezoidal obstacle, following the analogous concept of utilising virtual robots to represent obstacles in the CH-GRN. These obstacle robots, which emit black light, form the obstacle boundary. Targets are given the capability of simple mobility; they will move towards areas without robots to attempt to escape entrapment.

<sup>1</sup> <https://1drv.ms/f/s!ArltQV6iuxl0hGuH9lJOURM2k-V>

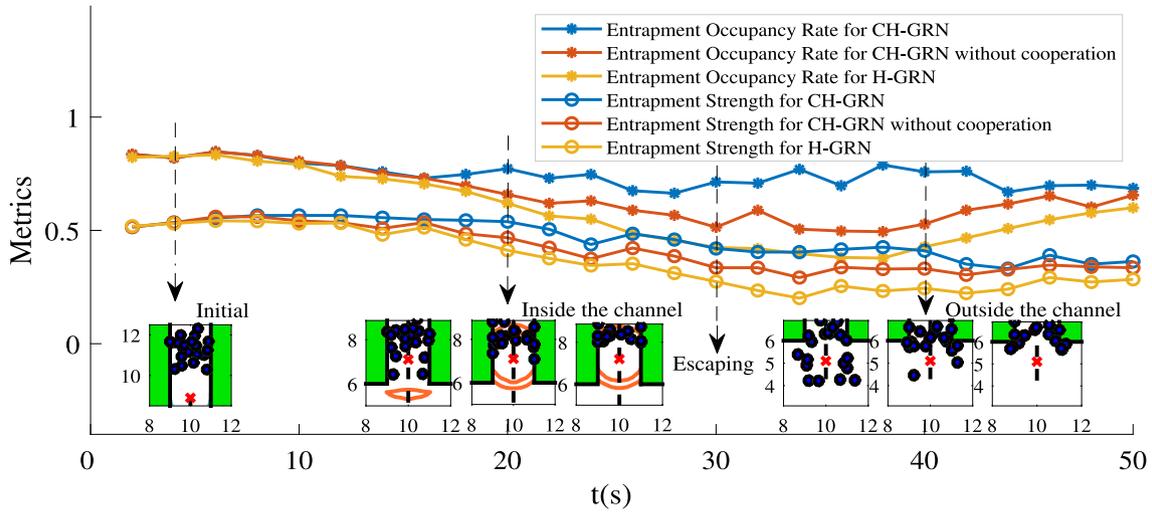


Fig. 14. Variation in enclosure occupancy rate and entrapment strength of the swarm under the complete CH-GRN model and the version without the cooperation mechanism, during the target's escape from the channel. The snapshots show the positions of a target and agents under the complete CH-GRN model, CH-GRN without the cooperation mechanism, and the H-GRN model, respectively. Corresponding to the curves, the more agents around the target, the higher the enclosure occupancy rate, and the better the entrapping performance.

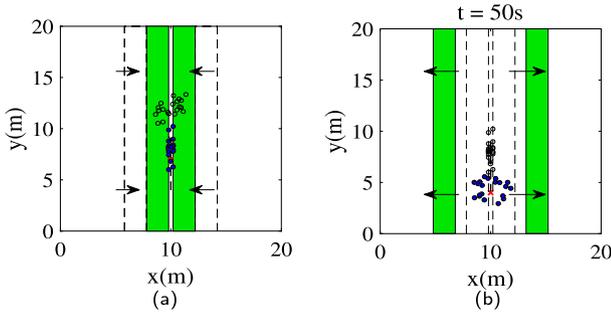


Fig. 15. Changes in task environment and swarm under control of CH-GRN. (a) Task state changes from  $t = 1$  s to  $t = 20$  s. (b) Task state changes from  $t = 20$  s to  $t = 50$  s.

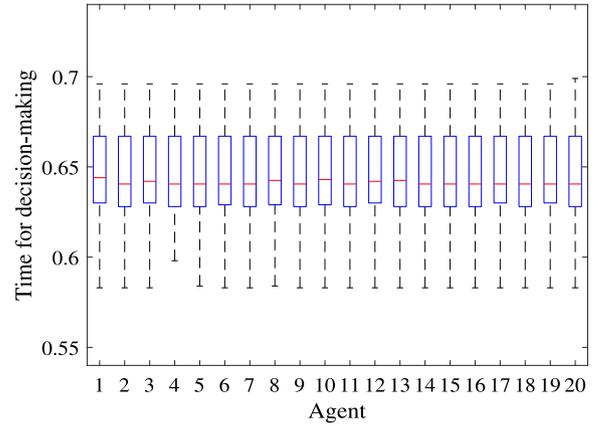


Fig. 17. The decision-making times of each agent.

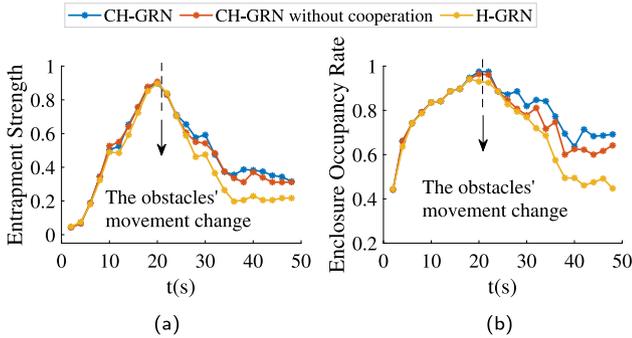


Fig. 16. Effect of cooperation mechanism in CH-GRN during entrapment of a target using dynamic obstacles. (a) Entrapment strength. (b) Enclosure occupancy rate.

A Kilobot can only process the local concentration information. Because a Kilobot has no sense of direction, the vector-concentration method cannot be implemented. Therefore, a simplified version of the lower layer of CH-GRN is used here, i.e. the robots simply approach the high-concentration area to entrap the targets.

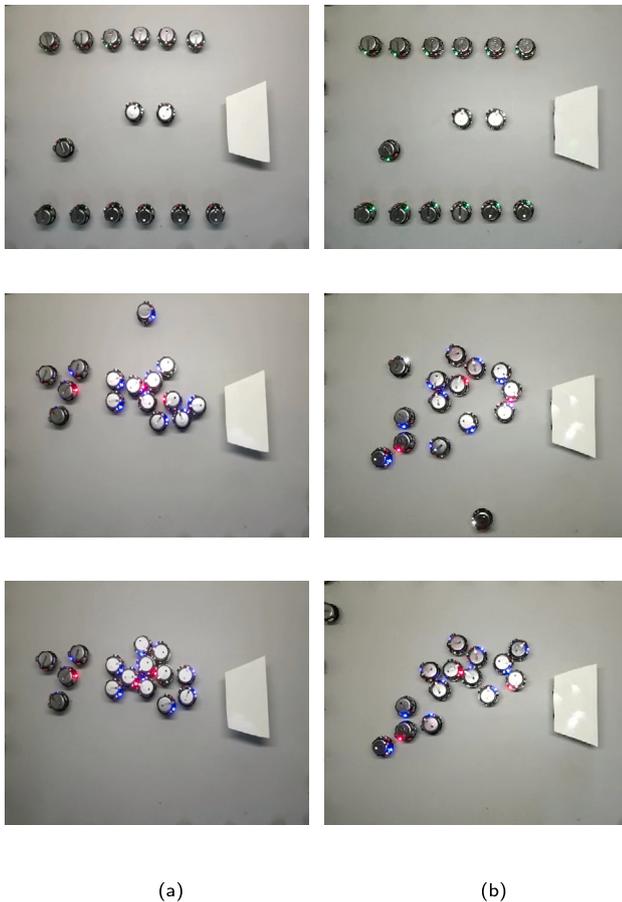
In the first run, the CH-GRN programme is uploaded to the Kilobots. The robots are initialised to be deployed along the boundary of the area. Fig. 18(a) shows photographs of the entrapment task operation, where it can be observed that the three targets have been successfully

surrounded. In addition, obstacles play a role in the target entrapment operation, reducing the number of entrapment robots needed.

For comparison, the cooperation mechanism is removed from the pattern generation process. Specifically, under the model without the cooperation mechanism, the protein concentration around the target will not be affected by surrounding neighbours or obstacles. This trial is performed in the same task environment; the results are shown in Fig. 18(b). It took 82 s and 121 s for the Kilobots to reach the final state with and without the cooperation mechanism, respectively. Without the cooperation mechanism, the Kilobots will not actively move towards areas of weakness in the encirclement, and the time required to achieve entrapment is obviously longer (compare Fig. 18(a)). In addition, the entrapment outcome is not good even using the same number of robots because of the obvious gaps in the encirclement. Specifically, there is a large gap to the right of the target nearest the obstacle, and it will be easy for the target at the bottom to escape the encirclement.

#### 4.3. Discussion

One impact of this work is to take a step towards the transformation of the entrapment task completion mechanism of swarm robots. The relationship between swarm robots and the task environment has



**Fig. 18.** Process using Kilobots in a target entrapment task. Twelve Kilobots are used for entrapping three targets. The entrapping robots emit blue light, and the target robots emit red light. (a) Entrapment under adjusted version of CH-GRN. (b) Entrapment under adjusted version of CH-GRN without the cooperation mechanism.

changed from being passively limited to active utilisation and cooperation. Our previous positive results show that target entrapment by swarm robots can indeed be improved with the cooperation mechanism. In particular, the work presented in this paper contributes by determining that performance (1) can improve if the swarm robots cooperate with obstacles and (2) is still generally positive even if the obstacles in the task scene are complex and dynamic.

The swarm robots' analysis of the encirclement state will affect their next movement direction, thus affecting the speed of entrapment completion. In the simulation experiments, the targets in the scene are dynamic, which increases the difficulty of the target entrapment task. When the targets are dynamic and there are obstacles around, it should be known that moving in the direction of the target moving and without obstacles will be a good choice. Under the proposed CH-GRN, the robots can generate the entrapment pattern by processing the input information, which indicates the best position for the status of the current encirclement. Then the robots quickly adapt to the pattern and complete the target entrapment task. However, robots under H-GRN do not know the best direction and they are only satisfied with being in the encirclement. They spend a lot of time completing the encirclement by mutual repulsion.

Of course, the proposed CH-GRN model cannot be applicable if the performance of the robots makes it unable to obtain and analyse the information of obstacles. The improvements of the CH-GRN model hint that it may perform well in the task scenes in which obstacles have a great impact on robots' behaviour. In order to improve the applicability of this method for simple robots, we leave the task of implementing a more efficient or simple pattern generation approach for future work.

## 5. Conclusion and future work

In this paper, a cooperative hierarchical gene regulatory network (CH-GRN) has been proposed for the entrapment of targets. In CH-GRN's upper layer, robots enhance their mutual cooperation by using the proposed target–neighbour–obstacle (TNO) pattern generation method. The entrapment pattern depends on information not only about the targets but also about the nearby robots and obstacles. In CH-GRN's lower layer, the concentration-vector method is used to drive the robots to form the pattern and thereby create effective encirclements around the targets.

To evaluate the model of the network, several experiments are conducted, including both simulations and physical tests. During the experiments, to entrap the targets, the swarms are deployed in the task scene that included obstacles. By adding the proposed cooperation mechanism, the swarm finds and moves to the blank areas of the enclosure thus accelerating encirclement. Simultaneously, because the obstacles are regarded as part of the enclosure in the CH-GRN model, the area of the surrounding circles that needs to be filled by swarm robots is reduced. Consequently, the enclosure surrounding the target is stronger. The results show that the cooperation among robots and with obstacles, embedded in CH-GRN, can substantially improve the entrapment performance.

The architecture of the CH-GRN model used in this study is designed manually. However, it is difficult to adapt manually designed architectures for other tasks of wider variety and greater complexity. Therefore, algorithms to optimise the design of the model will be investigated in the future to enhance the adaptability of swarm robots to a broader spectrum of challenging environments.

## CRedit authorship contribution statement

**Meng Wu:** Writing – original draft, Methodology, Investigation. **Xiaomin Zhu:** Writing – review & editing, Funding acquisition. **Li Ma:** Conceptualization, Validation. **Weidong Bao:** Supervision, Project administration. **Zhun Fan:** Writing – review & editing, Project administration, Funding acquisition. **Yaochu Jin:** Conceptualization, Supervision.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request

## References

- [1] Guang-Zhong Yang, Jim Bellingham, Pierre Dupont, Peer Fischer, Luciano Floridi, Robert Full, Neil Jacobstein, Vijay Kumar, Marcia McNutt, Robert Merrifield, Brad Nelson, Brian Scassellati, Mariarosaria Taddeo, Russell Taylor, Manuela Veloso, Zhong Wang, Robert Wood, The grand challenges of science robotics, *Sci. Robot.* 3 (2018) eaar7650.
- [2] Gábor Vásárhelyi, Csaba Virágh, Gergő Somorjai, Tamás Nepusz, A. Eiben, Tamás Vicsek, Optimized flocking of autonomous drones in confined environments, *Sci. Robot.* 3 (2018) eaat3536.
- [3] Michael Rubenstein, Alejandro Cornejo, Radhika Nagpal, Robotics. Programmable self-assembly in a thousand-robot swarm, *Science* 345 (2014) 795–799.
- [4] Jianing Chen, Melvin Gauci, Wei Li, Andreas Kolling, Roderich Groß, Occlusion-based cooperative transport with a swarm of miniature mobile robots, *IEEE Trans. Robot.* 31 (2015) 307–321.
- [5] A. Dongshu Wang, B. Haitao Wang, C. Lei Liu, Unknown environment exploration of multi-robot system with the FORDPSO - ScienceDirect, *Swarm Evol. Comput.* 26 (2016) 157–174.

- [6] Lorenzo Garattoni, Mauro Birattari, Autonomous task sequencing in a robot swarm, *Sci. Robot.* 3 (2018) eaat0430.
- [7] Ashish Macwan, Julio Vilela, Goldie Nejat, Beno Benhabib, A multirobot path-planning strategy for autonomous wilderness search and rescue, *IEEE Trans. Cybern.* 45 (2015) 1784–1797.
- [8] Xin Yi, Anmin Zhu, Simon Yang, Chaomin Luo, A bio-inspired approach to task assignment of swarm robots in 3-D dynamic environments, *IEEE Trans. Cybern.* 47 (2017) 974–983.
- [9] Lynne Parker, Claude Touzet, Multi-robot learning in a cooperative observation task, in: *Proc. Distributed Autonomous Robotic Systems 4*, DARS, Tennessee, USA, 2000, pp. 391–402.
- [10] Marco Mamei, Ronaldo Menezes, Robert Tolksdorf, Franco Zambonelli, Case studies for self-organization in computer science, *J. Syst. Archit. Euromicro J.* 52 (2006) 443–460.
- [11] Sucheendra Palaniappan, Ayako Yachie-Kinoshita, Samik Ghosh, Computational systems biology, *Nature* 420 (2002) 206–210.
- [12] Hiroki Sayama, Robust morphogenesis of robotic swarms [application notes], *IEEE Comput. Intell. Mag.* 5 (2010) 43–49.
- [13] Asif Khan, Bernhard Rinner, Andrea Cavallaro, Cooperative robots to observe moving targets: Review, *IEEE Trans. Cybern.* 48 (2018) 187–198.
- [14] Yaochu Jin, Surrogate-assisted evolutionary computation: Recent advances and future challenges, *Swarm Evol. Comput.* 1 (2011) 61–70.
- [15] Hyondong Oh, Yaochu Jin, Evolving H-GRNs for morphogenetic adaptive pattern formation of swarm robots, *Evol. Comput. Gene Regul. Netw. Res.* 1 (2016) 327–361.
- [16] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, Marco Dorigo, Swarm robotics: a review from the swarm engineering perspective, *Swarm Intell.* 7 (2013) 1–41.
- [17] C.J.M. Verhoeven, M.J. Benthum, J. Rotteveel, B. Monna, On the origin of satellite swarms, *Acta Astronaut.* 68 (2011) 1392–1395.
- [18] Anam Tahir, Jari Böling, Mohammad-Hashem Haghbayan, Hannu Toivonen, Juha Plosila, Swarms of unmanned aerial vehicles - a survey, *J. Ind. Inf. Integr.* 16 (2019) 100106.
- [19] Hyondong Oh, Yaochu Jin, Adaptive Swarm Robot Region coverage using gene regulatory networks, in: *Proc. Conference Towards Autonomous Robotic Systems, TAROS*, Birmingham, UK, 2014, pp. 197–208.
- [20] Aytac Altan, Rifat Hacioglu, Model predictive control of three-axis gimbal system mounted on UAV for real-time target tracking under external disturbances, *Mech. Syst. Signal Process.* 138 (2020) 1–23.
- [21] Aytac Altan, Performance of metaheuristic optimization algorithms based on swarm intelligence in attitude and altitude control of unmanned aerial vehicle for path following, 2020, pp. 1–6.
- [22] Aytac Altan, Özgür Aslan, Rifat Hacioglu, Real-time control based on NARX neural network of hexarotor UAV with load transporting system for path tracking, 2018, pp. 1–6.
- [23] Antonio Franchi, Paolo Stegagno, Giuseppe Oriolo, Decentralized multi-robot encirclement of a 3D target with guaranteed collision avoidance, *Auton. Robots* 40 (2015) 245–265.
- [24] Hitoshi Iima, Yasuaki Kuroe, Swarm reinforcement learning method for a multi-robot formation problem, in: *IEEE International Conference on Systems, Man, and Cybernetics, Manchester, SMC 2013*, United Kingdom, October 13–16, 2013, 2013, pp. 2298–2303.
- [25] D. Dormann, Bakhtier Vasiev, Cornelis Weijer, Becoming multicellular by aggregation; The morphogenesis of the social amoebae *dicyostelium discoideum*, *J. Biol. Phys.* 28 (2002) 765–780.
- [26] Hyondong Oh, Ataollah Ramezan Shirazi, Chaoli Sun, Yaochu Jin, Bio-inspired self-organising multi-robot pattern formation: A review, *Robot. Auton. Syst.* 91 (2016) 83–100.
- [27] Esben Ostergaard, David Christensen, Peter Eggenberger, Tim Taylor, Peter Ottery, Henrik Lund, HYDRA: From cellular biology to shape-changing artefacts, in: *Proc. International Conference on Artificial Neural Networks, ICANN*, Warsaw, Poland, 2005, pp. 275–281.
- [28] Qiuzhen Wang, Xinjun Mao, Shuo Yang, Yin Chen, Xinwang Liu, Grouping-based adaptive spatial formation of swarm robots in a dynamic environment, *Int. J. Adv. Robot. Syst.* 15 (2018) 1–14.
- [29] Hyondong Oh, Yaochu Jin, Evolving hierarchical gene regulatory networks for morphogenetic pattern formation of swarm robots, in: *Proc. IEEE Congress on Evolutionary Computation, CEC*, Beijing, China, 2014, pp. 776–783.
- [30] Qi Juntong, Guo Jinjin, Wang Mingming, Wu Chong, Ma Zhenwei, Formation tracking and obstacle avoidance for multiple quadrotors with static and dynamic obstacles, *IEEE Robot. Autom. Lett.* 7 (2) (2022) 1713–1720.
- [31] Tang Qirong, Xu Zhipeng, Yu Fangchao, Zhang Zhongqun, Zhang Jingtao, Dynamic target searching and tracking with swarm robots based on stigmergy mechanism, *Robot. Auton. Syst.* 120 (2019).
- [32] Linda Thomas, Julian Wimpenny, Gary Barker, Spatial interactions between subsurface bacterial colonies in a model system: a territory model describing the inhibition of *listeria monocytogenes* by a nisin-producing lactic acid bacterium, *Microbiology* 143 (1997) 2575–2582.
- [33] Hans Meinhardt, From fertilized eggs to complex organisms: Models of biological pattern formation, in: *Proc. 5th European Conference on Advances in Artificial Life, ECAL*, Lausanne, Switzerland, 1999, pp. 3–4.
- [34] Philip Ingham, The molecular genetics of embryonic pattern formation in *drosophila*, *Nature* 335 (1988) 25–34.
- [35] Rushikesh Sheth, Luciano Marcon, M. Bastida, Marisa Junco, Laura Quintana, Randall Dahn, Marie Kmita, James Sharpe, Marian Ros, Hox genes regulate digit patterning by controlling the wavelength of a turing-type mechanism, *Science* 338 (2012) 1476–1480.
- [36] Justin Werfel, Biologically realistic primitives for engineered morphogenesis, in: *Proc. 7th International Conference on Swarm Intelligence, ANTS*, Brussels, Belgium, 2010, pp. 131–142.
- [37] I. Slavkov, D. Carrillo-Zapata, N. Carranza, Xavier Diego, Fredrik Jansson, Jaap Kaandorp, S. Hauer, J. Sharpe, Morphogenesis in robot swarms, *Sci. Robot.* 3 (2018) eaau9178.
- [38] U. Kaupp, Nachiket Kashikar, Ingo Weyand, Mechanisms of sperm chemotaxis, *Annu. Rev. Physiol.* 70 (2008) 93–117.
- [39] Manolya Eyiurekli, Linge Bai, Peter Lelkes, David Breen, Chemotaxis-based sorting of self-organizing heterotypic agents, in: *Proc. ACM Symposium on Applied Computing, SAC*, Sierre, Switzerland, 2010, pp. 1315–1322.
- [40] Diego Federici, Keith. Downing, Evolution and development of a multicellular organism: Scalability, resilience, and neutral complexification, *Artif. Life* 12 (2006) 381–409.
- [41] Chen Wang, Zhun Fan, Weicheng Luo, Minqiang Gu, Dongliang Wang, Zhaohui Shi, Collaborative navigation based on gene regulatory networks and finite state machine, in: *2022 8th International Conference on Big Data and Information Analytics, Vol. 1*, No. 1, BigDIA, 2022, pp. 12–19.
- [42] Yaochu Jin, Hongliang Guo, Yan Meng, Robustness analysis and failure recovery of a bio-inspired self-organizing multi-robot system, in: *Proc. IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO*, California, USA, 2009, pp. 154–164.
- [43] Hongliang Guo, Yan Meng, Yaochu Jin, Swarm robot pattern formation using a morphogenetic multi-cellular based self-organizing algorithm, in: *Proc. IEEE International Conference on Robotics and Automation, ICRA*, Shanghai, China, 2011, pp. 3205–3210.
- [44] Yaochu Jin, Hongliang Guo, Yan Meng, A hierarchical gene regulatory network for adaptive multirobot pattern formation, *IEEE Trans. Syst. Man Cybern. B* 42 (2012) 805–816.
- [45] Yan Meng, Hongliang Guo, Evolving network motifs based morphogenetic approach for self-organizing robotic swarms, in: *Proc. Genetic and Evolutionary Computation Conference, GECCO*, PA, USA, 2012, pp. 137–144.
- [46] Ze Shi, Zhun Fan, Peili Ma, Juncao Hong, Wenji Li, Guijie Zhu, Huaxing Huang, Yuwei Cai, Zhaohui Dong, Xiaomin Zhu, A cooperative gene regulatory network target entrapping in GNSS-denied environments, in: *2022 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, Vol. 1*, No. 1, CYBER, 2022, pp. 382–387.
- [47] Fan Zhun, Ma Peili, Zhu Guijie, Xie Minchong, Chen Tianshan, Xie Fei, Shi Ze, Bao Weidong, Zhu Xiaomin, Method of swarm aggregation and control for intelligent RobotBased on three-dimensional gene regulatory network, *IEEE Transl. J. NanJing Norm. Univ. (Eng. Technol. Ed.)* 1 (2022) 9–15.
- [48] Meng Wu, Yun Zhou, Xiaomin Zhu, Li Ma, Yutong Yuan, Taosheng Fang, Ji Wang, Weidong Bao, Zhun Fan, Cooperation-based gene regulatory network for target entrapment, in: *Proc. International Conference on Swarm Intelligence, ICSI*, Chiangmai, Thailand, 2019, pp. 60–69.
- [49] Michael Rubenstein, Christian Ahler, Radhika Nagpal, Kilobot: A low cost scalable robot system for collective behaviors, in: *Proc. IEEE International Conference on Robotics and Automation, ICRA*, Minnesota, USA, 2012, pp. 3293–3298.
- [50] Michael Rubenstein, Alejandro Cornejo, Radhika Nagpal, Robotics. Programmable self-assembly in a thousand-robot swarm, *Science* 345 (2014) 795–799.
- [51] Ataollah Ramezan Shirazi, Yaochu Jin, Regulated morphogen gradients for target surrounding and adaptive shape formation, *IEEE Trans. Cogn. Dev. Syst.* 1 (2020) 1–11.
- [52] Nadav Kashtan, Uri Alon, Spontaneous evolution of modularity and network motifs, *Proc. Natl. Acad. Sci. USA* 102 (2005) 13773–13778.