A Hierarchical Gene Regulatory Network for Adaptive Multi-Robot Pattern Formation

Yaochu Jin, *Senior Member, IEEE,* Hongliang Guo, and Yan Meng, *Member, IEEE*

***Abstract*— Most existing multi-robot systems for pattern formation rely on a predefined pattern, which is impractical for dynamic environments where the pattern to be formed should be able to change as the environment changes. In addition, adaptation to environmental changes should be realized based only on local perception of the robots. In this work, we propose a hierarchical gene regulatory network (H-GRN) for adaptive multi-robot pattern generation and formation in changing environments. The proposed model is a two-layer gene regulatory network (GRN), where the first layer is responsible for adaptive pattern generation for the given environment, whilst the second layer is a decentralized control mechanism that drives the robots onto the pattern generated by the first layer. An evolutionary algorithm is adopted to evolve the parameters of the GRN subnetwork in layer 1 for optimizing the generated pattern. The parameters of the GRN in layer 2 are also optimized to improve the convergence performance. Simulation results demonstrate that the H-GRN is effective in forming the desired pattern in a changing environment. Robustness of the H-GRN to robot failure is also examined. A proof-of-concept experiment using e-puck robots confirms the feasibility and effectiveness of the proposed model.**

Keywords: Hierarchical gene regulatory networks, multi-robot pattern generation and formation, dynamic environment, self-organization, evolutionary algorithms

I. INTRODUCTION

Multi-robot systems (MRSs) are composed of a large number of small and simple robots, each having limited communication capability and computational resources. Therefore, robots in an MRS must work together to collectively accomplish complex tasks that are beyond the capability of any single robot. In addition, since they are often expected to work in hazardous and changing environments, MRSs should be able to function properly in the presence of uncertainties and partial system failures. Due to their attractive properties such as low-cost, strong robustness, and high adaptability, MRSs have found a wide range of successful applications, including collaborative search and rescue [2], [54], large object transportation and manipulation [42], [44], [66], aggregation and segregation [43], [48], cooperative localization and mapping [41], [56], shape formation and flocking [13], [28], [68], and collective construction [45], [47], [67].

Multi-robot shape construction and pattern formation, a typical task for MRSs, has been widely studied. Algorithms in this research field can be roughly divided into three groups, namely, leader/neighbor-following algorithms, potential field algorithms, and nature-inspired algorithms. Leader/neighbor-following algorithms [6], [38], [51] require that individual robots follow leader(s) that know where to go, or follow neighbors that are following leader(s). Meanwhile, the following robots should maintain a specific geometric relationship with the ones they follow. The second group of multi-robot shape construction algorithms is based on potential field method [28], [16]. The basic idea of this group of algorithms is that each robot moves under the governance of the gradients of potential fields, which are the sum of virtual attractive and repulsive forces. The third group is nature-inspired algorithms. A gas expansion model inspired by pheromone and flocking has been suggested by Cheng et al [7] to dispatch robots within a predefined shape. Shen et al [59] introduced a digital hormone model (DHM) for distributed multi-robot control using principles underlying biological development such as reaction and diffusion [65]. Mamei et al [40] have used a computational model of morphogen gradients for multi-robot pattern generation, where the robots can communicate with their neighbors to receive and propagate morphogen gradients. Bloom et al [3] studied a cell-based approach to arbitrary 2D-shape assembling. Sayama [58] has reported a distributed control algorithm for generating spatiotemporal patterns based on simple kinetic rules. It has been shown that the resulting patterns are robust to environmental changes and partial damage in the pattern. Most recently, morphogenetic robotics [32], which employs genetic and cellular mechanisms inspired from biological morphogenesis, has become a new emerging field of robotics for self-organization of swarm or modular robots. Guo et al [24] have proposed a decentralized control algorithm for multi-robot shape construction by establishing a metaphor between multi-cellular systems and multi-robot systems. The GRN model is evolved using a multi-objective evolutionary algorithm to optimize the construction performance, such as minimization of convergence time and travel distance. A variant of the algorithm has also been reported in [46] to enable the algorithm to construct complex 2D or 3D patterns by introducing a freeform shape representation, which also removes the dependance on the availability of a global coordinate system [23].

A substantial limitation of most existing MRSs for shape construction is that the target shape must be predefined. As a result, the generated patterns are not adaptable to unknown environmental changes [58], [59]. An additional limitation of many control algorithms for MRSs is that they can form only a small number of simple shapes [58], [62]. Little work has been reported on a decentralized algorithm for MRSs that can

Yaochu Jin is with the Department of Computing, University of Surrey, Guildford, Surrey, GU2 7XH, UK. Email: *yaochu.jin@surrey.ac.uk*.

Hongliang Guo and Yan Meng are with the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07030, USA. Email: {*hguo,yan.meng*}*@stevens.edu*.

adaptively generate a target pattern in a controllable manner in the presence of environmental changes.

It has been found that hierarchy in GRNs plays a central role in the evolution of developmental gene regulatory networks [17]. Findings in developmental biology also suggest that a hierarchical gene regulatory network is responsible for the patterning strategies in Drosophila [9]. In computational modeling of biological GRNs for pattern formation, hierarchical GRNs have shown to be able to generate stable complex patterns [12]. Although the exact reasons behind the importance of a hierarchical gene regulatory network remain elusive, we hypothesize that hierarchy makes it possible for GRNs to produce a modular functional structure, resulting in better adaptability and evolvability. Motivated from the above findings, we adopted an H-GRN in this paper for designing a decentralized control mechanism for adaptive multi-robot pattern generation in changing environments. The H-GRN consists of two layers. Layer 1 is a GRN for pattern generation, which is able to generate various complex patterns under different environments based on local sensory inputs. Layer 2 is a GRN model whose regulatory dynamics can be influenced by that of layer 1, thus driving the robots to the target pattern generated by layer 1.

The rest of the paper is organized as follows. Section II provides a detailed problem statement and lists a few assumptions based on which this model works. Section III presents a brief introduction to computational models of developmental gene networks and introduces a metaphor illustrating how genetic and cellular mechanisms underlying biological morphogenesis can be applied to self-organization of multi-robot systems. The proposed H-GRN for adaptive multi-robot pattern formation in changing environments is presented in Section IV. Section V discusses the evolutionary optimization of the parameters in the H-GRN. The performance of the H-GRN based control algorithm is evaluated with simulations in Section VI. A proof-of-concept experiment using e-puck robots in an indoor environment is reported in Section VII. Section VIII discusses exisiting work that is related to this research but has a different focus. Section IX concludes the paper and discusses future work.

II. Problem Statement and Assumptions

A. Problem Statement

The problem we are addressing is to entrap stationary and/or mobile targets using a few mobile robots. The entrapping task consists of two steps, namely, pattern generation and pattern formation. During pattern generation, the robots are expected to generate an appropriate pattern according to the number and location of the targets. In pattern formation, the robots should deploy themselves onto the generated pattern to entrap the targets without a centralized control.

B. Assumptions

In order to employ the H-GRN for adaptive multi-robot pattern formation, the following assumptions have been made:

1) The robots can localize themselves at anytime with their on-board sensors, such as encoders and sonar sensors. This assumption can be lifted by building up a local coordinate system via local communications among the robots, as reported in [23].
2) There is a base station containing a sufficient number of robots. Once the robots in mission cannot accomplish the pattern formation task, e.g., the perimeter of the generated target pattern is too long to be fully covered by the robots in mission, they can call for additional robots from the base station. On the other hand, if some robots are no longer needed in entrapping the existing targets, these robots can find their way back to the base station.
3) All robots have a limited sensing range and therefore, they can detect targets and other robots that are within their sensing range only.
4) The communication range between robots is also limited. Robots can communicate information such as targets' location and velocity with their immediate neighbors. Immediate neighbors mean that the distance between the two robots is within the communication range. We assume that the communication between the robots and the base station is not limited, which, however, does not imply that the proposed system is subject to the same weakness of a centralized system.
5) The movement of the robots is much faster than the targets. Consequently, we can assume that the target patterns generated in different robots are the same and that the robots are able to keep entrapping the mobile targets.
6) All targets in the concerned region can be detected by at least one robot. To satisfy this assumption, the coverage algorithm proposed in [27] has been adopted in this work to deploy the robots in the region during initialization.

C. Terminologies

In this paper, *patterns* refer to 1D curves (shapes) embedded in a 2D workspace or 2D surfaces in a 3D workspace. In the examples in this work, only 1D curves are involved. *Targets* are the static or moving objects in the environment that need to be encircled by the robots. *Organizing robots* refer to the robots that detect at least one target in the environment. Non-organizing robots (follower robots) refer to those that have not yet detected any target in the environment. Non-organizing robots can become organizing robots if they detect a target. *Target patterns* refer to the patterns generated by the organizing robots that can encircle the targets. A target pattern is generated by the GRN in layer 1 based on the locations of the detected targets.

III. Biological Background

A. Biological Morphogenesis and Gene Networks

Biological morphogenesis is the biological process in which cells divide, grow and differentiate, and finally resulting in the mature morphology of a biological organism. Morphogenesis is under the governance of a developmental gene regulatory network and the influence of the environment [20]. This environment includes concentration gradients of substances known as morphogens, which are responsible for cell specialization and migration. Morphogen gradients are either directly present in the environment of the fertilized cell (maternal gradients) or generated by a few cells known as organizers [5].

To understand biological development, a large number of GRN models have been suggested [10] either for reconstructing developmental subnetworks based on biological data [18], [29], [57], or for simulating biological development in computational environments [26], [60] for solving engineering problems, such as structural design [15], [61], electronic circuits design [70], control [55], [63] and self-configuration of modular robots [49], [50]. Furthermore, computational GRN models have been used for analyzing fundamental properties of GRNs such as robustness and evolvabilily [8], [33], and for synthesizing typical regulatory dynamics [19], [31]. Among others, ordinary or partial differential equations (ODEs /PDEs) are the most widely used models.

B. A Metaphor between Multi-Cellular Organisms and Multi-Robot Systems

To employ the genetic and cellular mechanisms that govern biological morphogenesis for multi-robot pattern formation, it is necessary to establish a metaphor between multi-cellular organisms and multi-robot systems. In this metaphor, a cell is mapped to a robot, where protein concentrations in a cell correspond to the internal states or the location of the robot. The proteins that stand for the location of the robots can diffuse out of the cell to generate cell-cell interactions, which in the multi-robot systems is a distance-based mechanism that can change the movement dynamics of the robots to avoid collision. Finally, morphogen gradients in multi-cellular organisms are used to describe the target pattern to be formed by the MRS. The robots that first detect the targets in the environment are termed organizing robots, which are responsible for building up the target pattern.

There is, however, an important difference between a multi-cellular organism and a multi-robot system. In biological morphogenesis,the morphogen gradients are distributed in the form of biochemicals in the embryo, which are available to all cells inside the embryo. In a multi-robot system, by contrast, the target pattern cannot be distributed in terms of chemical concentrations in the environment. Instead, the target pattern can only be generated by the organizing robots (robots that detect all or a subset of the targets).

IV. The H-GRN Model

The proposed H-GRN for adaptive multi-robot pattern generation consists of two layers, as illustrated in Fig. 1. Fig. 1 shows the control methodology on pattern generation and formation from multi-cellular systems point of view. Layer 1 generates patterns in terms of protein concentrations depending on the location of the targets. The concentration of one particular protein (g_3) plays the role of morphogen and its gradients can be read in by layer 2. Although the two layers of the GRN-based controller have different dynamics (because they have different objectives, one is for pattern generation, and the other is for pattern formation), both models are abstracted from mathematical models for describing dynamics of gene expression. Actually, the model used in layer 1 is adapted from a GRN model for pattern formation of a multi-cellular system [1] and the model in layer 2 was adapted from [57]. We will discuss the detailed diffusion process on layer 1 and layer 2 in robotic systems in the following two subsections, respectively.

The dynamics of the GRN in layer 1 is activated only in the organizing robots, which detect one or more targets. Based on the location of the detected targets, a target pattern will be generated by the GRN in layer 1. The generated pattern (the concentration of g_3) will be read in by the GRN in layer 2 to drive the organizing robot to the target pattern. Layer 2 provides the dynamics of robot movements with two vectors (G and P), which represent the current position and internal states of the robots, respectively. If the patterns are generated in a two-dimensional (2D) workspace, the vector length of G and P is two. In a 3D workspace, the dimension of both position and velocity vectors is three. So the target pattern may change as the organizing robot is moving toward the target pattern, because the number of targets the organizing robot can detect may change. Meanwhile, if a robot does not detect any target, it will follow the movement of the neighboring organizing robots. Here, the neighboring organizing robots can be the immediate neighbors of the robot or other organizing robots with which the robot can communicate through a series of local communications. Once the targets are within its sensing range, it will become an organizing robot and move toward the targets.

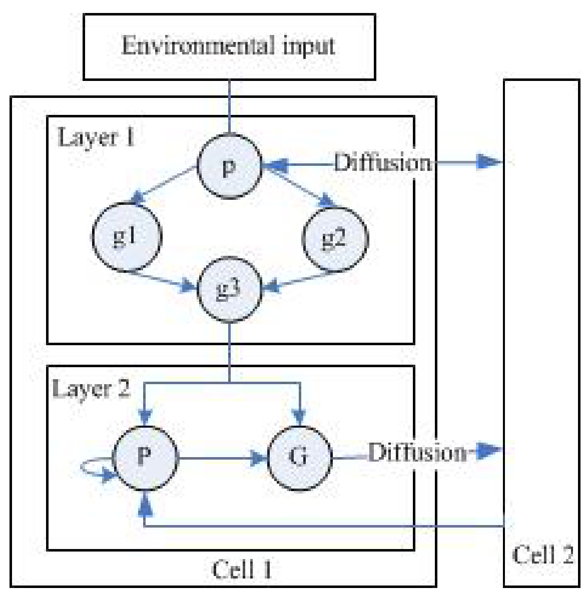


Fig. 1. A diagram of the H-GRN. Layer 1 is a GRN having four proteins, of which protein p can be regulated by environmental inputs, and protein g_3 is the morphogen gradient describing the pattern to be formed, which can influence the production of both proteins G and P.

A. Layer 1 of H-GRN: Pattern Generation

In our previous work, the target pattern to be formed by the robots is pre-defined explicitly using either an analytic function [24] or a freeform representation [46], such as the NURBS model [53]. In the present work, the target pattern is generated on-line using the GRN in layer 1 of the H-GRN.

It can happen that a robot detects only a subset of the targets. Therefore, the target patterns (in terms of protein concentration simulated in each robot) generated by different robots may be different when they are approaching to the targets. We assume that when the robots are close to the targets, they are able to know the position of all targets either detected by the robots themselves, or informed of by other organizing robots through local communication. In this way, the pattern generated by all robots will be the same after convergence.

During pattern generation, each organizing robot will simulate the following dynamic equations to generate chemical concentrations that define the target pattern. γ_j is a scalar value which represents the environmental inputs of these dynamic equations. γ_j holds a positive constant value at those positions in the environment with a target, and is zero elsewhere (without any target). Here, the organizing robot can either detect the target position through their on-board sensors or acquire the target position through neighboring organizing robots that know the target position.

$$\frac{dp_j}{dt} = \nabla^2 p_j + \gamma_j - p_j \tag{1}$$

$$p = \sum_{j=1}^{N_t} p_j \tag{2}$$

$$\frac{dg_1}{dt} = -g_1 + \text{sig}(p, \theta_1, k) \tag{3}$$

$$\frac{dg_2}{dt} = -g_2 + [1 - \text{sig}(p, \theta_2, k)] \tag{4}$$

$$\frac{dg_3}{dt} = -g_3 + \text{sig}(g_1 + g_2, \theta_3, k) \tag{5}$$

$$\text{sig}(x, z, k) = \frac{1}{1 + e^{-k(x-z)}} \tag{6}$$

where p_j represents a protein concentration that is produced from the environmental input (γ_j) resulting from the j-th target. p presents the integrated protein concentration at the current robot position produced by all the detected targets from $j = 1$ to N_t, where N_t is the total number of the detected targets. ∇^2 is the Laplacian operator, which is defined as the second-order derivative of p_j in the spatial domain and can be treated as the diffusion process in the biological system. g_1, g_2 and g_3 are protein concentrations, where g_3 is the morphogen gradient that defines the target pattern. In the model, g_3 is regulated by both g_1 and g_2, which are regulated by p_j. This setup can be easily implemented by the robotic system, as the targets can be detected by the on-board sensors. θ_1, θ_2 and θ_3 are thresholds of a sigmoid function. The integrated protein concentration p will impact the expression level of g_1 and g_2 through Eqns 3 and 4.

In layer 1, protein p will diffuse into neighboring cells and influence the dynamics of the same protein in neighboring cells. From the robotic control point of view, this diffusion process corresponds to obtaining the target information either through the direct detection (if the target is within the robot's sensing range) or through robot-robot communications from neighbors (if no target exists within the robot's sensing range). The target information is carried on by the summarized concentration value of the protein.

To summarize, the input of layer 1 is environmental input γ and the output is g_3. p, g_1 and g_2 are all internal states of the robot. γ represents the obtained target information either through the sensory input of the robot or through local communications with other organizing robots. The concentration distribution of g_3 represents the generated target pattern based on the detected target information. Note, however that if the robot does not detect any target by itself, layer 1 is not activated and the robot remains a follower robot.

Now let us explain briefly how g_3 can generate a stable pattern in the space based on different environmental inputs. For simplicity, we assume that there is only one target in the environment. At first, a constant input γ at the target position will be generated. Then, protein p will build up a concentration distribution that decays as the distance to the target increases, as shown in Fig. 2. From the figure, we can see that proteins g_1, g_2, and g_3 can be regulated by protein p only within a particular range of its concentration, refer to Equations (1)-(5). Particularly, protein g_3 can be regulated by protein p only when the concentration of p is between θ_1 and θ_2. Therefore, the activation pattern of g_3 resulting from one detected target in a two-dimensional space will be a band of circle.

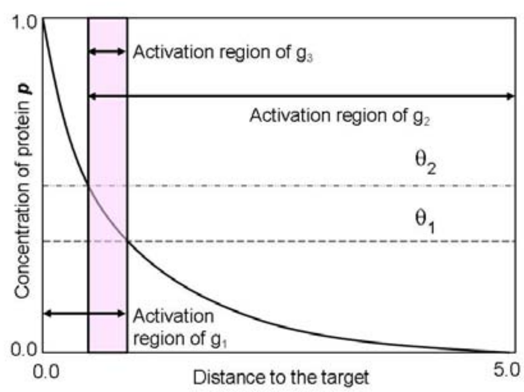


Fig. 2. The concentration distribution of protein p and the resulting expression profiles of proteins g_1, g_2 and g_3 in one-dimensional space.

In order for the generated target pattern to be used by the GRN in layer 2, it is necessary to extract the contour of the generated pattern and represent it with a NURBS model. This can be done by choosing a few representative points on the pattern and use these data points to generate control points of the NURBS representation. In the next section, we assume that the generated target pattern has already been represented by a number of control points of the NURBS curve.

B. Layer 2 of the H-GRN Model: Pattern Formation Layer

Once the target pattern is generated by layer 1 of the H-GRN model, it will function as the input of layer 2 to trigger its dynamics. The dynamics of the GRN in layer 2 guides the robots to the target pattern. In this work, we assume that each robot has a sensing range of r. The GRN in layer 2 is in principle the same as the model used in our previous work:

$$\frac{dG_{i,x}}{dt} = -a\mathbf{z}_{i,x} + mP_{i,x}, \quad (7)$$
$$\frac{dG_{i,y}}{dt} = -a\mathbf{z}_{i,y} + mP_{i,y}, \quad (8)$$
$$\frac{dP_{i,x}}{dt} = -cP_{i,x} + rf(\mathbf{z}_{i,x}) + bD_{i,x}, \quad (9)$$
$$\frac{dP_{i,y}}{dt} = -cP_{i,y} + rf(\mathbf{z}_{i,y}) + bD_{i,y}, \quad (10)$$

where $i = 1, 2, \ldots, n$, and n is the total number of organizing robots. $G_{i,x}$ and $G_{i,y}$ represent the x- and y-positions of the i-th organizing robot, which correspond to concentrations of two proteins of type G within cells. $P_{i,x}$ and $P_{i,y}$ are two internal states of the robots, which are the concentration of two proteins of type P in the cell. a, b and c are constants. D_i can be seen as the concentration of protein G that is diffused out of the cell. In the robotic system, it is a term that indicates the "density" of robots and obstacles in the neighborhood, i.e., the number of robots and obstacles in the neighborhood. The size of the neighborhood is pre-defined and should be smaller than the sensing range. It can be seen that simulating protein diffusion in multi-robot systems in this way is practical, as it requires only the detection of neighboring robots and obstacles. More specifically,

$$D_{i,x} = \sum_{j=1}^{N_i} D_{i,x}^j, \quad (11)$$
$$D_{i,y} = \sum_{j=1}^{N_i} D_{i,y}^j, \quad (12)$$

where N_i denotes the number of robots in the neighborhood of robot i, $D_{i,x}^j$ and $D_{i,y}^j$ represents the diffused protein concentrations along x- and y-axis received by robot i from robot j, respectively, which is defined as:

$$D_{i,x}^j = \frac{(G_{i,x} - G_{j,x})}{\sqrt{(G_{i,x} - G_{j,x})^2 + (G_{i,y} - G_{j,y})^2}} \quad , \quad (13)$$

$$D_{i,y}^j = \frac{(G_{i,y} - G_{j,y})}{\sqrt{(G_{i,x} - G_{j,x})^2 + (G_{i,y} - G_{j,y})^2}} \quad . \quad (14)$$

Here, $f(\mathbf{z}_i)$ is defined to be the following sigmoid functions:

$$f(\mathbf{z}_{i,x}) = \frac{1-e^{-\mathbf{z}_{i,x}}}{1+e^{-\mathbf{z}_{i,x}}}$$
$$f(\mathbf{z}_{i,y}) = \frac{1-e^{-\mathbf{z}_{i,y}}}{1+e^{-\mathbf{z}_{i,y}}} \quad (15)$$

where $\mathbf{z}_{i,x}$ and $\mathbf{z}_{i,y}$ are defined by:

$$\mathbf{z_{i,x}} = (\mathbf{G_{i,x}} - \mathbf{G_{i,x}(u)}),$$
$$\mathbf{z_{i,y}} = (\mathbf{G_{i,y}} - \mathbf{G_{i,y}(u)}), \quad (16)$$

where $G_{i,x}(u)$ and $G_{i,y}(u)$ are $x-$ and $y-$ coordinates of the target pattern represented by the NURBS model [53]. Here, u is a parameter in the NURBS model ranging from 0 to 1.

We have proved in our previous work [46] that the dynamics of the GRN in layer 2 converges to the target pattern, provided that certain constraints on the parameters in Equations (7)-(10) are satisfied. Since no target position is pre-defined for the robots, a random value for $u \in [0, 1]$ will be chosen by each robot. As a result, it is possible that more than one robot goes to the same point on the target shape. However, due to the diffusion term in Eqns. (9) and (10), the robots will adjust their dynamics automatically to keep a certain distance from each other so that the robots distribute on the target shape evenly. These diffusion terms also contribute to obstacle avoidance. Once a robot detects an obstacle in its neighborhood, the diffusion terms will drive the robots away from the obstacle. Here, protein type G can diffuse into other cells and influence the motion dynamics of other robots. Meanwhile, protein type P receives diffusion from neighboring robots. The diffusion process of protein type G and P corresponds to the inter-robot communication to avoid collision when the robots converge to the target patterns.

Fig. 3 provides a diagram of the GRN dynamics in layer 2. From this diagram, it can be seen that the dynamics of the GRN in layer 2 is actually a position control system with multiple coupled feed-forward and feedback loops. These regulatory loops work together to ensure robust and accurate position tracking control.

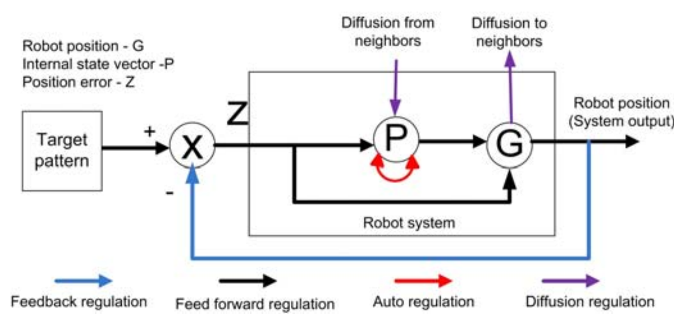


Fig. 3. A diagram of the GRN dynamics of layer 2.

C. Following Dynamics

For those robots that do not detect any target, their movement behavior is governed by the following dynamics. Assume that a non-organizing robot has N neighbors, which are within its sensing range. The dynamics of this robot is determined by:

$$\frac{dx}{dt} = \sum_{j=1}^{N} \left(\frac{dx_j}{dt} - \frac{dx}{dt} \right), \quad (17)$$

where x denotes the current position of the non-organizing robot. $\frac{dx}{dt}$ is the velocity of the robot. Once a following robot detects a target, it will become an organizing robot and its GRN dynamics in layer 1 will be activated.

To summarize, the movement dynamics of organizing robots is governed by the GRN dynamics in layer 2, driving the robots to the target pattern generated by the GRN of layer 1, whereas the movement dynamics of following robots is governed by the following dynamics in Equation (17). Both dynamics are run online in the robots and no pre-defined map is needed in the pattern formation process.

V. Evolutionary Optimization of the H-GRN

A. Optimization of the Pattern Generation Layer

Several parameters, including θ_1, θ_2, θ_3 and k in Layer 1 (Eqns. (1)-(5)) need to be specified. In this work, we adopt the covariance matrix adaptation evolution strategy (CMA-ES) [25] to optimize these parameters, as CMA-ES has shown to be efficient for continuous optimization even with a small population size. For evolutionary optimization of the H-GRN parameters, we first need to define a fitness function for pattern generation. Since we are considering pattern generation for entrapping multiple targets, the formulated pattern, to which the robots will converge, should neither be too far away from (may not be able to trap the targets), nor too close to the targets (the targets may pose danger to the robots if they are too close to the targets). Therefore, the fitness function is defined as follows:

$$\text{fitness} = 1 - \text{sig}(d_{\min}, d_0, k) + \text{sig}(d_{\max}, d_0 + 1, k), \quad (18)$$

where d_0 is set to 1, sig() is a sigmoid function as defined in Eqn.(6). $d_{\min}$ and $d_{\max}$ are the allowed minimum and maximum distances between the pattern and the targets, respectively, where we define $d_{\min} > d_0$ and $d_{\max} < d_0 + 1$. In other words, the distance between the generated pattern and the targets should be between d_0 and $d_0 + 1$. Ideally, $d_{\min}$ and $d_{\max}$ are the minimal and maximum distances from the generated pattern to the entrapping targets, respectively. However, since the generated target pattern is a continuous shape, we have to discretize the pattern to a few reference points. Therefore, $d_{\min}$ is calculated as the minimal distance between all targets (at discrete positions) and the reference points of the generated target pattern. The similar method is applied to calculate $d_{\max}$. The above fitness function is to be minimized.

In evolutionary optimization, θ_1 and θ_2 are initialized randomly between 0 to 1. The mean value of θ_3 is initialized between 1 to 2. The mean value of k is initialized between 1 to 100. The evolutionary process is terminated when the sum of fitness changes within the most recent 20 generations is smaller than 0.005 or the number of generations exceeds 50. The population size is set to 100. These parameters are chosen empirically. The optimized parameters are: $\theta_1 = 0.271$, $\theta_2 = 0.326$, $\theta_3 = 1.672$ and $k = 81.338$. γ is set to $\sqrt{\pi/2}$.

B. Optimization of the Pattern Formation Layer

The robots are expected to quickly converge to the target pattern generated by the GRN in layer 1 with a minimum travel distance. Therefore, optimization of the parameters of the GRN in layer 2 has two objectives, namely, to minimize the robots' travel distance and to minimize the convergence time.

This is a multi-objective optimization problem and mathematically, the problem can be formulated as: $\min_{x}[f_1(\mathbf{x}), f_2(\mathbf{x})]$, where $f_i(\mathbf{x})$ $(i = 1, 2)$ is the i-th objective function, $\mathbf{x}$ is a vector of the meta-parameter combination (a, m, c, r, b) in the GRN in layer 2. The two objective functions are:

$$f_1(\mathbf{x}) = \sum_{i=1}^{n} td_i(\mathbf{x}), \quad (19)$$

$$f_2(\mathbf{x}) = \max_{i=1\dots n} ct_i(\mathbf{x}), \quad (20)$$

where n is the total number of robots in the system. $td_i(\mathbf{x})$ is the travel distance of the i-th robot given the meta-parameter combination $\mathbf{x}$, $ct_i(\mathbf{x})$ is the convergence time for the i-th robot given the meta-parameter combination $\mathbf{x}$. Note that $td_i(\mathbf{x})$ and $ct_i(\mathbf{x})$ need to be measured during the simulation.

We employ NSGA-II [11], a popular evolutionary multi-objective optimization algorithm, to optimize the GRN parameters in layer 2 to minimize the two objectives. The population size of NSGA-II is set to 100. The crossover probability is set to 0.9 and the distribution index for the SBX crossover is 20. Mutation probability is defined to be inversely proportional to the number of the decision variables. As the GRN in layer 2 has five parameters, the probability is set to be 0.2 and the distribution index for mutation is set to be 20. The distribution index for crossover and mutation is determined as recommended in [11]. The simulation is run for 50 generations. Initially, r, c, a, and m are assigned to a random number ranging from 1 to 100. To avoid collision between robots, we assign b to be a random number ranging from 200 to 1000.

TABLE I

THE PARETO-OPTIMAL SOLUTIONS (TD REFERS TO TRAVEL DISTANCE IN METERS AND CT REFERS TO THE CONVERGENCE TIME IN SECONDS)

Solution	r	c	b	a	m	td (m)	ct(s)
(a)	45.88	69.13	387.5	69.28	63.77	19.64	0.39
(b)	79.46	70.06	420.8	44.39	6.85	19.77	0.25
(c)	91.81	57.65	599.9	1.04	1.00	20.5	0.09

Three representative Pareto-optimal solutions achieved by NSGA-II are listed in Table I, where the unit for convergence time is second and the unit for travel distance is meter. Without loss of generality, solution (b) is chosen for the following case studies, which has a good balance between the total travel distance and convergence time. The parameters of solution (b) are used in the following simulations unless otherwise specified.

C. Discussion

Note that the optimal parameter setup for the two GRN models used in layer 1 and layer 2 depends on the environmental settings such as the number of robots, the number of targets and their locations. Therefore, we have tested 100 independent and different environmental settings with different numbers and locations of targets and different numbers of robots for evaluating the fitness of each individual (candidate solution). The mean fitness value averaged over the 100 independent tests is set to be the final fitness of the individual for selection. In this case, each individual's fitness value reflects its average performance over the various environmental settings.

After 30 generations for layer 1 and 50 generations for layer 2, we obtained the optimal parameter combination that gives the best performance on average against various environmental settings. Note that given a specific environmental setting, there might exist a better parameter combination. However, assigning different parameters to the system for different environmental settings is not practical because the environmental settings are unknown beforehand.

VI. Simulation Results and Analysis

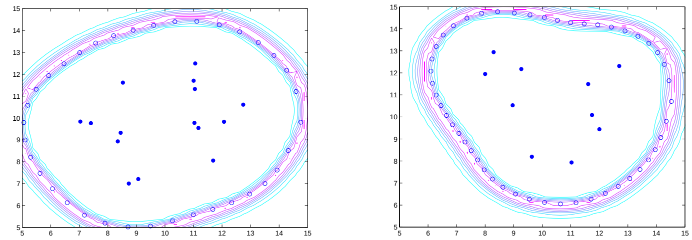
In this section, we will evaluate the proposed H-GRN model for adaptive pattern formation by performing a set of simulations. First, the robots are required to entrap multiple stationary targets to verify the feasibility of the model. Then, we will evaluate the adaptability, scalability and robustness of the model. Finally, we will discuss the unique features of our model compared to the state-of-the-art pattern formation algorithms.

A. Pattern Formation for Entrapping Stationary Targets

To examine the pattern formation ability of the proposed model, we test it in a scenario where multiple robots are used to entrap a few stationary targets in a region of 20 by 20 meters. The sensing range (r) of the robots is set to 3 meters. All GRN parameters are selected according to the optimization results obtained in Section V.

Once a pattern is generated by an organizing robot, the robot will move to the pattern governed by the GRN dynamics in layer 2. Six points sampled from the generated pattern are used to generate the NURBS representation of the target pattern. This pattern will be read in by the GRN of layer 2. The starting and end control points are the left-most (minimal x-position) and right-most (maximal x-position) points where the g_3 concentration is larger than 0.8. The other four control points are equally distant in x-axis between the starting and end control points.

Fig. 4 shows two examples of entrapping stationary robots, where 36 and 49 robots are used to entrap 15, 10 targets, respectively. From these results, we can see that the robots correctly move to the target pattern generated by the GRN of layer 1 under the governance of the dynamics of the GRN of layer 2, which takes the concentration of g_3, the output of layer 1 GRN, as the input. To quantitatively evaluate the pattern formation performance, we performed a set of simulations to examine the average position error, which is defined to be the average of the minimal distance between the robots and the target pattern. Note that no target position on the target shape is predefined for any robot. The mean and standard deviation of the position errors in various setups are listed in Table II, which are averaged over 25 independent runs. From the table, we can conclude that the robots can move onto the target pattern accurately in all these setups.



(a) (b)

Fig. 4. Two examples of entrapping stationary targets with multiple robots using the H-GRN, where the targets are denoted by a dot and the robots by a circle. The contours denote the concentration distribution of protein g_3. (a) 36 robots entrap 15 targets, (b) 49 robots entrap 10 targets.

TABLE II
Average position error (Mean $\pm$ Std) (m)

No. of Targets	No. of Robots 25	No. of Robots 36	No. of Robots 49
5	0.022$\pm$0.015	0.021$\pm$0.015	0.038$\pm$0.008
10	0.029$\pm$0.012	0.029$\pm$0.012	0.036$\pm$0.009
15	0.034$\pm$0.008	0.032$\pm$0.010	0.035$\pm$0.010

B. Adaptability to Environmental Changes

To demonstrate the adaptation ability of the proposed model, we have designed a test scenario where the targets move in the simulated area in such a way that the desired target pattern must change in both the shape and the number of the patterns. In the considered scenario, two targets move together at first. After some time, the two targets move apart toward two different directions. In this case, two patterns need to be constructed to entrap both targets. Then, the two targets move close to each other again and the target pattern should merge too. Snapshots of the pattern generation process during in the above scenario are provided in Fig. 5, showing the ability of the model to adapt its generated pattern to environmental changes. Note that splitting or merging of the patterns fully resulted from the dynamics of the GRN of layer 1; the robots only need to detect the locations of the targets and update the input (γ_j) of the GRN dynamics of layer 1.

C. Scalability of the H-GRN Model

We examine the scalability of the proposed model in terms of the time needed for all robots to converge to the target pattern. The time to convergence includes the time needed for deploying the robots from the base to cover the whole area, the time for the organizing robots to generate a stable target pattern, and the time for all robots to converge to the target pattern. In the following simulations, we assume that the deployment is complete if the velocity of all robots is lower than 1% of the maximum velocity during the employment. A pattern is regarded as stable when the concentration change is smaller than 1% of the maximum concentration change. In the pattern formation stage, a robot is considered to have converged to the target pattern once the position error (distance

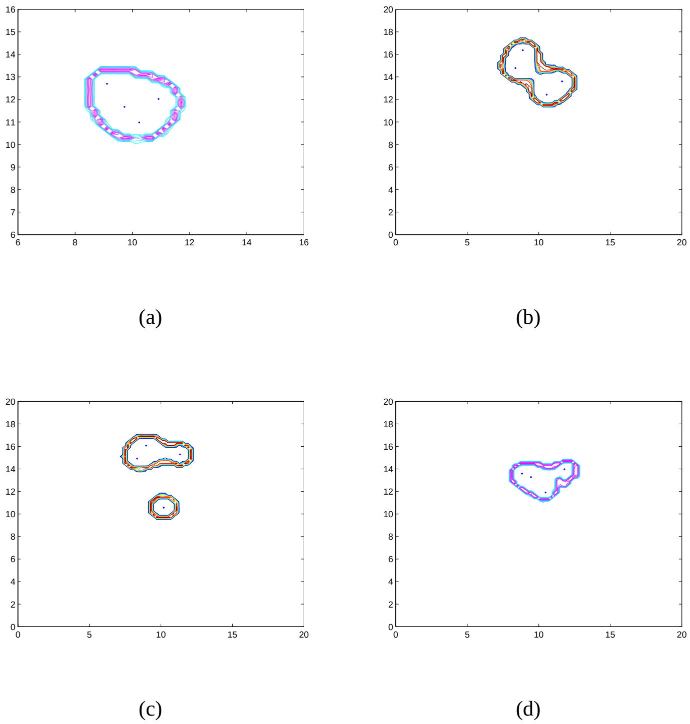


Fig. 5. Entrapping moving targets. (a)Eight robots have entrapped four targets. (b) The targets changed their relative position but remain close. The target pattern is adapted accordingly. (c) The targets move away, and therefore, two patterns must be formed to entrap them. (d) The targets move together again and one target pattern is formed to entrap the targets.

between the robots and a point on the target pattern closest to the robot) is smaller than 0.05, which is 5% of the allowed minimum distance between two robots entrapping targets. These criteria are empirically defined but a slight change in these criteria will not influence our conclusion.

In the simulations, the area to be covered is 20 by 20 meters, and the targets are randomly put in a 6 by 6 meters grid in the center of the area. All robots are initialized in a base of 5 by 5 meters in bottom-left corner of the area to be covered. In the simulations, we have considered nine cases where the number of targets are 5, 10 and 15, and the number of robots are 25, 36 and 49, respectively.

The processing time needed for each of the three stages are listed in Table III, Table IV, and Table V, respectively. From the tables, we can see that the convergence time is fairly insensitive to the number of robots and the number of targets. Note, however that the time for robot-robot communication has not been taken into account in these tables. One assumption we made in Section II is that the time for communications should be much smaller than that for the robots to converge to the target pattern. In real robotic system, the time for convergence may be dominating if we take the movement speed into account, which is scalable to the number of targets and number of robots.

TABLE III

TIME FOR DEPLOYMENT (MEAN $\pm$ STD) (S)

No. of Targets	No. of Robots		
	25	36	49
5	8.891$\pm$1.063	9.046$\pm$0.965	9.297$\pm$1.197
10	8.936$\pm$1.539	9.810$\pm$1.436	10.700$\pm$1.259
15	9.589$\pm$1.389	10.193$\pm$1.071	10.533$\pm$1.537

TABLE IV

TIME FOR PATTERN GENERATION (MEAN $\pm$ STD) (S)

No. of Targets	No. of Robots		
	25	36	49
5	8.326$\pm$0.343	8.314$\pm$0.304	8.383$\pm$0.263
10	8.336$\pm$0.234	8.352$\pm$0.309	8.375$\pm$0.205
15	8.394$\pm$0.250	8.415$\pm$0.199	8.352$\pm$0.199

TABLE V

TIME FOR THE ROBOTS TO CONVERGE TO THE TARGET PATTERN (MEAN $\pm$ STD) (S)

No. of Targets	No. of Robots		
	25	36	49
5	1.198$\pm$0.031	1.203$\pm$0.025	1.203$\pm$0.031
10	1.192$\pm$0.029	1.191$\pm$0.025	1.200$\pm$0.031
15	1.195$\pm$0.030	1.202$\pm$0.028	1.202$\pm$0.031

D. Robustness to Robot Failures

We have also shown in our previous work [30] that a GRN-based mechanism for multi-robot shape construction, which performs the functionality of the GRN in layer 2 of the proposed H-GRN in this work, is fairly insensitive to noise in measurements, and robust to individual robot failures. To verify the robustness of the H-GRN model proposed in this work, we consider a situation in which 16 robots have converged to a target pattern. We then assume that eight of the 16 robots become defective, see Fig. 6(a)-(b). The remaining robots will first adjust their neighborhood size and autonomously re-arrange their positions, trying to cover the gaps left by the defective robots, as shown in Fig. 6(c). Since the remaining eight robots are insufficient to cover the whole pattern (recall that there is a required minimum distance between two neighboring robots to cover the generated pattern), the two robots in Fig. 6 (c) that fail to find one of their neighbors will call the base station for additional robots. Consequently, two robots will be sent from the base station. These two robots, after detecting the targets, will activate the GRN of layer 1 and generate the target pattern, as other organizing robots have done. This way, they will converge to the target pattern guided by the dynamics of the GRN of layer 2. However, after rearrangements, it is found that ten robots are still insufficient to entrap the targets. Consequently, two additional robots are called from the base and finally the entrapping condition is satisfied with 12 robots, refer to Fig. 6 (c)-(f).

E. Discussions

In this section, we have shown that the proposed H-GRN model can generate different yet suitable patterns to entrap different target configurations in various simulations. When

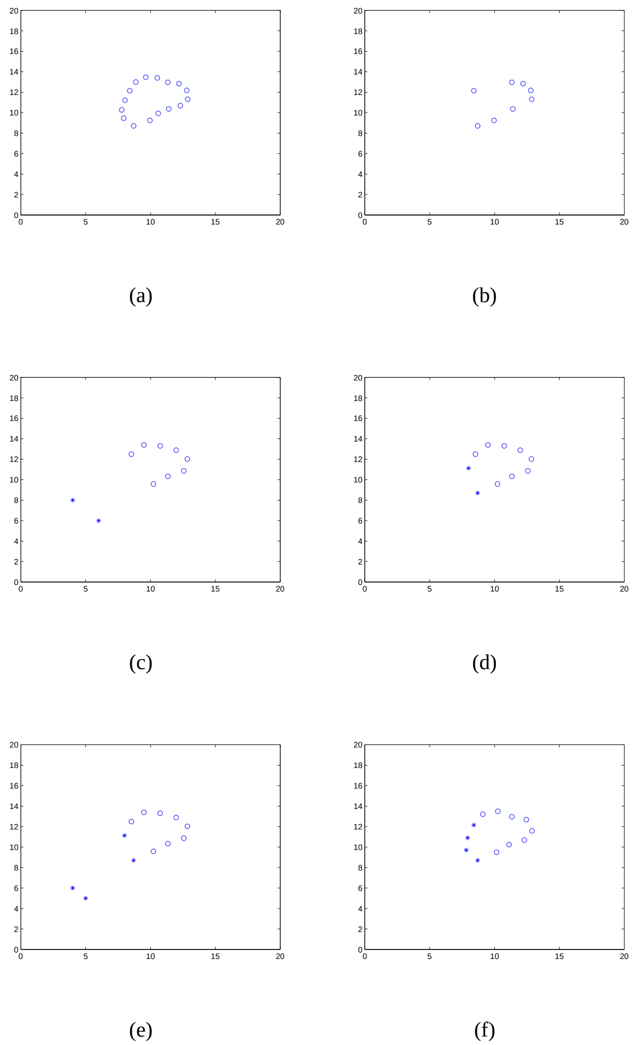


Fig. 6. Robustness to robot failures. (a) 16 robots cover the target pattern. (b) Eight of them become defective (disappeared). (c) The robots autonomously adjust their position, yet two robots fail to find one of their two neighbors. Thus, two robots are sent from the base. (d) The two new robots attempt to fit into the team, coming to recognize that there is still a hole. (e) Two additional robots are called. (f) After autonomous rearrangement of the positions of all robots, four new robots replace the eight defective ones to entrap the targets.

the targets are moving, the model can adaptively generate a dynamic pattern to entrap them. We have also demonstrated the robustness of the system to robot failures, particularly when the number of functioning robots is still large enough. Otherwise, we assume that the system is able to call for additional robots in the base station. This is based on the assumption that the need for additional robots can be communicated to the base station. If this assumption is violated, no self-organization mechanisms will be able to resolve this problem. Finally, we demonstrate that the performance of the system is scalable to different numbers of robots and numbers of targets.

Compared to existing multi-robot pattern formation algorithms, one major advantage of our approach is that it provides an adaptive pattern generation mechanism that can dynamically generate an appropriate pattern to adapt to environmental changes. For example, in the entrapping tasks, the pattern has to be dynamically changed due to the constant movement of targets. Most existing multi-robot systems for pattern formation rely on a predefined pattern, which is not applicable to changing environments.

VII. Experimental Results

A. Experimental Setup

We aim to test whether the e-puck robots can adapt the formed pattern when the environment changes. More specifically, eight robots are used to entrap two targets. The targets are stationary, however, the position of one target is changed during the experiment. Consequently, the eight robots should adapt the formed pattern accordingly to entrap the two targets.

In this experiment, we provide each robot with its initial location and orientation. The robots can then localize themselves using their encoder. To reduce the accumulated localization errors, a scaled version of the UMBMark [4] calibration procedure was performed on all e-puck robots. Infrared proximity sensors are used for distance detection.

The evolutionary optimization was initially conducted in the simulated sysyem, where no speed limitation of the robots has been taken into account. Note, however that the maximum velocity of an e-puck robot is 13 cm/s, which prevents us from directly applying the optimized parameters in the simulation to the model for the experiments using e-puck robots. Therefore, we have fine tuned the parameters combination around the optimized setup found in simulation and obtained the following parameter setups for the e-puck based experimental system. The parameters of the GRN in layer 1 are as follows: $\theta_1 = 0.30$, $\theta_2 = 0.4$, $\theta_3 = 1.2$, and $k = 20$. The parameters of the GRN in layer 2 are set up as follows: $a = 62.6$, $m = 63.9$, $c = 70.35$, $r = 45.5$, and $b = 380$.

In this experimental setup, the two targets (two robots covered by a yellow sticker) existing in the concerned area can be detected by all robots, refer to Fig. 7(a). Therefore, a target pattern can be generated by each robot without communication based on the position of the targets. Snapshot showing the process in which the robots entrap the two targets are provided in Fig. 7(c)-(d). To demonstrate the adaptation ability of the proposed model, one of the target changes its position after the robots converge to the first target pattern. A few snapshots illustrating the adaptation of the formed pattern are given in Fig. 8. From these results, we can see that the robots are able to adapt their pattern based on the given locations of the targets without predefining the desired patterns.

B. Discussions

We have demonstrated experimentally the effectiveness and feasibility of the proposed H-GRN model for adaptive multi-robot pattern formation using e-puck robots. From the experimental results, we can see that the robots are able to encircle the two targets without a centralized control. Moreover, the robots are able to autonomously construct a new shape after one of the targets changes its position and moves out of the previously formed shape. This illustrates that the basic principle of the models works properly.

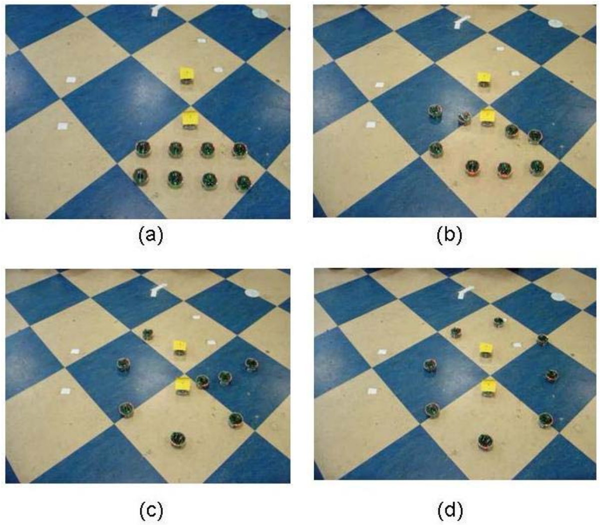


Fig. 7. Entrapping two targets with eight e-puck robots.

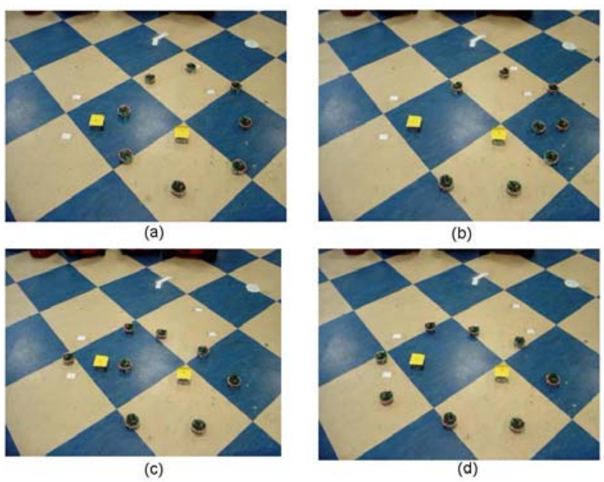


Fig. 8. Adaptation of the formed pattern to the changed position of the targets.

VIII. Related Work

Entrapping multiple targets discussed in this work is closely related to multi-robot target tracking, where multiple robots are used to track the positions of single or multiple targets [35], [39]. Algorithms for multi-robot target tracking can be divided into two groups. The first group is region-based approach [34], [35], in which the robots coordinate with each other to cover a certain region. This way, all the targets in the whole region can be detected and tracked. The advantage of this type of algorithm is that the robots do not need to know the target distribution information. One implicit assumption here is that there is always a sufficient number of robots available to cover the whole region. To improve the coverage efficiency, a virtual region based on the latest tracking information from the neighboring robots will be constructed [35]. Since the virtual region is a sub-area of the whole region, the coverage becomes more efficient.

The second group of multi-robot target tracking algorithms is target-oriented approach [22], [36], [37]. In contrast to the region-based algorithm, the robots will continuously update the number and location of targets (but they don't have to form patterns to entrap the targets). As a result, a large number of targets can still be tracked without covering the whole area. The target tracking rate [52] is often used to measure the performance of such algorithms.

Entrapping targets by forming multi-robot patterns also appears similar to membrane formation [14], which can be realized with a simple heuristic function. However, we should point out that the GRN dynamics for adaptive pattern formation is far more than membrane pattern generation. As shown in the simulation results in Section VI-B, when the targets move far away from each other, the dynamics of GRN of layer 1 can automatically form two separate patterns, which is not achievable by a heuristic membrane function. To the best of our knowledge, no research work has been reported on using a membrane function for multi-robot pattern generation. The dynamics of the GRN of layer 1 offer a solution to generating rather than predefining a pattern without the need of manually designing a membrane formation function. Taylor et al. [64] have proposed a GRN-inspired real-time controller for a group of robots. However, the two-layer GRN-based model was mainly applied to a 2-class clustering task without dedicated controllers for adaptive pattern generation. Other researchers have used various mathematical tools to generate shapes [21], [69], which were not investigated for multi-robot shape formation.

IX. Conclusion and Future Work

In this paper, a bio-inspired H-GRN model is presented for adaptive multi-robot pattern formation. The main new feature of the proposed model is that the target pattern generated by the robots needs not to be pre-defined and is adaptable to environmental changes, e.g., the number and location of the targets to be entrapped. The H-GRN is composed of two layers of GRN, where the GRN in layer 1 generates the target pattern online based on the position of the detected targets. If needed, more than one pattern can be automatically generated at the same time. This ability of adaptive pattern generation is not available in the GRN models we reported before, nor in any existing models for distributed multi-robot pattern formation, to the best of our knowledge. The proposed model enables a distributed self-organization of multi-robot systems and exhibits strong robustness to robot failures. Empirical results show that the performance of the system in terms of the time needed for constructing the target pattern is fairly scalable to the number of robots and targets due to the distributed nature of the whole system. A proof-of-concept experiment has also been performed successfully to demonstrate the basic ability of the proposed framework using eight e-puck robots.

However, it should be pointed out that successful entrapping of the mobile targets is conditioned on the assumption that the movement speed of the robots is faster than that of the targets and that the time for communication is much shorter than that for pattern generation and for the robots to converge to the target pattern. In the future, we would like to investigate in detail the conditions under which the whole system is able to keep encircling the moving targets.

A few quite conservative assumptions are still made in this model, including the reliance on a global coordinate system. A more realistic implementation is to build up a local coordinate system through limited robot-robot communications. In addition, the position of all the detected targets are used for generating the target pattern, which may cause computational problems if the number of targets is huge. A computationally more efficient approach is to build up the target pattern incrementally. As a result, only the position of a few targets, which determine the shape of the pattern, will be needed in pattern generation. In the experiment using the e-puck robots, the targets are basically stationary except for one change in position of one target. More practical entrapping situations, e.g., entrapping moving targets, will be investigated in future experiments.

Entrapping targets can be seen as a generic application of multi-robot systems. The H-GRN proposed in this work can be conceivably extended to other applications such as anti-missile systems. When a multi-agent anti-missile system is launched, if the attacking missiles have different trajectories or split into smaller independent vehicles, the anti-missiles can generate an adaptive pattern to entrap all the attacking missiles. A potential application of this model is for cordoning off hazardous materials. When the distribution of the hazardous materials is detected, the H-GRN model can generate a suitable shape to encircle those detected hazardous materials and prevent people from moving into the dangerous area. Another example is to measure the size of unknown objects using multiple robots.

ACKNOWLEDGMENT

The authors are grateful to the Editors and anonymous reviewers for their insightful comments that have significantly improved the quality of the paper. The authors would like to thank Xinglong Ju for implementing the algorithms on the e-puck robots.

REFERENCES

[1] S. Basu, Y. Gerchman, C. H. Collins, F. H. Arnold, and R. Weiss, "A synthetic multicellular system for programmed pattern formation," *Nature*, vol. 434, pp. 1130–1134, 2005.

[2] J. L. Baxter, E. K. Burke, J. M. Garibaldi, and M. Norman, "Multirobot search and rescue: A potential field based approach," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 9–16.

[3] R. Bloom, C. Chang, and A. Kondacs, "Compilation and biologically inspired self-assembly of two-dimensional shapes," in *International Joint Conference on Artificial Intelligence*, 2003, pp. 633–638.

[4] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 6, pp. 869–880, 1996.

[5] T. Bouwmeester, "The Spemann-Mangold organizer: the control of the fate specification and morphogenetic rerrangements during gastrulation in *xenopus*," *Int. Journal of Developmental Biology*, vol. 45, pp. 251–258, 2001.

[6] Y. Chen and Y. Tian, "A backstepping design for directed formation control of three-coleader agents in the plane," *International Journal of Robust and Nonlinear Control*, vol. 19, no. 7, pp. 729–745, 2009.

[7] J. Cheng, W. Cheng, and R. Nagpal, "Robust and self-repairing formation control for swarms of mobile agents," in *Proceedings of the National Conference on Artificial Intelligence*, 2005, pp. 59–64.

[8] S. Ciliberti, O. Martin, and A. Wagner, "Innovation and robustness in complex regulatory gene networks," *Proceedings of the National Academy of Sciences*, vol. 104, no. 34, pp. 13 591–13 596, 2007.

[9] D. Clyde, M. Corado, X. Wu, A. Pare, D. Papatsenko, and S. Small, "A self-organizing system of repressor gradients establishes segmental complexity in drosophila," *Nature*, pp. 849–853, 2003.

[10] H. De Jong, "Modeling and simulation of genetic regulatory systems: A literature review," *Journal of Computational Biology*, vol. 9, no. 1, pp. 67–103, 2002.

[11] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[12] L. Diambra and L. da Fontoura Costa, "Pattern formation in a gene network model with boundary shape dependence," *Physical Review E*, vol. 73, no. 031917, 2006.

[13] T. Dierks and S. Jagannathan, "Neural network output feedback control of robot formations," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 2, pp. 383–399, 2010.

[14] K. E. K. Douglas R. Lloyda and H. Tsengb, "Microporous membrane formation via thermally induced phase separation," *Journal of Membrane Science*, vol. 52, no. 3, pp. 239–261, 1990.

[15] P. Eggenberger Hotz, "Combining developmental processes and their physics in an artificial evolutionary system to evolve shapes," in *On Growth, Form, and Computers*, 2003, pp. 301–318.

[16] S. Ekanayake and P. N. Pathirana, "Geometric formations in swarm aggregation: An artificial formation force based approach," in *Proceedings of the 3rd International Conference on Information and Automation for Sustainability*, 2007, pp. 82–87.

[17] D. Erwin and E. Davidson, "The evolution of hierarchical gene regulatory networks," *Nature Reviews Genetics*, vol. 10, pp. 141–148, 2009.

[18] Y. Fomekong-Nanfack, J. Kaandorp, and J. Blom, "Efficient parameter estimation for spatio-temporal models of pattern formation: Case study of drosophila melanogaster," *Bioinformatics*, vol. 23, no. 24, pp. 3356–3363, 2007.

[19] P. Francois and V. Hakim, "Design of genetic networks with specified functions by evolution *in silico*," *Proceedings of the National Academy of Sciences of USA*, vol. 101, no. 2, pp. 580–585, 2007.

[20] S. Gilbert, *Developmental Biology*. Sinauer Associates, 2003.

[21] J. B. Greer, A. L. Bertozzi, and G. Sapiro, "Fourth order partial differential equations on general geometries. ucla computational and applied mathematics reports," in *University of California Los Angeles*, 2005, p. 2006.

[22] D. Gu, "A game theory approach to target tracking in sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 1, pp. 2–13, 2011.

[23] H. Guo, Y. Jin, and Y. Meng, "A unified framework for self-organized multi-robot pattern formation and boundary coverage inspired from morphogenesis," *ACM Transactions on Adaptive and Autonomous Systems*, 2010, in press.

[24] H. Guo, Y. Meng, and Y. Jin, "A cellular mechanism for multi-robot construction via evolutionary multi-objective optimization of a gene regulatory network," *BioSystems*, vol. 98, no. 3, pp. 193–203, 2009.

[25] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.

[26] S. Harding and W. Banzhaf, "Artificial development," in *Organic Computing*, R. P. Würtz, Ed. Springer, Mar. 2008, pp. 201–220. [Online]. Available: http://www.springer.com/978-3-540-77656-7

[27] Howard, M. Mataric, and S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problema," in *6th International Symposium on Distributed Autonomous Robotics Systems*, 2002, pp. 299–308.

[28] M. Hsieh, V.Kumar, and L.Chaimowicz, "Decentralized controllers for shape generation with robotic swarms," *Robotica*, vol. 26, no. 5, pp. 691–701, 2008.

[29] J. Jaeger, S. Surkova, M. Blagov, H. Janssens, D. Kosman, K. Kozlov, E. Myasnikova, C. Vanario-Alonso, M. Samsonova, D. Sharp, and J. Reinitz, "Dynamic control of positional information in the early drosophila embryo," *Nature*, 2004.

[30] Y. Jin, H. Guo, and Y. Meng, "Robustness analysis and failure recovery for a bio-inspired self-organizing multi-robot system," in *Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, 2009, pp. 154–164.

[31] Y. Jin and Y. Meng, "Emergence of robust regulatory motifs from in silico evolution of sustained oscillation," *BioSystems*, vol. 103, no. 1, pp. 38–44, 2011.

[32] ——, "Morphogenetic robotics: An emerging new field in developmental roboticsmorphogenetic robotics: An emerging new field in developmental robotics," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, no. 2, pp. 145–160, 2011.

[33] Y. Jin and J. Trommler, “A fitness-independent evolvability measure for evolutionary developmental systems,” in *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, 2010, pp. 69–76.

[34] B. Jung and G. Sukhatme, “Tracking targets using multiple robots: The effect of environment occlusion,” *Autonomous Robots*, vol. 13, pp. 191–205, 2002.

[35] ——, “A generalized region-based approach for multi-target tracking in outdoor environments,” in *IEEE International Conference on Robotics and Automation*, vol. 3, 2004, pp. 2189–2195.

[36] ——, “Cooperative multi-robot target tracking,” in *Distributed Autonomous Robotic Systems 7*, 2006, pp. 81–90.

[37] S. Kamath, E. Meisner, and V. Isler, “Triangulation based multi target tracking with mobile sensor networks,” in *IEEE International Conference on Robotics and Automation*, 2007, pp. 3283 –3288.

[38] K. Kanjanawanishkul and A. Zell, “A model-predictive approach to formation control of omnidirectional mobile robots,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems ,IROS 2008*, 2008, pp. 2771–2776.

[39] K. LeBlanc and A. Saffiotti, “Multirobot object localization: A fuzzy fusion approach,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 5, pp. 1259–1276, 2009.

[40] F. Z. M. Mamei, M. Vasirani, “Experiments in morphogenesis in swarms of simple mobile robots,” *Applied Artificial Intelligence*, vol. 18, no. 9-10, pp. 903–919, 2004.

[41] R. Madhavan, K. Fregene, and L. E. Parker, “Distributed heterogeneous outdoor multi-robot localization,” in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, 2002, pp. 374–381.

[42] A. Martinoli, K. Easton, and W. Agassounon, “Modeling swarm robotic systems: A case study in collaborative distributed manipulation,” *International Journal of Robotics Research*, vol. 23, no. 4, pp. 415–436, 2004.

[43] A. Martinoli, A. J. Ijspeert, and F. Mondada, “Understanding collective aggregation mechanisms: From probabilistic modeling to experiments with real robots,” *Robotics and Autonomous Systems*, vol. 23, no. 4, pp. 415–436, 1999.

[44] M. J. Mataric, M. Nilsson, and K. T. Simsarian, “Cooperative multi-robot box-pushing,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1995, pp. 556–561.

[45] Y. Meng and J. Gan, “LIVS: Local interaction via virtual stigmergy coordination in distributed search and collective cleanup,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 1371–1376.

[46] Y. Meng, H. Guo, and Y. Jin, “A morphogenetic approach to flexible and robust shape formation for swarm robotic systems,” *Robotics and Autonomous Systems*, 2010, submitted.

[47] Y. Meng and Y. Jin, “Distributed multi-agent systems for a collective construction task based on virtual swarm intelligence,” *Int. Journal of Swarm Intelligence Research*, vol. 1, no. 2, pp. 531–551, 2010.

[48] Y. Meng, J. Nickerson, and J. Gan, “Multi-robot aggregation strategies with limited communication,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robot and Systems*, 2006, pp. 2691–2696.

[49] Y. Meng, Y. Zhang, and Y. Jin, “A morphogenetic approach to self-reconfigurable modular robots using a hybrid hierarchical gene regulatory network,” in *12th International Conference on the Synthesis and Simulation of Living Systems (Artificial Life XII*, 2010.

[50] ——, “Autonomous self-reconfiguration of modular robots using a hierarchical mechanochemical model,” *IEEE Computational Intelligence Magazine*, vol. 6, no. 1, pp. 43–54, 2011.

[51] Z. Meng, W. Ren, Y. Cao, and Z. You, “Leaderless and leader-following consensus with communication and input delays under a directed network topology,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 1, pp. 75–88, 2011.

[52] F. M. Mirzaei, A. I. Mourikis, and S. I. Roumeliotis, “On the performance of multi-robot target tracking,” in *IEEE International Conference on Robotics and Automation*, 2007, pp. 3482–3489.

[53] L. Piegl and W. Tiller, *The NURBS Book*, 2nd ed. Springer, 1997.

[54] J. Pugh and A. Martinoli, “Distributed adaptation in multi-robot search using particle swarm optimization,” in *Proc. of the Int. Conf. on Simulation of Adaptive Behavior*, ser. LNCS 5040, 2008, pp. 393–402.

[55] T. Quick, C. Nehaniv, K. Dautenhahn, and G. Roberts, “Evolving embodied genetic regulatory network-driven control systems,” in *European Conference on Artificial Life*, 2003, pp. 266–277.

[56] S. Roumeliotis and G. Bekey, “Distributed multi-robot localization,” in *Proceedings of the Fifth International Symposium on Distributed Autonomous Robotic Systems*, 2000, pp. 241–250.

[57] I. Salazar-Ciudad, J. Jernvall, and S. Newman, “Mechanisms of pattern formation in development and evolution,” *Development*, vol. 130, pp. 2027–2037(, 2003.

[58] H. Sayama, “Robust morphogenesis of robotic systems by simple kinetic interaction, local information transmission and stochastic differentiation,” *IEEE Computational Intelligence Magazine*, vol. 5, no. 3, pp. 43–49, 2010.

[59] W. Shen, P. Will, and A. Galstyan, “Hormone-inspired self-organization and distributed control of robotic swarms,” *Autonomous Robots*, 2004.

[60] K. O. Stanley and R. Miikkulainen, “A taxonomy for artificial embryogeny,” *Artificial Life*, pp. 93–130, 2003.

[61] T. Steiner, J. Trommler, M. Brenn, Y. Jin, and B. Sendhoff, “Global shape with morphogen gradients and motile polarized cells.” in *Congress on Evolutionary Computation*, 2009, pp. 2225–2232.

[62] I. Suzuki and M. Yamashita, “Distributed anonymous mobile robots: Formation of geometric patterns,” *SIAM Journal on Computing*, vol. 28, no. 4, pp. 1347–1363, 1999.

[63] T. Taylor, “A genetic regulatory network-inspired real-time controller for a group of underwater robots,” in *Intelligent Autonomous Systems*, vol. 8, 2004, pp. 403–412.

[64] T.Taylor, P.Ottery, and J.Hallam, “An approach to time- and space-differentiated pattern formation in multi-robot systems,” in *Towards Autonomous Robotic Systems (TAROS 2007)*, 2007, pp. 160–167.

[65] A. M. Turing, “The chemical basis of morphogenesis,” *R. Soc. Lond. Phil. Trans.*, vol. B, no. 237, pp. 37–72, 1952.

[66] Z. Wang and T. T. Y. Kimura, “A control method of a multiple non-holonomic robot system for cooperative object transportation,” in *Distributed Autonomous Robotic Systems*, L. Parker, G. Bekey, and J. Barhen, Eds. Springer, 2000, pp. 447–456.

[67] J. Werfel and R. Nagpal, “Extended stigmergy in collective construction,” *IEEE Intelligent Systems*, vol. 21, no. 2, pp. 20–28, 2006.

[68] D. Yamins and R. Nagpal, “Automated global-to-local programming in 1-d spatial multi-agent systems,” in *7th Int. Joint Conf on Autonomous Agents and Multiagent Systems*, vol. 2, 2008, pp. 615–622.

[69] C. T. Zahn and R. Z. Roskies, “Fourier descriptors for plane closed curves,” *IEEE Transactions on Computers*, vol. 21, pp. 269–281, 1972.

[70] S. Zhan, J. F. Miller, and A. M. Tyrrell, “An evolutionary system using development and artificial genetic regulatory networks for electronic circuit design,” *BioSystems*, vol. 98, no. 3, pp. 176–192, 2009.