The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Adaptive Flocking Control for Dynamic Target Tracking in Mobile Sensor Networks

Hung Manh La and Weihua Sheng

***Abstract*— Target tracking is an important task in sensor networks, especially in mobile sensor networks. Flocking control is used to control a mobile sensor network to track a target. However, there are some existing problems in this control method, such as network fragmentation, loss of formation and poor tracking performance. In order to handle these problems we propose a novel approach to flocking control of a mobile sensor network to track a moving target within changing environments. In our approach, each agent can cooperatively learn the network's parameters to decide the size of network in a decentralized fashion so that the connectivity, formation and tracking performance can be improved when avoiding obstacles. In addition, to demonstrate the benefit of our approach a comparison between this approach and the existing method is given. Computer simulations are performed to demonstrate the effectiveness of the proposed approach.**

Keywords: Flocking, target tracking, mobile sensor network.

I. Introduction

A. Motivation

Mobile sensor networks [1] have advantages over stationary sensor networks such as adaptation to environmental changes and reconfigurability for better performance. A main issue for multiple mobile sensors to track a moving target is that these sensors have to move together without collision among them during tracking. This requires us to apply cooperative control methods, and one of these methods is flocking control [2], [3], [4], [5]. However, these existing works have some limitations when the environment changes, for example when the mobile sensor network has to pass through a narrow space among obstacles. These limitations include:

1. Connectivity is lost because of the fragmentation phenomenon.
2. Formation of the network is totally changed.
3. Low speed or getting stuck causes poor tracking performance.

Therefore to design an adaptive flocking control algorithm to deal with these problems is a challenging task. In this paper, we present a novel approach to flocking control of a mobile sensor network to track a moving target in changing environments. In this approach, each agent cooperatively learns the network's parameters in a decentralized fashion so that the connectivity, formation and tracking performance can be improved when avoiding obstacles. The reason of maintaining the connectivity and similar formation is that when the network shrinks its size to deal with changing environments the neighborhood of each agent can be maintained. This allows the network to keep the same topology, which reduces the complexity of control during the tracking process. Computer simulations are conducted to prove our theoretical results.

B. Related work

Flocking control has received considerable attention due to its wide applications, such as space exploration and surveillance. In [2], the theoretical framework for design and analysis of distributed flocking algorithms was proposed. These algorithms solved the flocking in free space and in the presence of obstacles. The static and dynamic virtual leaders were used as a navigational feedback for all mobile sensors. This established a background for flocking control design for a group of sensors. Adaptive flocking control, an extension of flocking control, has also gained attention from researchers in recent years. Yang *et al.* [6] proposed an adaptive flocking control algorithm to avoid collision among robots themselves and between robots and obstacles. However, their algorithm did not consider the problem of formation, connectivity and tracking performance in complex environments. In addition, their algorithm only considered a static target or a rendezvous point, which leads all agents to get there. Lee and Chong [7] introduced a motion planning framework for a large number of autonomous robots that enables the robots to configure themselves adaptively into an area of arbitrary geometry. Their proposed method allows the robots to converge to the uniform distribution by forming an equilateral triangle with their two neighbors. However, the problem of target tracking was not addressed in their work. An extension of their work was developed in [8] by the same authors to allow the swarm of robots to go to predetermined rendezvous points. Their approach was based on a decentralized approach that enables a swarm of robots navigate autonomously in complex environments populated by obstacles. The problem of splitting/merging mobile robots in the network according to the environments is addressed in their paper. Namely, when the swarm of robots detects obstacles, each robot splits from the network and determines its direction toward the static goal based on the width of space among obstacles. However, in reality it is difficult for each robot to sense the whole environment and compute the width of space among obstacles. Also, in their work

This project is supported by the Vietnamese Government, MOET (Ministry of Education and Training) program and the DoD ARO DURIP grant 55628-CS-RIP.

Hung Manh La and Weihua Sheng are with the school of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078, USA (hung.la@okstate.edu, weihua.sheng@okstate.edu).

the problem of controlling the size of the network was not considered, and the connectivity and formation were not guaranteed in complex environments.

In summary, most of existing work focused on the coordination, formation and splitting/merging problems in both fixed and switching topologies. The problem of how to control the size of the network in a decentralized and adaptive fashion in complex environments while maintaining connectivity, formation and tracking performance is still an open problem.

The rest of this paper is organized as follows. In the next section we present the background of flocking control. Section III describes our adaptive flocking control algorithm to track a moving target while avoiding obstacles. Section IV provides the simulation results. Finally, section V concludes this paper.

II. Flocking control background

In this section we will present the graph preliminary and the flocking control background. We consider n sensors moving in an m (e.g.,$m = 2,3$) dimensional Euclidean space. The dynamic equation of each sensor is described as follows:

$$\begin{cases} \dot{q}_i = p_i \\ \dot{p}_i = u_i, \quad i = 1,2,...,n. \end{cases} \tag{1}$$

To describe the topology of flocks or swarms we consider a dynamic graph G consisting of a vertex set $\vartheta = \{1,2...,n\}$ and an edge set $E \subseteq \{(i,j) : i,j \in \vartheta, j \neq i\}$. In this topology each vertex denotes one member of flocks, and each edge denotes the communication link between two members.

Let $q_i, p_i \in R^m$ be the position and velocity of node i, respectively. We know that during the movement of sensors, the relative distance between them may change, hence the neighbors of each sensor also change. Therefore, we can define a set of neighborhood of sensor i as follows:

$$N_i^\alpha = \left\{ j \in \vartheta : \|q_j - q_i\| \leq r,\ \vartheta = \{1,2,...,n\}, j \neq i \right\}, \tag{2}$$

here, r is an active range (radius of neighborhood circle in the case of two dimensions, $m = 2$, or radius of neighborhood sphere in the case of three dimensions, $m = 3$), and $\|.\|$ is the Euclidean distance.

The geometry of flocks is modeled by an α-lattice [2] that meets the following condition:

$$\|q_j - q_i\| = d, j \in N_i^\alpha, \tag{3}$$

here d is a positive constant indicating the distance between sensor i and its neighbor j.

To construct a collective potential that is differentiable at singular configuration ($q_i = q_j$), the set of algebraic constrains is rewritten in term of σ - norm as follows:

$$\|q_j - q_i\|_\sigma = d^\alpha, j \in N_i^\alpha, \tag{4}$$

here the constraint $d^\alpha = \|d\|_\sigma$ with $d = r/k_c$, where k_c is the scaling factor. The σ - norm [2], $\|.\|_\sigma$, of a vector is a map $R^m \Longrightarrow R_+$ defined as $\|z\|_\sigma = 1/\varepsilon[\sqrt{1+\varepsilon\|z\|^2} - 1]$ with $\varepsilon > 0$. Unlike the Euclidean norm $\|z\|$, which is not differentiable at $z = 0$, the σ - norm $\|z\|_\sigma$, is differentiable every where. This property allows to construct a smooth collective potential function for agents.

The flocking control law in [2] controls all sensors to form an α-lattice configuration. This algorithm consists of three components as follows:

$$u_i = f_i^\alpha + f_i^\beta + f_i^\gamma. \tag{5}$$

The first component of (5) f_i^α, which consists of a gradient-based component and a consensus component (more details about these components see [9], [10], [11]), is used to regulate the potentials (impulsive or attractive forces) and the velocity among sensors.

$$f_i^\alpha = c_1^\alpha \sum_{j \in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma) n_{ij} + c_2^\alpha \sum_{j \in N_i^\alpha} a_{ij}(q)(p_j - p_i), \tag{6}$$

where each term in (6) is computed as follows [2]: The action function $\phi_\alpha(z)$ that vanishes for all $z \geq r^\alpha$ with $r^\alpha = \|r\|_\sigma$ is defined as follows:

$$\phi_\alpha(z) = \rho_h(z/r_\alpha)\phi(z - d^\alpha) \tag{7}$$

with the uneven sigmoidal function $\phi(z)$ defined as $\phi(z) = 0.5[(a+b)\sigma_1(z+c) + (a-b)]$, here $\sigma_1(z) = z/\sqrt{1+z^2}$, and parameters $0 < a \leq b$, $c = |a-b|/\sqrt{4ab}$ to guarantee $\phi(0) = 0$. The bump function $\rho_h(z)$ with $h \in (0,1)$

$$\rho_h(z) = \begin{cases} 1, & z \in [0,h) \\ 0.5[1 + cos(\pi(\frac{z-h}{1-h}))], & z \in [h,1) \\ 0, & otherwise. \end{cases} \tag{8}$$

Vector along the line connecting q_i to q_j is defined as

$$n_{ij} = (q_j - q_i)/\sqrt{1+\varepsilon\|q_j - q_i\|^2}. \tag{9}$$

The adjacency matrix $a_{ij}(q)$ is defined as

$$a_{ij}(q) = \begin{cases} \rho_h(\|q_j - q_i\|_\sigma / r_\alpha), & if \quad j \neq i \\ 0, & if \quad j = i. \end{cases} \tag{10}$$

The second component of (5) f_i^β is used to control the sensors to avoid obstacles,

$$f_i^\beta = c_1^\beta \sum_{k \in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{n}_{i,k} + c_2^\beta \sum_{k \in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i), \tag{11}$$

where the set of β neighbors (virtual neighbors, [2]) is

$$N_i^\beta = \left\{ j \in \vartheta_\beta : \|\hat{q}_{i,k} - q_i\| \leq r', \vartheta_\beta = \{1,2,...,K\} \right\}, \tag{12}$$

here K is the number of obstacles, r' is an obstacle detecting range, and $\hat{q}_{i,k}, \hat{p}_{i,k}$ are the position and velocity of sensor i projected on the obstacle k, respectively (more details please see [2]).

Similar to vector n_{ij} established in (9), vector $\hat{n}_{i,k}$ is defined as

$$\hat{n}_{i,k} = (\hat{q}_{i,k} - q_i)/\sqrt{1+\varepsilon\|\hat{q}_{i,k} - q_i\|^2}. \tag{13}$$

The heterogeneous adjacent matrix $b_{i,k}(q)$ is defined as

$$b_{i,k}(q) = \rho_h(|\hat{q}_{i,k} - q_i\|_\sigma / d^\beta), \tag{14}$$

where $d^{\beta} = \|r'\|_{\sigma}$.

The repulsive action function of β neighbors is defined as

$$\phi_{\beta}(z) = \rho_h(z/d^{\beta})(\sigma_1(z - d^{\beta}) - 1). \quad (15)$$

The third component of (5) f_i^{γ} is a distributed navigational feedback.

$$f_i^{\gamma} = -c_1^{\gamma}(q_i - q_{\gamma}) - c_2^{\gamma}(p_i - p_{\gamma}), \quad (16)$$

where the γ - agent (q_{γ}, p_{γ}) is the virtual leader that leads the flock to follow its trajectory, and it is defined as follows

$$\begin{cases} \dot{q}_{\gamma} = p_{\gamma} \\ \dot{p}_{\gamma} = f_{\gamma}(q_{\gamma}, p_{\gamma}). \end{cases} \quad (17)$$

The constants of three components used in (5) are chosen as $c_1^{\alpha} < c_1^{\gamma} < c_1^{\beta}$, and $c_2^{\nu} = 2\sqrt{c_1^{\nu}}$. Here c_{η}^{ν} are positive constants for $\forall \eta = 1, 2$ and $\nu = \alpha, \beta, \gamma$.

III. Adaptive flocking control for tracking a moving target

In this paper, we consider the γ agent as a moving target. Hence, based on Olfati-Saber's flocking control [2] we design a control law with a moving target as

$$\begin{aligned} u_i \;=\; & c_1^{\alpha} \sum_{j \in N_i^{\alpha}} \phi_{\alpha}(\|q_j - q_i\|_{\sigma}) n_{ij} + c_2^{\alpha} \sum_{j \in N_i^{\alpha}} a_{ij}(q)(p_j - p_i) \\ & + c_1^{\beta} \sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}) \hat{n}_{i,k} + c_2^{\beta} \sum_{k \in N_i^{\beta}} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \\ & - c_1^{mt}(q_i - q_{mt}) - c_2^{mt}(p_i - p_{mt}), \end{aligned} \quad (18)$$

here $(q_{mt},\ p_{mt})$ is the position and velocity of the moving target, respectively, and c_1^{mt}, c_2^{mt} are positive constants. In this control law, we assume that each agent has ability to sense the position and velocity of the moving target.

The problem here is how to cooperatively control the size of the network in an adaptive and decentralized fashion in order to maintain the network's connectivity, similar formation and tracking performance in the presence of obstacles. One example of such flocking control is illustrated in Figure 1.

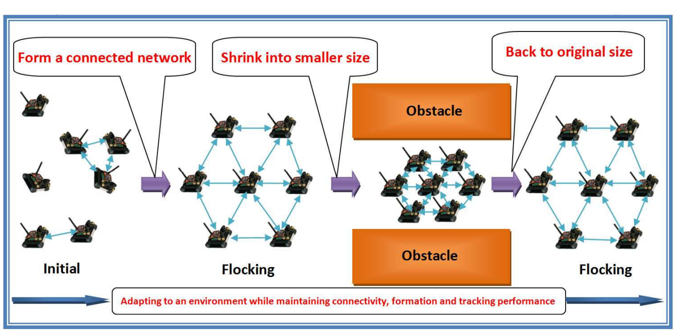

Fig. 1. Illustration of the adaptive flocking control.

A. Adaptive flocking control

To control the size of the network, we need to control the set of algebraic constraints in Equation (4), which means that if we want the size of the network to be smaller to pass the narrow space then d^{α} should be smaller. This raises the question of how small the size of network should be reduced and how to control the size in a decentralized and dynamic fashion.

To control the constraint d^{α} one possible method is based on the knowledge of obstacle obtained by any sensor in the network, which will broadcast a new d^{α} to all other sensors. However, it is difficult for a single sensor to learn the size of the obstacles due to its limited sensing range. To overcome this problem we propose a method based on the repulsive force, $\sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma})$, which is generated by the β-agent (virtual agent) projected on the obstacles. If any sensor in the network gets this repulsive force it will shrink its own d_i^{α}. If this repulsive force is big (sensor is close to obstacle(s)) d_i^{α} will be further reduced. Then, in order to maintain the neighborhood (topology) the active range of each sensor is re-designed. To create the agreement on the relative distance and active range among sensors in a decentralized way, a consensus or a local average update law is proposed. Furthermore, to maintain the connectivity each sensor is designed with an adaptive weight of attractive force from the target and an adaptive weight of interaction force from its neighbors so that the network reduces or recovers the size gradually. That is, if an sensor has weak connection to the network it should have a big weight of attraction force to the target and a small weight of interaction force from its neighbors.

Firstly, we control the set of algebraic constraints as

$$\|q_j - q_i\|_{\sigma} = d_i^{\alpha}, j \in N_i^{\alpha}, \quad (19)$$

and let each agent have its own d_i^{α}, which is designed as

$$d_i^{\alpha} = \begin{cases} d^{\alpha}, if \sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}) = 0 \\ \frac{c_a}{\sum_{k \in N_i^{\beta}} |\phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma})| + 1}, if \sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}) \neq 0, \end{cases} \quad (20)$$

here c_a is the positive constant.

From Equation (20) we see that if the repulsive force generated from the obstacles $\sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}) = 0$ or $N_i^{\beta} = \emptyset$ (empty set) then the agent will keep its original d^{α}. When the agent senses the obstacles it reduces its own d_i^{α}, and the value of d_i^{α} depends on the repulsive force that the agent gets from obstacles.

In order to control the size of network each sensor needs its own r_i^{α} that relates to d_i^{α} as follows: $r_i^{\alpha} = \|k_c d\|_{\sigma}$ with $\|d\|_{\sigma} = d_i^{\alpha}$ or $d = \sqrt{\frac{(\epsilon d_i^{\alpha}+1)^2 - 1}{\epsilon}}$. Explicitly, r_i^{α} is computed as in Equation (21).

$$r_i^{\alpha} = \begin{cases} r^{\alpha}, if \sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}) = 0 \\ \frac{1}{\epsilon}\left[\sqrt{k_c^2 \frac{(\epsilon d_i^{\alpha}+1)^2 - 1}{\epsilon} + 1} - 1\right], \\ if \sum_{k \in N_i^{\beta}} \phi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}) \neq 0. \end{cases} \quad (21)$$

Similar to computing r_i^α, r_i is computed as

$$r_i = \begin{cases} r, & if \quad \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma) = 0 \\ \sqrt{\frac{1}{\varepsilon}[(\varepsilon r_i^\alpha + 1)^2 - 1]}, & if \quad \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma) \neq 0. \end{cases} \tag{22}$$

It should be pointed out that the active range r_i is different from the physical communication (sensing) range. Namely, the active range is the range that each agent decides its neighbors to talk with, but the physical communication range is the range defined by the RF module. This implies that even a robot can communicate with many other robots in the network, it will only talk (interact) with robots in its active range. That is why we want to control the active range of each robot in order to reduce the communication and maintain the similar formation when the network shrinks.

To achieve agreement on d_i^α, r_i^α and r_i among sensors in the connected network we use the following update law based on local average for d_i^α, r_i^α and r_i:

$$\begin{cases} d_i^\alpha = \frac{1}{|N_i^\alpha|+1} \sum_{j=1}^{|N_i^\alpha|+1} d_j^\alpha \\ r_i^\alpha = \frac{1}{|N_i^\alpha|+1} \sum_{j=1}^{|N_i^\alpha|+1} r_j^\alpha \\ r_i = \frac{1}{|N_i^\alpha|+1} \sum_{j=1}^{|N_i^\alpha|+1} r_j, \end{cases} \tag{23}$$

here $|N_i^\alpha|$ is the number of neighbors of agent i.

In addition, to better maintain the network connectivity each agent should have an adaptive weight of attractive force from the target and interaction force from its neighbors as discussed before. Firstly, in the control protocol (18), the first two terms are used to control the formation (velocity matching, collision avoidance among robots). The third and fourth terms are used to allow robots to avoid obstacles, and the last term is used for target tracking. If the last term is absent the control will lead to network fragmentation [2]. The coefficients of the interaction forces (c_1^α, c_2^α), (c_1^β, c_2^β) and attractive force (c_1^{mt}, c_2^{mt}) which deliver desired swarm-like behaviour are used to adjust the weight of interaction forces and attractive force. The bigger (c_1^{mt}, c_2^{mt}) the faster convergence to the target. However if (c_1^{mt}, c_2^{mt}) is too big the center of mass (CoM) as defined in Equation (24)

$$\begin{cases} \overline{q} = \frac{1}{n}\sum_{i=1}^n q_i \\ \overline{p} = \frac{1}{n}\sum_{i=1}^n p_i \end{cases} \tag{24}$$

oscillates around the target, and the formation of network is not guaranteed. In addition, in order to guarantee that no agent hit obstacles, the pair (c_1^β, c_2^β) is selected to be bigger than the other two pairs, (c_1^α, c_2^α) and (c_1^{mt}, c_2^{mt}). Finally we have the relationship among these pairs as: ($c_{1,2}^\alpha < c_{1,2}^{mt} < c_{1,2}^\beta$).

From the above analysis we see that these adaptive weights allow the network to reduce and recover the size gradually. They also allow the network to maintain the connectivity during the obstacle avoidance. We let each sensor have its own weight of the interaction forces as in Equation (25) and attractive force as in Equation (26). In the α-lattice configuration if the sensor has less than 3 neighbors it is considered as having a weak connection to the network. This means that this sensor is on the border of network, or far from the target hence it should have bigger weight of attractive force from its target and smaller weight of interaction forces from its neighbors to get closer to the target. This design also has the benefit of making the whole network track the target faster. From this analysis $c_{1,2}^\alpha$ and $c_{1,2}^{mt}$ of each agent are designed as follows:

$$c_1^\alpha(i) = \begin{cases} c_1^\alpha, & if \quad |N_i^\alpha| \geq 3 \\ c_1^{\alpha'}, & if \quad |N_i^\alpha| < 3, \end{cases} \tag{25}$$

here $c_1^{\alpha'} < c_1^\alpha$, $c_2^\alpha(i) = 2\sqrt{c_1^\alpha(i)}$, and $i = 1, 2, ..., n$.

$$c_1^{mt}(i) = \begin{cases} c_1^{mt}, & if \quad |N_i^\alpha| \geq 3 \\ c_1^{mt'}, & if \quad |N_i^\alpha| < 3, \end{cases} \tag{26}$$

here $c_1^{mt'} > c_1^{mt}$, $c_2^{mt}(i) = 2\sqrt{c_1^{mt}(i)}$, and $i = 1, 2, ..., n$.

Now, the neighborhood of sensor i ($N_i^{'\alpha}$), the new adjacency matrix $a_{ij}(q)$ and the new action function $\phi_\alpha(z)$ are redefined as follows:

$$N_i^{'\alpha} = \{ j \in \vartheta : \|q_j - q_i\| \leq r_i, \ \vartheta = \{1, 2, ..., n\}, \ j \neq i\}; \tag{27}$$

$$a_{ij}^{'}(q) = \begin{cases} \rho_h(\|q_j - q_i\|_\sigma / r_i^\alpha), & if \quad j \neq i \\ 0, & if \quad j = i; \end{cases} \tag{28}$$

$$\phi_\alpha^{'}(\|q_j - q_i\|_\sigma) = \rho_h(\|q_j - q_i\|_\sigma / r_\alpha)\phi(\|q_j - q_i\|_\sigma - d_i^\alpha). \tag{29}$$

Finally, the adaptive flocking control law for dynamic target tracking is

$$\begin{aligned} u_i \quad = \quad & c_1^\alpha(i) \sum_{j\in N_i^{'\alpha}} \phi_\alpha^{'}(\|q_j - q_i\|_\sigma) n_{ij} \\ & + c_2^\alpha(i) \sum_{j\in N_i^{'\alpha}} a_{ij}^{'}(q)(p_j - p_i) \\ & + c_1^\beta \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{n}_{i,k} + c_2^\beta \sum_{k\in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \\ & - c_1^{mt}(i)(q_i - q_{mt}) - c_2^{mt}(i)(p_i - p_{mt}). \end{aligned} \tag{30}$$

B. Stability Analysis

By applying the control protocol (30), the CoM (defined in Equation (24)) of positions and velocities of all mobile sensors in the network will exponentially converge to the target in both free space and obstacle space. In addition, the formation (collision free and velocity matching among mobile sensors) will maintain in the process of the target tracking.

Let us consider adaptive flocking control in free space and obstacle space, respectively.

Case 1 (Free space): In free space, $\sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma) = 0$, hence we can rewrite the control protocol (30) by ignoring constants c_η^ν (for $\forall \eta = 1, 2$ and $\nu = \alpha, \beta$) as follows:

$$\begin{aligned} u_i \quad = \quad & -\sum_{j\in N_i^\alpha} \nabla_{q_i} \psi_\alpha(\|q_j - q_i\|_\sigma) + \sum_{j\in N_i^\alpha} a_{ij}(q)(p_j - p_i) \\ & - c_1^{mt}(q_i - q_{mt}) - c_2^{mt}(p_i - p_{mt}) \end{aligned} \tag{31}$$

where $\psi_\alpha(z) = \int_{d_\alpha}^{z} \phi_\alpha(s)ds$ is the pairwise attractive/repulsive potential function. From (31), we can compute the average of control law u as follows:

$$\begin{aligned}\overline{u} = \frac{1}{n}\sum_{i=1}^{n} u_i \quad &= \quad \frac{1}{n}\sum_{i=1}^{n}\Big(-\sum_{j\in N_i^\alpha} \nabla_{q_i}\psi_\alpha(\|q_j - q_i\|_\sigma) \\ &\quad + \sum_{j\in N_i^\alpha} a_{ij}(q)(p_j - p_i)\Big) \\ &\quad - c_1^{mt}(\overline{q} - q_{mt}) - c_2^{mt}(\overline{p} - p_{mt}). \qquad (32)\end{aligned}$$

Obviously, we see that the pair $(\psi_\alpha, a(q))$ are symmetric. Hence we can rewrite (32) as:

$$\overline{u} \quad = \quad -c_1^{mt}(\overline{q} - q_{mt}) - c_2^{mt}(\overline{p} - p_{mt}). \qquad (33)$$

Equation (33) implies that

$$\begin{cases} \dot{\overline{q}} = \overline{p} \\ \dot{\overline{p}} = -c_1^{mt}(\overline{q} - q_{mt}) - c_2^{mt}(\overline{p} - p_{mt}). \end{cases} \qquad (34)$$

The solution of (34) indicates that the CoM of positions and velocities exponentially converge to those of the target.

The formation (collision-free and velocity matching among mobile sensors) is maintained in the free space tracking because the gradient-based term and the consensus term are considered in this situation (more details please see [2]).

Case 2 (Obstacle space): Since d_i^α is designed to be reduced when each agent senses the obstacles. Therefore, when the sensor network have to pass through the narrow space between two obstacles its size will be shrunk gradually, and when the network already passed this narrow space it grows back to the original size gradually. This reduces the impact of the obstacle on the network hence the speed of sensors can be maintained or the CoM keeps tracking the target. Also, the connectivity and similar formation can be maintained in this scenario.

IV. SIMULATION RESULTS

In this section we will test our adaptive flocking control algorithm (30) and compare it with the existing flocking algorithm (18) in terms of the network connectivity, formation and tracking performance. The parameters used in this simulation are specified as follows:

- Parameters of flocking: number of sensors = 50 (randomly distributed in the box of 100x100 size); $a = b = 5$; $d = 7$; the scaling factor $k_c = 1.2$; the active range $r = k_c * d = 8.4$; $\epsilon = 0.1$ for the σ-norm; h = 0.2 for the bump function ($\phi_\alpha(z)$); h = 0.9 for the bump function ($\phi_\beta(z)$).

- Parameters of target movement: The target moves in the line trajectory: $q_{mt} = [100 + 130t,\ 1t]^T$ with $0 \le t \le 3.5$, and $p_{mt} = (q_{mt}(t) - q_{mt}(t-1))/\Delta_t$ with step size $\Delta_t = 0.002$.

To analyze the connectivity of the network we define a connectivity matrix $c_{ij}(t)$ as follows:

$$c_{ij}(t) = \begin{cases} 1, & if \quad j \in N_i^\alpha(t), i \neq j \\ 0, & if \quad j \notin N_i^\alpha(t), i \neq j \end{cases} \qquad (35)$$

and $c_{ii} = 0$.

Because the rank of Laplacian of a connected graph [2] $c_{ij}(t)$ of order n is at most $(n-1)$ or $rank(c_{ij}(t)) \le (n-1)$, the relative connectivity of a network at time t is defined as

$$C(t) = \frac{1}{n-1} rank(c_{ij}(t)). \qquad (36)$$

If $0 \le C(t) < 1$ the network is broken, and if $C(t) = 1$ the network is connected. Based on this metric we can evaluate the network connectivity in our adaptive flocking control algorithm (30).

Figures 2 represents the results of moving target (red/dark line) tracking in the line trajectory using the existing flocking control algorithm (5). Figures 3 represents the results of moving target tracking in the line trajectory using the adaptive flocking control algorithm (30). Figure 4 shows the results of velocity matching among sensors (a, a'), connectivity (b, b') and error positions between the CoM (black/darker line) and the target (tracking performance) (c, c') of both flocking control algorithms (30) and (5), respectively. To compare these algorithms we use the same initial state (position and velocity) of mobile sensors. By comparing these figures we see that by applying the adaptive flocking control algorithm (30) the connectivity, similar formation and tracking performance are maintained when the network passes through the narrow space between two obstacles (two red/dark circles) while the existing flocking control algorithm (5) could not handle these problems. In Figures 3 when the network enters the small gap between two obstacles its size is shrunk gradually in order to pass this space, then the network size grows back gradually when it passed. Therefore the connectivity and similar formation are maintained.

V. CONCLUSION

This paper studied the approach to flocking control of a mobile sensor network to track and observe a moving target in changing environments. We designed an adaptive flocking control algorithm that can cooperatively learn the network's parameters in a decentralized fashion to change the size of the network in order to maintain connectivity, formation and tracking performance when passing through obstacles. In addition, to see the benefit of the adaptive flocking algorithm we compared it with the normal flocking control algorithm, and we found that the connectivity, similar formation and tracking performance in the adaptive flocking control algorithm are better than those in the existing flocking control algorithm. The computer simulation verified our theoretical results.

REFERENCES

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

[2] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

[3] R. Olfati-Saber. Distributed tracking for mobile sensor networks with information driven mobility. *Proceedings of the 2007 American Control Conference,*, pages 4606–4612, 2007.

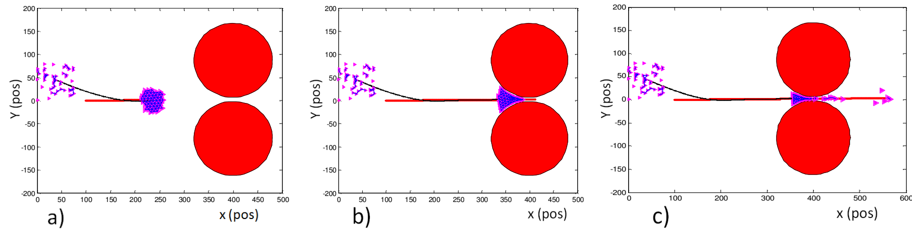

Fig. 2. Snapshots of the mobile sensor network (a) when the mobile sensors form a network, (b) when the mobile sensors avoid obstacles, (c) when the mobile sensors get stuck in the narrow space between two obstacles. These results are obtained by using algorithm (5).

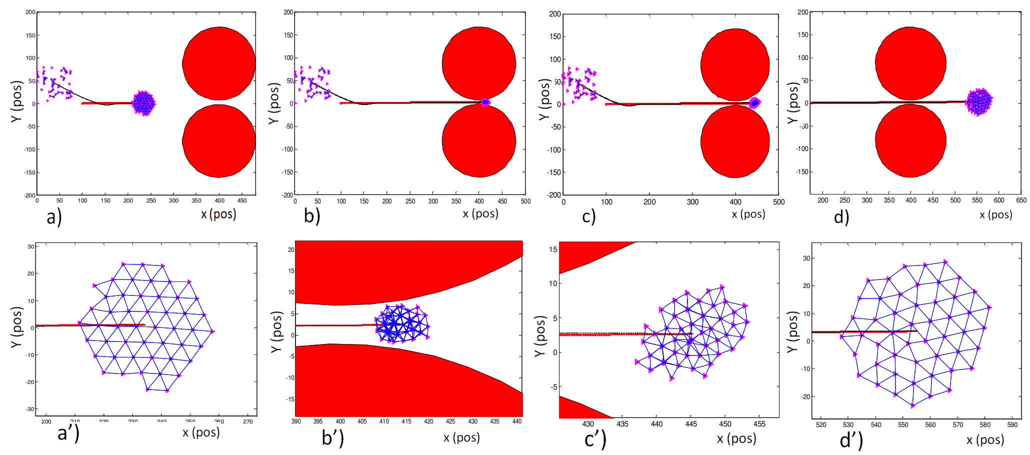

Fig. 3. Snapshots of the mobile sensor network (a) when the mobile sensors form a network, (b) when the mobile sensors avoid obstacles, (c) when the mobile sensors successfully passed through the narrow space between two obstacles, (d) when the mobile sensors recover the original size. (a', b', c', d') are closer look of (a, b, c, d), respectively. These results are obtained by using algorithm (30).

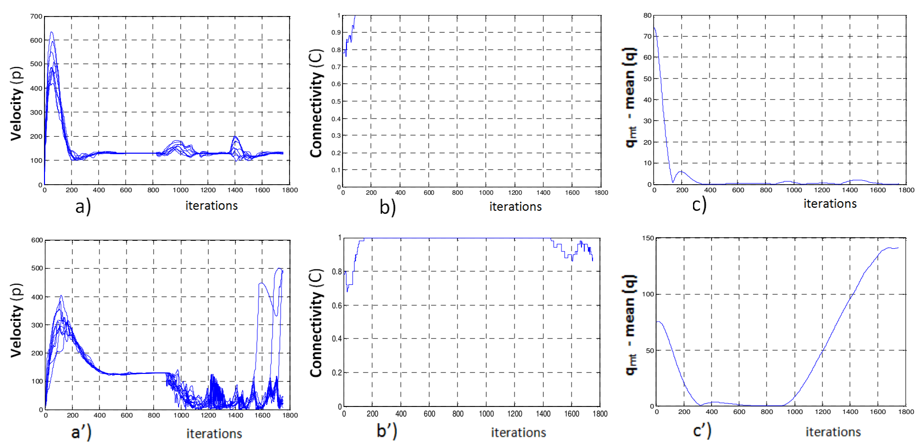

Fig. 4. Velocity matching among sensors, connectivity, and error of positions between CoM and the moving target in (a, b, c) using algorithm (30), (a', b', c') using algorithm (5), respectively.

[4] H. G. Tanner, A. Jadbabai, and G. J. Pappas. Stable flocking of mobile agents, part i: fixed topology. *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 2010–2015, 2003.

[5] H. G. Tanner, A. Jadbabai, and G. J. Pappas. Stable flocking of mobile agents, part ii: dynamic topology. *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 2016–2021, 2003.

[6] Y. Yang, N. Xiong, N. Y. Chong, and X. Defago. A decentralized and adaptive flocking algorithm for autonomous mobile robots. *The 3rd International Conference on Grid and Pervasive Computing–Workshops, IEEE computer society*, pages 262–268, 2008.

[7] G. Lee and N. Y. Chong. Adaptive self-configurable robot swarms based on local interactions. *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4182–4187, 2007.

[8] G. Lee and N. Y. Chong. Adaptive flocking of robot swarms: Algorithms and properties. *IEICE transactions on Communications*, (9):2848–2855, 2008.

[9] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2006.

[10] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time–delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[11] R. Olfati-Saber, J. Alex Fax, and R. M. Murray. Consensus and cooperative in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.