2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Flocking Control of a Mobile Sensor Network to Track and Observe a Moving Target

Hung M. La, Weihua Sheng

School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, 74078
email: hung.la@okstate.edu, weihua.sheng@okstate.edu

Abstract— **This paper presents a new approach to flocking control of a mobile sensor network to track a moving target. In our approach, the center of mass (CoM) of positions and velocities of all mobile sensors in the network (Single-CoM) or the center of mass of position and velocity of each sensor and its neighbors (Multi-CoM) is controlled to track and observe a moving target. In addition, we prove that the CoM of position and velocity exponentially converges to the moving target in free space. Based on this approach, the target is kept at the center of the sensor network. This is of great advantage for sensors to track and observe the target for recognition or identification purposes. In addition, collision-free and velocity matching among mobile sensors are guaranteed in the whole process of the target tracking. We also investigate the stability of our algorithms. The numerical simulations are performed to demonstrate the proposed approach.**

Keywords: Flocking, target tracking, mobile sensor network, obstacle avoidance.

I. Introduction

A. Motivation

Sensor networks [1] have been studied extensively for years, but mostly in stationary environments where sensors do not move. Hence to track and sense a moving target, a stationary network is limited in the accuracy of sensing or need a large number of sensors distributed in the whole environment where the target moves. On the other hand, a mobile sensor network has advantages such as adaptation to environmental changes and reconfigurability for better performance. Therefore mobile sensor networks can be applied in many fields, for example target tracking [2] such as underwater submarine detection, and protection of endangered species [3], etc. A main issue for multiple mobile sensors to track a moving target is that these sensors should move together without collision among them during tracking. This requires us to apply cooperative control methods. One of these methods is flocking control [4], and we know that flocking is a phenomenon in which a number of mobile sensors move together and interact with each other while ensuring the conditions of no collision (avoid collisions with nearby flock-mates), velocity matching (attempt to match velocity with nearby flock-mates), and flock centering (attempt to stay close to nearby flock-mates) [5]. In the nature, schools of fish (see Figure 1) , birds, ants, and bees, etc. demonstrate the phenomena of flocking. The problems of flocking have been studied for many years. These problems have greatly attracted many researchers in physics [6], [7], mathematics [8], biology [9], and especially in control science in recent years [4], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19].

Fig. 1. Schooling of fish (source: http://images.inmagine.com/img/ image100/01028/01028009.jpg).

In this paper, we present a new approach to flocking control of a mobile sensor network to track a moving target while avoiding obstacles. In our approach, the center of mass of positions and velocities of all mobile sensors in the network (Single-CoM) or the center of mass of position and velocity of each sensor and its neighbors (Multi-CoM) is controlled to track a moving target. The main motivation of this approach is to make the CoM of the mobile sensor network track the moving target. This means that all mobile sensors can surround the target closely. This will allow the sensor network to observe and recognize the target easily and accurately. In addition, during the process of tracking collision-free and velocity matching among mobile sensors are guaranteed. We also investigate the stability of our algorithms. We prove that the CoM of position and velocity exponentially converges to the moving target in free space. Numerical simulations are worked out to prove our theoretical results.

B. Literature review

A variety of approaches have been proposed for flocking control. Wang and Gu [13] presented a survey of research achievements of robot flocking. Their paper gave an overview of the related basic knowledge of graph theory, potential function, network communication and system stability analysis. In [4], 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. An extension of flocking algorithms in [4], flocking of agents with a virtual leader in case of a minority of informed agents and in case of varying velocity of virtual leader, was presented in [10] and [11]. Shi and Wang [12] investigated the dynamic properties of mobile agents for the case where the state of virtual leader is time varying and the topology of the neighboring relations between agents is dynamic was proposed. Tanner *et al.* [15] and [16] studied the stability properties of a system of multiple mobile agents with double integrator dynamics in case of fixed topology and dynamic topology. In addition, the experimental implementation of flocking algorithms in [15] and [16] for wheeled mobile robots was presented in [17]. .Olfati-Saber [20] addressed a distributed algorithm for mobile sensor networks based on flocking algorithm to track a moving target. In his paper, an extension of a distributed Kalman filtering algorithm was applied for each sensor estimating the target's position. In [19], a scalable multi-vehicle platform was developed to demonstrate cooperative control systems and sensor networks. Also in their work flocking problem is implemented with five TXT-1 monster truck robots.

The rest of this paper is organized as follows. In the next section we present the basics of the flocking algorithm with obstacle avoidance. Section III presents the Single-CoM and Multi-CoM based flocking control to track and observe a moving target while avoiding obstacles. Section IV presents the simulation results. Finally, Section V concludes this paper.

II. Flocking background

To describe a topology of flocks or swarms we consider a 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 $q_i, p_i \in R^m$ (e.g., $m = 2, 3$) be the position and velocity of node i respectively. We know that during the motion of sensors, the relative distances between them may change, hence the neighbors of each sensor also change. Therefore, we can define a set of neighborhood of sensor i at time t as follows:

$$N_i(t) = \left\{ j \in \vartheta : \|q_j - q_i\| \leq r,\ \vartheta = \{1, 2, ..., n\}, j \neq i \right\} \quad (1)$$

here r is an interaction 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 norm.

We consider n sensors moving in an m dimensional Euclidean space. The dynamic equation of each sensor is described as follows:

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

In [4], Olfati-Saber proposed his control law for flocking of multi mobile agents with obstacle avoidance. This algorithm consists of three components as follows:

$$u_i = f_i^\alpha + f_i^\beta + f_i^\gamma \quad (3)$$

The first component of (3) f_i^α which consists of a gradient-based term and a consensus term is used to regulate the potentials (impulsive or attractive forces) and the velocity among sensors.

$$f_i^\alpha = c_1^\alpha \sum_{j \in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma) n_{ij} + c_2^\alpha \sum_{j \in N_i^\alpha} a_{ij}(q)(p_j - p_i) \quad (4)$$

where each term in (4) is computed as follows [4]:

The set of α neighbors at time t, $N_i^\alpha(t)$, is defined the same as $N_i(t)$ in (1).

The $\sigma - norm$, $\|.\|_\sigma$, of a vector is a map $R^m \Longrightarrow R_+$ defined as $\|z\|_\sigma = 1/\varepsilon[\sqrt{1 + \varepsilon\|z\|^2} - 1]$.

The action function $\phi_\alpha(z)$ that vanishes for all $z \geq r_\alpha$ with $r_\alpha = \|r\|_\sigma$ is defined as follows:

$\phi_\alpha(z) = \rho_h(z/r_\alpha)\phi(z - d_\alpha)$

with the uneven sigmoidal function: $\phi(z) = 0.5[(a + b)\sigma_1(z + c) + (a - b)]$, here $\sigma_1(z) = z/\sqrt{1 + z^2}$, and parameters $0 < a \leq b$, $c = |a - b|/\sqrt{4ab}$ to guarantee $\phi(0) = 0$, and constraints $d_\alpha = \|d\|_\sigma$ with $d = r/k$ for k being the scaling factor (in whole simulation of this paper $k = 1.2$).

The bump function $\rho_h(z)$ with $h \in (0, 1)$ is defined as

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

Vector along the line connecting q_i to q_j is defined as $n_{ij} = (q_j - q_i)/\sqrt{1 + \varepsilon\|q_j - q_i\|^2}$.

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

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

The second component of (3) f_i^β is used to control sensors to avoid obstacles,

$$f_i^\beta = c_1^\beta \sum_{k \in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{n}_{i,k} + c_2^\beta \sum_{k \in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \quad (5)$$

where the set of β neighbors at time t with number of obstacles k is

$$N_i^\beta(t) = \left\{ j \in \vartheta_\beta : \|\hat{q}_{i,k} - q_i\| \leq r', \vartheta_\beta = \{1, 2, ..., k\} \right\} \quad (6)$$

with r' greater than r, in our simulations $r' = 1.2 * r$, and $\hat{q}_{i,k}, \hat{p}_{i,k}$ are the position and velocity of sensor i projecting on the obstacle k, respectively.

Similar to vector n_{ij}, vector $\hat{n}_{i,k}$ is defined as $\hat{n}_{i,k} = (\hat{q}_{i,k} - q_i)/\sqrt{1 + \varepsilon\|\hat{q}_{i,k} - q_i\|^2}$.

The heterogeneous adjacency matrix $b_{i,k}(q)$ is defined as $b_{i,k}(q) = \rho_h(|\hat{q}_{i,k} - q_i\|_\sigma / d_\beta)$ with $d_\beta = \|r'\|_\sigma$.

The repulsive action function of β neighbors is defined as $\phi_\beta(z) = \rho_h(z/d_\beta)(\sigma_1(z - d_\beta) - 1)$. More details of these terms, please see [4].

The third component of (3) f_i^γ is a distributed navigational feedback.

$$f_i^\gamma = -c_1^\gamma(q_i - q_\gamma) - c_2^\gamma(p_i - p_\gamma) \quad (7)$$

here the γ - agent (q_γ, p_γ) is the virtual leader defined as follows

$$\begin{cases} \dot{q}_\gamma = p_\gamma \\ \dot{p}_\gamma = f_\gamma(q_\gamma, p_\gamma). \end{cases} \quad (8)$$

The constants of three components used in (3) 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. Flocking control for tracking and observing a moving target

A. Algorithm Description

In this section, we will extend the above described flocking algorithm with obstacle avoidance [4]. Two problems, named Single-CoM and Multi-CoM, respectively, will be investigated. In the Single-CoM problem, the CoM of positions and velocities of all sensors is controlled to track the moving target. In this case, each sensor need to know the positions and velocity of all other sensors, or it requires the global knowledge of the whole network. To address the scalability problem the Multi-CoM (CoM of positions and velocities of each sensor and its neighbors) problem is studied, where each sensor only need to know the positions and velocity of its neighbors.

In the following algorithms we assume if one of the sensors in the network can estimate the position and velocity of the target, it will broadcast this obtained information to all other nodes. Consequently all sensors in the network can get the knowledge of target.

1) Single-CoM tracking: Firstly, based on Olfati-Saber's flocking algorithm we design an algorithm with a dynamic γ-agent. In this scenario, the dynamic γ-agent is considered as the moving target.

$$\begin{aligned} u_i \;=\; & c_1^\alpha \sum_{j\in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma) n_{ij} + c_2^\alpha \sum_{j\in N_i^\alpha} a_{ij}(q)(p_j - p_i) \\ & + c_1^\beta \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{n}_{i,k} + c_2^\beta \sum_{k\in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \\ & - c_1^{mt}(q_i - q_{mt}) - c_2^{mt}(p_i - p_{mt}) \end{aligned} \tag{9}$$

here the pair $(q_{mt},\ p_{mt})$ is the position and velocity of the moving target, respectively, and $c_1^{mt},\ c_2^{mt}$ are positive constants, and $c_2^{mt} = 2\sqrt{c_1^{mt}}$.

By observing the control protocol (9), we see that the CoM is difficult to reach the target in the presence of obstacles. This creates the difficulty for sensors to track and observe the target. Therefore this protocol should be extended with more constraint on the CoM as follows:

$$u_i = f_i^\alpha + f_i^\beta + f^{mt} \tag{10}$$

where f^{mt} is a tracking feedback applied to sensor i by a moving target with position and velocity $(q_{mt},\ p_{mt})$, respectively.

$$\begin{aligned} f_i^{mt} \;=\; & -c_1^{mt}(q_i - q_{mt}) - c_2^{mt}(p_i - p_{mt}) \\ & - c_1^{mt}(\overline{q} - q_{mt}) - c_2^{mt}(\overline{p} - p_{mt}) \end{aligned} \tag{11}$$

where the pair $(\overline{q},\overline{p})$ is the center of mass (CoM) of positions and velocities of all sensors, respectively, as defined in (12).

$$\begin{cases} \overline{q} = \frac{1}{n}\sum_{i=1}^n q_i \\ \overline{p} = \frac{1}{n}\sum_{i=1}^n p_i. \end{cases} \tag{12}$$

Consequently, the extended control protocol (10) is explicitly specified as follows:

$$\begin{aligned} u_i \;=\; & c_1^\alpha \sum_{j\in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma) n_{ij} + c_2^\alpha \sum_{j\in N_i^\alpha} a_{ij}(q)(p_j - p_i) \\ & + c_1^\beta \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{n}_{i,k} + c_2^\beta \sum_{k\in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \\ & - c_1^{mt}(q_i - q_{mt}) - c_2^{mt}(p_i - p_{mt}) \\ & - c_1^{sc}(\overline{q} - q_{mt}) - c_2^{sc}(\overline{p} - p_{mt}) \end{aligned} \tag{13}$$

here c_1^{sc}, c_2^{sc} are positive constants.

In control protocol (13), each mobile sensor at each time t need to know the position and velocity of all other sensors for computing the CoM $(\overline{q},\overline{p})$. This means that this protocol is limited by the number of sensors, or the scalability is limited because at each time t all other sensors have to send their positions to sensor i. Hence the communication problem is a big challenge and need to be considered when implementing this protocol in real sensor networks.

2) Multi-CoM tracking: To make the algorithm scalable we want to implement a distributed tracking algorithm called Multi-CoM tracking in which the CoM of each sensor and its neighbors is controlled to track the target.

$$\begin{aligned} u_i \;=\; & c_1^\alpha \sum_{j\in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma) n_{ij} + c_2^\alpha \sum_{j\in N_i^\alpha} a_{ij}(q)(p_j - p_i) \\ & + c_1^\beta \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{n}_{i,k} + c_2^\beta \sum_{k\in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \\ & - c_1^{mt}(q_i - q_{mt}) - c_2^{mt}(p_i - p_{mt}) \\ & - c_1^{mc}(\overline{q}_{(i+N_i^\alpha)} - q_{mt}) - c_2^{mc}(\overline{p}_{(i+N_i^\alpha)} - p_{mt}) \end{aligned} \tag{14}$$

here $(c_1^{mc},\ c_2^{mc})$ are the positive constants, and the pair $(\overline{q}_{(i+N_i^\alpha)},\ \overline{p}_{(i+N_i^\alpha)})$ is defined as (15).

$$\begin{cases} \overline{q}_{(i+N_i^\alpha)} = \frac{1}{|N_i^\alpha|+1}\sum_{i=1}^{|N_i^\alpha|+1} q_i \\ \overline{p}_{(i+N_i^\alpha)} = \frac{1}{|N_i^\alpha|+1}\sum_{i=1}^{|N_i^\alpha|+1} p_i \end{cases} \tag{15}$$

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

In control protocol (14), each mobile sensor only need local knowledge, or it means that each sensor only requires the position and velocity knowledge of itself and its neighbors. In α-lattice configuration [4] the maximum number of each sensor's neighbors is 6. Therefore this protocol can scale up to lager mobile sensor networks.

B. Stability Analysis

In this sub-section we will analyze the stability of our algorithms, flocking control with Single-CoM and Multi-CoM, respectively, in free space, and we will explain why the tracking performance in the presence of CoM constraint is better than without CoM constraint in obstacle space.

Theorem 1. In free space, by controlling the CoM based on the control protocol (13), the CoM of positions and velocities of all sensors in the network will exponentially converge to the target. In addition, the formation of all mobile sensors will maintain in the process of the moving target tracking.

Proof: In free space, this means that $\sum_{k\in N_i^\beta}\phi_\beta(\|\hat{q}_{i,k}-q_i\|_\sigma)=0$. Hence we can rewrite control protocol (13) with ignoring constants c_η^ν (for $\forall\eta=1,2$ and $\nu=\alpha,\beta$) as follows:

$$\begin{aligned} u_i &= -\sum_{j\in N_i^\alpha}\nabla_{q_i}\psi_\alpha(\|q_j-q_i\|_\sigma)+\sum_{j\in N_i^\alpha}a_{ij}(q)(p_j-p_i)\\ &\quad -c_1^{mt}(q_i-q_{mt})-c_2^{mt}(p_i-p_{mt}).\\ &\quad -c_1^{sc}(\overline{q}-q_{mt})-c_2^{sc}(\overline{p}-p_{mt}). \end{aligned} \tag{16}$$

where $\psi_\alpha(z)=\int_{d_\alpha}^{z}\phi_\alpha(s)ds$ is the pairwise attractive/repulsive potential function. From (16), we can compute the average of the control law u as follows:

$$\begin{aligned} \overline{u}=\frac{1}{n}\sum_{i=1}^{n}u_i &= \frac{1}{n}\sum_{i=1}^{n}(-\sum_{j\in N_i^\alpha}\nabla_{q_i}\psi_\alpha(\|q_j-q_i\|_\sigma)\\ &\quad +\sum_{j\in N_i^\alpha}a_{ij}(q)(p_j-p_i))\\ &\quad -(c_1^{mt}+c_1^{sc})(\overline{q}-q_{mt})\\ &\quad -(c_2^{mt}+c_2^{sc})(\overline{p}-p_{mt}). \end{aligned} \tag{17}$$

Obviously, we see that the pair $(\psi_\alpha,a(q))$ are symmetric. Hence we can rewrite (17) as:

$$\overline{u} = -(c_1^{mt}+c_1^{sc})(\overline{q}-q_{mt})-(c_2^{mt}+c_2^{sc})(\overline{p}-p_{mt}) \tag{18}$$

Equation (18) implies that

$$\begin{cases} \dot{\overline{q}}=\overline{p}\\ \dot{\overline{p}}=-(c_1^{mt}+c_1^{sc})(\overline{q}-q_{mt})-(c_2^{mt}+c_2^{sc})(\overline{p}-p_{mt}). \end{cases} \tag{19}$$

The solution of (19) indicates that the position and velocity of the CoM will exponentially converge to those of the target.

The formation or collision-free and velocity matching among mobile sensors will be maintained in the free space tracking because the gradient-based term and the consensus term are considered in this situation.

Remark. To see why the tracking performance in the presence of obstacles of the flocking control with Single-CoM is better than that of the flocking control without CoM (No-CoM) (9), we analyze the forces acting on the α-agents (sensors) when they avoid the obstacle as shown in Figure 2. In this figure, without losing generality we simply consider two sensors tracking the target (γ-agent) which moves in an arbitrary trajectory.

Firstly, when two sensors track the target in the free space (without obstacle), in the equilibrium state the CoM is close to the target (in this case sensor2 is the neighbor of sensor1). The total interaction forces between two sensors are equal to zero, and also because of velocity matching the sum of different velocities between these sensors is equal to zero. Hence we obtain $\overline{u}$ as in (18). This means that the CoM ($\overline{q}$, $\overline{p}$) converges to the target (q_{mt}, p_{mt}).

When these sensors move in the obstacle space (the obstacle in the sensing ranges of both sensors) the projection of each sensor on the surface of obstacle is called β-agent (virtual agent). In this scenario, two β-agents, β_1 and β_2, of two sensor1 and sensor2 are created, respectively (see Figure 2). These β-agents generate the repulsive forces, $F_{\beta_1\alpha_1}$ and $F_{\beta_2\alpha_2}$ to push these sensors away from the obstacle. This

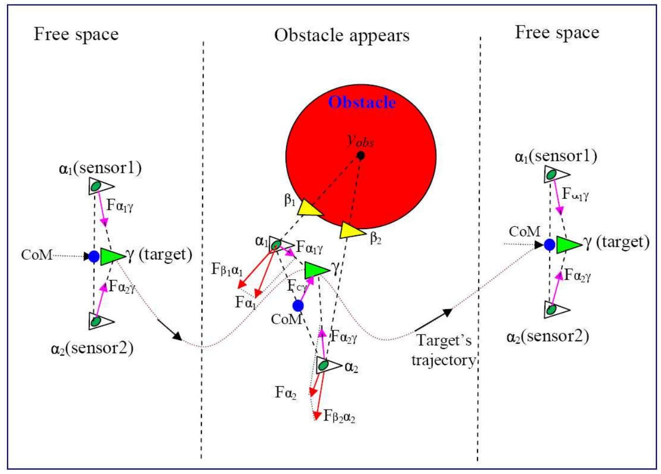

Fig. 2. Demonstration of two sensors tracking the moving target with Single-CoM in both free and obstacle spaces.

causes the sensors being pushed away in a certain distance from the target, or the CoM is no longer close to the target. When the CoM considered as a virtual agent is directly controlled to track the target the offset distance between the CoM and the target, $\|\overline{q}-q_{mt}\|$, creates the negative feedback to the whole system then it makes the CoM converge to the target faster.

In addition, the weights of the attractive force between the target and the CoM c_1^{sc},c_2^{sc} are freely set so that the CoM can converge to the target as soon as possible. Namely, the bigger weight the faster convergence, but if it is too big the overshoot will appear. Keep in mind that the choice of c_1^{sc},c_2^{sc} does not cause the collision with the obstacle. This is different from the choice of c_1^{mt},c_2^{mt} which are selected less than that of c_1^{β},c_2^{β}, respectively. As shown in Figure 2, when the CoM is controlled to track the target directly the force $F_{C\gamma}$ is created to support the sensors to move back to surround the target faster, or the CoM will converge to the target faster. For more information, see the simulation results.

Theorem 2. In free space, by controlling the CoM of each sensor and its neighbors ($\overline{q}_{(i+N_i^\alpha)},\overline{p}_{(i+N_i^\alpha)}$) based on the control protocol (14), the CoM of positions and velocities of all sensors in the network will exponentially converge to the target. Also, the formation, or no collision and velocity matching among mobile sensors maintain during target tracking.

Proof: Proof of this theorem is similar to that of Theorem 1 hence it is omitted.

IV. SIMULATION RESULTS

In this section we test our theoretical results in simulation with different trajectories of the moving target. First of all we test target moving with a sine wave trajectory. Parameters used in this simulation are specified as follows:

- Parameters of flocking: number of sensors = 120; the initial positions of sensors are randomly distributed in a box with a size of [0 90]x[0 90]; the initial velocities of sensors are set to zero. Parameters $a=b=5$; the interaction range

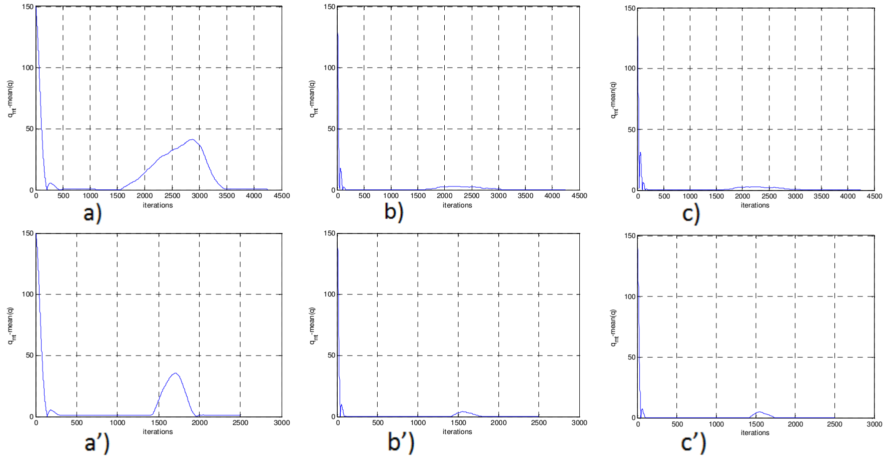

Fig. 3. Error of positions between the CoM’s positions and the moving target in the sine wave trajectory (a, b, c) and the circle trajectory (a’, b’, c’) using flocking control algorithms with No-CoM (9), Single-CoM (13) and Multi-CoM (14), respectively.

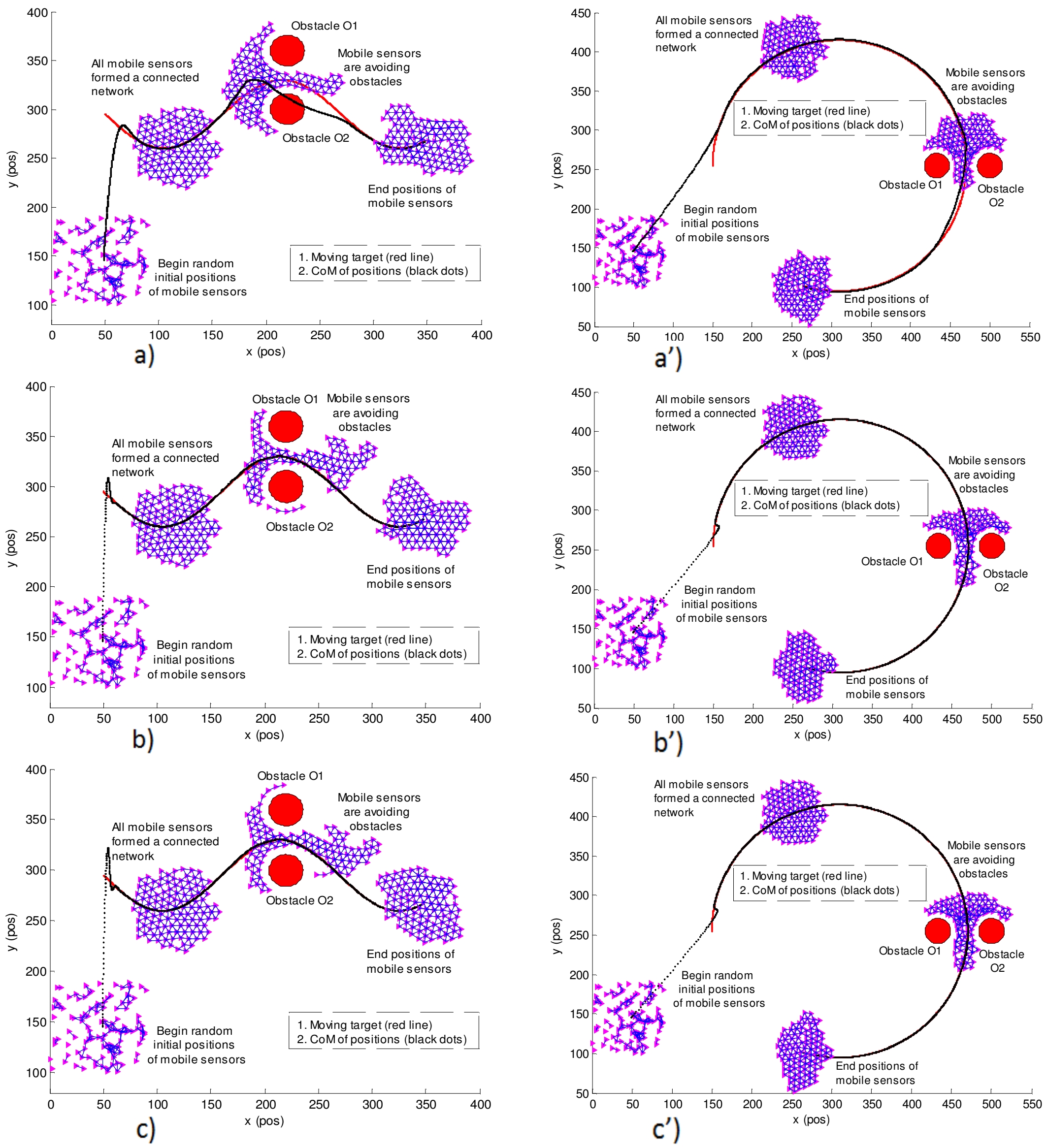

Fig. 4. Snapshots of the positions of the mobile sensors at the beginning, forming a connected network, avoiding obstacles and at the ending when they are tracking the target moving in the sine wave trajectory (a, b, c) and the circle trajectory (a’, b’, c’) using flocking control algorithms with No-CoM (9), Single-CoM (13) and Multi-CoM (14), respectively.

$r = 1.2d = 9$; $\varepsilon = 0.1$ 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 sine wave trajectory: $q_{mt} = [50 + 35t,\ 295 - 35sin(t)]^T$ with $0 \leq t \leq 8.5$, and $p_{mt} = (q_{mt}(t) - q_{mt}(t-1))/\Delta_t$ with $\Delta_t = 0.002$.

Second we test the case where the target moves with a circle trajectory. Parameters used in this simulation are specified as follows:

- Parameters of flocking: parameters used in this case are the same with those in the sine trajectory case.

- Parameters of target movement: The target moves in a circle trajectory: $q_{mt} = [310 - 160cos(t),\ 255 + 160sin(t)]^T$ with $0 \leq t \leq 5$, and $p_{mt} = (q_{mt}(t) - q_{mt}(t-1))/\Delta_t$.

To compare three algorithms No-CoM (9), Single-CoM (13) and Multi-CoM (14) we use the same initial state (position and velocity) of mobile sensors. Figures 3 represents the error between the CoM's positions and the target (tracking performance) in the sine wave and circle trajectories using three algorithms, No-CoM, Single-CoM and Multi-CoM, respectively. Figures 4 represents the snapshots of mobile agents tracking the target moving in the sine wave and circle trajectories using three algorithms, No-CoM, Single-CoM and Multi-CoM, respectively. We see that the results of tracking performance in Figure 3 (b, b', c, c') for both trajectories of the target using Single-CoM and Multi-CoM algorithms, respectively, are better than that in Figure 3 (a, a') using No-CoM algorithm. In addition, we can see the snapshots of mobile robots avoiding obstacle taken at the same time, but in Figures 4 (b, b', c, c') more agents (sensors) passed through the narrow space between two obstacles than that in Figures 4 (a, a'). This means that the CoM in the algorithms Single-CoM and Multi-CoM (Figures 3 b, b', c, c') is closer to the target than that in the No-CoM algorithm (Figures 3 a, a').

V. Conclusion

This paper studied the approach to flocking control of a mobile sensor network to track and observe a moving target. We designed a flocking control algorithm with Single-CoM and Multi-CoM to enable mobile sensors to track and observe the moving target more effectively while maintaining their formation and no collision among them. We prove that the CoM of positions and velocities of all mobile sensors exponentially converges to the target. By controlling the CoM explicitly, the mobile sensor network can track and observe the moving target better. This means that all mobile sensors in the network can surround the target closely which will allow them to see the target easily for recognition purpose. In addition, flocking control with No-CoM, flocking control with Single-CoM, and flocking control with Multi-CoM are compared. The numerical simulations are done with different target trajectories to demonstrate our theoretical results.

Acknowledgement

We would like to thank the Vietnamese Government, and specially thank to the MOET (Ministry of Education and Training), who supported us to implement this project.

References

[1] D. Culler, D. Estrin, , and M. Srivastava. Overview of sensor networks. *IEEE Computer*, 37(8):41–49, 2004.

[2] A. Ganguli, S. Susca, S. Martinez, F. Bullo, and J. Cortes. On collective motion in sensor networks: sample problems and distributed algorithms. *in Proc. 44th IEEE Conf. Decision and Control, and the European Control Conference 2005*, pages 4239–4244, December 12-15 2005.

[3] E. Biagioni and K. Bridges. The application of remote sensor technology to assist the recovery of rare and endangered species. *Special issue on Distributed Sensor Networks for the International Journal of High Performance Computing Applications*, 16(3):112–121, 2002.

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

[5] C. Reynolds. Flocks, birds, and schools: A distributed behavioral model. *Computer Graphics,ACM SIGGRAPH '87 Conference Proceedings, Anaheim, California*, 21(4):25–34, 1987.

[6] T. Vicsek, A. Czirok, E.B. Jacob, I. Cohen, and O. Schochet. Novel type of phase transitions in a system of self-driven particles. *Phys. Rev. Lett*, 75:1226–1229, 1995.

[7] H. Levine, W. J. Rappel, and I. Cohen. Self-organization in systems of self-propelled particles. *Phys. Review. E*, 63:017101–017104, 2000.

[8] A. Mogilner, L. Edelstein-Keshet, L. Bent, and A. Spiros. Mutual interactions, potentials, and individual distance in a social aggregation. *J. Math. Biol*, 47:353–389, 2003.

[9] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks. Collective memory and spatial sorting in animal groups. *J. Theor.Biol*, 218:1–11, 2002.

[10] H. Su, X. Wang, and Z. Lin. Flocking of multi–agents with a virtual leader part i: with a minority of informed agents. *in Proc. 46th IEEE Conf. Decision and Control*, pages 2937–2942, 2007.

[11] H. Su, X. Wang, and Z. Lin. Flocking of multi–agents with a virtual leader part ii: with a virtual leader of varying velocity. *in Proc. 46th IEEE Conf. Decision and Control*, pages 1429–1434, 2007.

[12] H. Shi, L. Wang, T. Chu, F. Fu, and M. Xu. Flocking of multi–agents with a virtual leader. *in Proc of the 2007 IEEE Symposium on Artificial Life*, pages 287–29, 2007.

[13] Z. Wang and D. Gu. A survey on application of consensus protocol to flocking control. *in Proc of the 12th Chinese Automation and Computing Society Conference in the UK, England*, pages 1–8, 2006.

[14] M. Anderson, E. McDaniel, and S. Chenney. Constrained animation of flocks. *Eurographics/SIGGRAPH Symposium on Computer Animation*, pages 1–12, 2003.

[15] 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.

[16] 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.

[17] A. Regmi, R. Sandoval, R. Byrne, H. Taner, and C. T. Abdallah. Experimental implementation of flocking algorithms in wheeled mobile robots. *American Control Conference*, pages 4917–4922, 2005.

[18] V. Gervasi and G. Prencipe. Coordination without communication: the case of the flocking problem. *Discrete Applied Mathematics*, pages 324–344, 2004.

[19] D. Cruz, J. McClintock, B. Perteet, O. A. orqueda, Y. Cao, and R. Fierro. Decentralized cooperative control–a multivehicle platform for research in networked embedded systems. *IEEE Control Systems magazine*, pages 58–78, 2007.

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