The 11th International Symposium on Visual Computing (ISVC),
December 14-16, 2015 Las Vegas, Nevada, USA.

Adaptive Flocking Control of Multiple Unmanned Ground Vehicles by Using a UAV

Mohammad Jafari(✉), Shamik Sengupta, and Hung Manh La

Department of Computer Science and Engineering,
University of Nevada, Reno 89557, USA
mohamadj@gmail.com, {ssengupta,hla}@unr.edu

Abstract. In this paper we aim to discuss adaptive flocking control of multiple Unmanned Ground Vehicles (UGVs) by using an Unmanned Aerial Vehicle (UAV). We utilize a Quadrotor to provide the positions of all agents and also to manage the shrinking or expanding of the agents with respect to the environmental changes. The proposed method adaptively causes changing in the sensing range of the ground robots as the quadrotor attitude changes. The simulation results show the effectiveness of proposed method.

Keywords: Hybrid system · Multi-agent · Flocking control · UAVs · UGVs

1 Introduction

Multi-agent control has attracted a lot of interests in recent years. A multi-agent system consists of a group of agents (for instance: robots, vehicles, etc.) cooperating with each other to do a task. The importance of multi-agent systems is shown specifically when a single agent can not easily solve a problem by itself. Utilizing a team of robots has several advantages over a single one such as: (i) there are some intrinsically distributed tasks, (ii) working multiple robots simultaneously, parallelizable problems can be solved in less time, (iii) minimizing the effects of failure by working together to accomplish the same task [1,2].

The applications of multiple robots include but not limited to rescue operations, environmental monitoring, robot-soccer and target tracking [1–6]. Search and rescue operations are definitely among the most important tasks that should be done when natural and/or manmade disasters such as earthquakes, tsunamis or explosions happen. Disaster environments are always hazardous and dangerous for people. In past years, researchers mostly focused on UGVs for employing them in search and rescue operations. Recently, UAVs attract many researchers because of their various capabilities. Employing UAVs and UGVs together as a hybrid system, UAVs could cooperate with UGVs to better accomplish the tasks.

Utilizing hybrid systems raises specific challenges and questions. Why do we need hybrid systems? What are their advantages? What kind of UAVs are more appropriate for hybrid systems? What kind of constraints need to be considered?

G. Bebis et al. (Eds.): ISVC 2015, Part II, LNCS 9475, pp. 628–637, 2015.
DOI: 10.1007/978-3-319-27863-6_58

What are the challenges of using hybrid systems? Because of the challenges of disaster environments (i.e., buildings pose 3-D constraints on visibility, low visibility, communication, GPS, etc.) a network of aerial and ground vehicles working in cooperation is more beneficial. Also in noisy cluttered environments top layer UAVs, by relying on measurements from the GPS/IMU and camera parameters, could help bottom layer UGVs by providing localization data and acting as communication relays. For example UAVs at the higher attitude could localize UGVs using a sequence of images taken which relates the position of the robot in a global coordinate frame with its pixel coordinates in the image or by using a set of known landmarks in the image.

Furthermore, having adaptive formation for multiagent systems seems to be more practical compared with fixed formation of the system. Changing the formation can be caused by several reasons that include but not limited to:

(i) Covering the greater part of the environment: In some application such as mapping an area, it is important that the multiagent system has the ability to spread out or gather. Therefore, by having the ability to change the formation, the agents can successfully accomplish the predefined mission.
(ii) Moving along an obstacle or through obstacles: Changing the formation in case of facing an obstacle is a preferable way to avoid collision.

"Formation control is a multi-agent application where the objective is for the robotic network to move into a desired formation (particular pattern, i.e., triangular) and to accomplish the desired task" [6]. There are diverse approaches for formation control of a multi-agent system such as: the behavior based, virtual structure, leader follower, and graph theoretical approaches [3].

In the leader-follower approach, several mobile robots are chosen as leaders and the rest of robots as followers. The leaders track predefined trajectories while the followers chase after the leaders. The main advantage of this method is its simplicity [3]. The goal of virtual structure method is to force a group of agents to stay in a rigid formation [5]. However, existing approaches have some limitations dealing with the terrain changes, for example when the UGV fleet has to pass a narrow space through a difficult terrain. Formation of the fleet might be changed or the connectivity of them might be lost. Also, the fleet might get stuck and this would affect the tracking performance. Furthermore, it is difficult for each agent to sense the whole area and estimate the size of the obstacles or the passing space among them. Therefore, designing an adaptive flocking control by utilizing a UAV to providing the global information about the environment is an interesting and challenging task.

The main contribution of this paper is to propose a new approach to the adaptive flocking control with the consideration of utilizing a quadrotor to provide the position information for ground agents and to manage their shrinking or expansion through altitude adaptation so the collision and obstacle avoidance could perform more effectively. The proposed hybrid system idea can allow ground agents to shrink or expand to effectively avoid the obstacles due to the difficult terrain. The simulation results for the proposed work is given (Fig. 1).

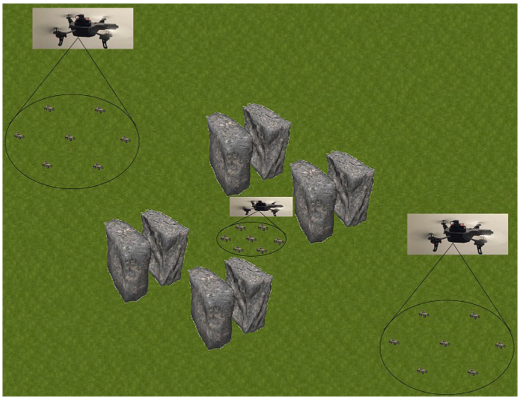

Fig. 1. Agents shrinking and expansion due to the terrain changes.

The rest of the paper is organized as follows. The next section presents modeling of both UGVs and UAV. Section 3 addresses the problem definition and our proposed formulation for hybrid system. The simulation results are presented in Sect. 4. The conclusion of the paper is provided in Sect. 5.

2 Preliminaries

2.1 Modeling Unmanned Ground Vehicles

We consider the flocks topology for modeling the Unmanned Ground Vehicles (UGVs) in this paper. It is based on the model introduced by Olfati-Saber [7] in 2006. Assuming that n agents are moving in an m dimensional space ($m = 2, 3$), the equation of motion of each dynamic agent could be described as follow:

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

where $q_i, p_i, u_i \in \mathbb{R}^{\mathrm{m}}$ are position,velocity, and control input of the agent i respectively. Let us consider a dynamic graph $G(\upsilon, \varepsilon)$ consisting of a set of edges and vertices as follow: $\upsilon = \{1, 2, ..., n\}, \varepsilon \subseteq \{(i, j) : i, j \in \upsilon, j \neq i\}$. In topology of flocks each agent is represented by a vertex and each edge shows that there exists a communication link between two agents. The neighborhood set of agent i could be defined by

$$N_i^{\alpha} = \{j \in \upsilon_{\alpha} :\| q_j - q_i\| < r, j \neq i\} \tag{2}$$

where r is an interaction range between two agents and $\||\cdot\|$ is the Euclidean norm in $\mathbb{R}^{\mathrm{m}}$. By solving the following set of algebraic conditions, a geometric model of flocks (α -lattice) [7] could be described:

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

where d (a positive constant) is the distance between two neighbors i and j. The above equation causes a singularity for the collective potential function at $q_i = q_j$. To resolve the aforementioned problem, (3) could be rewritten as follow [7]:

$$\| q_j - q_i\|_\sigma = d_\alpha \quad \forall j \in N_i^\alpha \tag{4}$$

where $d_\alpha = \| d\|_\sigma$ and $\| \cdot\|_\sigma$ is called σ-norm which is defined as follow: $\| z\| = \frac{1}{\epsilon}[\sqrt{1+\epsilon\| z\|^2} - 1], \epsilon > 0$. For a vector z, σ-norm is a map from $\mathbb{R}^{\mathrm{m}}$ to $\mathbb{R} \geq 0$. The new map (i.e., $\| z\|_\sigma$) is differentiable everywhere while the Euclidean norm (i.e., $\| z\|$) is not differentiable at $z = 0$. A smooth collective potential function which is induced by the above constraints is obtain from: $V(q) = \frac{1}{2}\sum_i \sum_{j\neq i} \psi_\alpha(\| q_j - q_i\|_\sigma)$. where $\psi_\alpha(z)$ is a smooth pairwise potential function which is defined by $\psi_\alpha(z) = \int_{d_\alpha}^{z} \phi_\alpha(s)ds$. Here $\phi_\alpha(z) = \rho_h(z/r_\alpha)\phi(z - d_\alpha)$, $\phi(z) = \frac{1}{2}[(a+b)\sigma_1(z+c) + (a-b)]$ and $\sigma_1(z) = z/\sqrt{1+z^2}$ while $\phi(z)$ is an uneven sigmoidal function with parameters $0 < a \leq b, c = |a-b|/\sqrt{4ab}$ to guarantee $\phi(0) = 0$. There is a bump function $\rho(z)$ which is scalar function that smoothly varies between [0,1]. One possible choice is the following [7]:

$$\begin{cases} 1, & z \in [0,h) \\ \frac{1}{2}\left[1 + cos(\pi\frac{(z-h)}{(1-h)})\right], & z \in [h,1] \\ 0, & otherwise \end{cases} \tag{5}$$

The main flocking control algorithm ($u_i = u_i^\alpha + u_i^\beta + u_i^\gamma$) in [7] has the capability of controlling all agents to form a lattice configuration (which is called α-lattice) while avoiding the obstacles. The algorithm consists of three terms: u_i^α is the interaction component between two α-agents, u_i^β is the interaction component between α-agent and an obstacle (which is called β-agent), and u_i^γ is a goal component which consist of a distributed navigational feedback.

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

$$u_i^\beta = c_1^\beta \sum_{k\in N_i^\beta} \phi_\beta(\| \hat{q}_{i,k} - q_i\|_\sigma)\hat{\mathbf{n}}_{i,k} + c_2^\beta \sum_{k\in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \tag{7}$$

$$u_i^\gamma = -c_1^\gamma\sigma_1(q_i - q_r) - c_2^\gamma(p_i - p_r) \tag{8}$$

where $c_1^\alpha, c_2^\alpha, c_1^\beta, c_2^\beta, c_1^\gamma$, and c_2^γ are all positive constants. The pair (q_r, p_r) is the virtual leader (i.e., γ-agent) which could be defined as follow:

$$\begin{cases} \dot{q}_r = p_r \\ \dot{p}_r = f_r(q_r, p_r) \end{cases} \tag{9}$$

The vectors $\mathbf{n}_{i,j}$ and $\hat{\mathbf{n}}_{i,k}$ are described as below: $\mathbf{n}_{i,j} = \frac{q_j - q_i}{\sqrt{1+\epsilon\|q_j - q_i\|^2}}$, $\hat{\mathbf{n}}_{i,k} = \frac{\hat{q}_{i,k} - q_i}{\sqrt{1+\epsilon\|\hat{q}_{i,k} - q_i\|^2}}$. The $a_{ij}(q)$ and $b_{i,k}(q)$ are the elements of spatial adjacency matrix $A(q)$ and heterogeneous adjacency matrix $B(q)$ respectively which could be defined by: $a_{ij}(q) = \rho_h(\| q_j - q_i\|_\sigma / r_\alpha) \in [0,1], j \neq i$, $b_{i,k}(q) = \rho_h$ $(\| \hat{q}_{i,k} - q_i\|_\sigma / d_\beta)$. where $r_\alpha = \| r\|_\sigma$, $a_{ii}(q) = 0$ for all i and q, $d_\beta = \| d^{'}\|_\sigma$, and $r_\beta = \| r^{'}\|_\sigma$. The repulsive action function $\phi_\beta(z)$ is defined as: $\phi_\beta(z) = \rho_h(z/d_\beta)(\sigma_1(z - d_\beta) - 1)$.

Similar to (2) we can define the set of β-neighbors of an α-agent i as follow: $N_i^\beta = \{k \in \upsilon_\beta : \| \hat{q}_{i,k} - q_i\| < r^{'}\}$, where $r' > 0$ is interaction range of an α-agent with obstacles.

2.2 Modeling Unmanned Aerial Vehicles

Nowadays, the knowledge of UAVs is at the cutting edge of researches. Among them, flights which have the capability of Vertical Takeoff and Landing (VTOL), specially Quadrotors, are under our consideration. Various control methods have been employed for stabilizing and controlling of Quadrotors. Coza et al. [8] utilized Adaptive fuzzy control for a quadrotor helicopter. Adaptive neural control of a quadrotor helicopter with extreme learning machine was designed by Zhang et al. [9]. In [10] adaptive sliding mode control design was presented. Attitude control of a quadrotor using brain emotional learning based intelligent controller was proposed in 2013 [11]. Bou-Ammar and his colleagues [12] employed the reinforcement method to control the quadrotor. Nonlinear robust output feedback tracking control of a quadrotor by using quaternion representation was proposed in [13]. Robust attitude controller design for miniature quadrotors has been recently proposed by Liu et al. [14]. To model a quadrotor, a hybrid frame consisting of E-frame and B-frame was utilized [15], where the equations in H-frame (i.e., linear equation WRT E-frame and angular equations WRT B-frame) were defined as follows:

$$\begin{cases} \ddot{X} = (sin\psi sin\phi + cos\psi sin\theta cos\phi)\frac{U_1}{m} \\ \ddot{Y} = (-cos\psi sin\phi + sin\psi sin\theta cos\phi)\frac{U_1}{m} \\ \ddot{Z} = -g + (cos\theta cos\phi)\frac{U_1}{m} \\ \dot{p} = \frac{I_{YY} - I_{ZZ}}{I_{XX}}qr - \frac{J_{TP}}{I_{XX}}q\boldsymbol{\Omega} + \frac{U_2}{I_{XX}} \\ \dot{q} = \frac{I_{ZZ} - I_{XX}}{I_{YY}}pr - \frac{J_{TP}}{I_{YY}}p\boldsymbol{\Omega} + \frac{U_3}{I_{YY}} \\ \dot{r} = \frac{I_{XX} - I_{YY}}{I_{ZZ}}pq + \frac{U_4}{I_{ZZ}} \end{cases} \tag{10}$$

where ψ, ϕ, and θ are Yaw, Roll, and Pitch angles respectively. I_{XX}, I_{YY}, and I_{ZZ} are body moment of inertia around x, y, and z axis respectively. J_{TP} is total rotational moment of inertia around propeller axis, g is the acceleration due to gravity and m is quadrotor mass. Relation between basic movements and the propellers' speed are defined by

$$\begin{cases} U_1 = b_q(\boldsymbol{\Omega}_1^2 + \boldsymbol{\Omega}_2^2 + \boldsymbol{\Omega}_3^2 + \boldsymbol{\Omega}_4^2) \\ U_2 = b_q l(-\boldsymbol{\Omega}_2^2 + \boldsymbol{\Omega}_4^2) \\ U_3 = b_q l(-\boldsymbol{\Omega}_1^2 + \boldsymbol{\Omega}_3^2) \\ U_4 = d_q(-\boldsymbol{\Omega}_1^2 + \boldsymbol{\Omega}_2^2 - \boldsymbol{\Omega}_3^2 + \boldsymbol{\Omega}_4^2) \\ \boldsymbol{\Omega} = -\boldsymbol{\Omega}_1 + \boldsymbol{\Omega}_2 - \boldsymbol{\Omega}_3 + \boldsymbol{\Omega}_4 \end{cases} \tag{11}$$

where $\boldsymbol{\Omega}_1$, $\boldsymbol{\Omega}_2$, $\boldsymbol{\Omega}_3$, and $\boldsymbol{\Omega}_4$ are front, right, rear, and left propeller speeds respectively. Also, b_q is trust factor of the quadrotor, d_q is drag factor of the quadrotor and l is distance between center of the quadrotor and center of the propeller.

3 Problem Definition

There are several challenges existing when a fleet of UGVs has to pass through difficult terrain with obstacles. The fleet might get stuck behind the obstacles and this will cause problem in tracking the target [7]. Also because of 3-D constraints on visibility, communication constraints, GPS denied area and so on, the UGV fleet might lost their connectivities and/ or their path. Because of their capabilities, UAVs can be of help in guiding the UGV fleet. In this work, we are proposing methodology for UAV so that it can help the UGV fleet most effectively.

The idea of this work is based on changing of the quadrotor height. As the UAV senses the area is free it flies to the higher attitude so this should force the agents to expand and cover more area. On the contrary, if the UAV sees any obstacle or narrow path it will fly down to a lower height and this will cause the agents to shrink so they could successfully pass the narrow area. We assume that the quadrotor is providing the positional information for the UGVs. As Fig. 2 (a) shows, changing of the quadrotor height will change the Field of View (FOV) of the bottom camera. The higher the quadrotor flies, the bigger FOV will be achieved. The radius of FOV of the quadrotor could be defined as follow:

$$r_{FOV} = H_q tan(\alpha_q) \tag{12}$$

where H_q is the quadrotor attitude and α_q is the half-angle of view of the quadrotor bottom camera. The quadrotor tracks the same trajectory as the UGV fleet while its x-y position is same as the center of mass of the UGV fleet. The x-y position of the quadrotor can be defined by:

$$\bar{q} = \frac{1}{n}\sum_{i=1}^{n} q_i \tag{13}$$

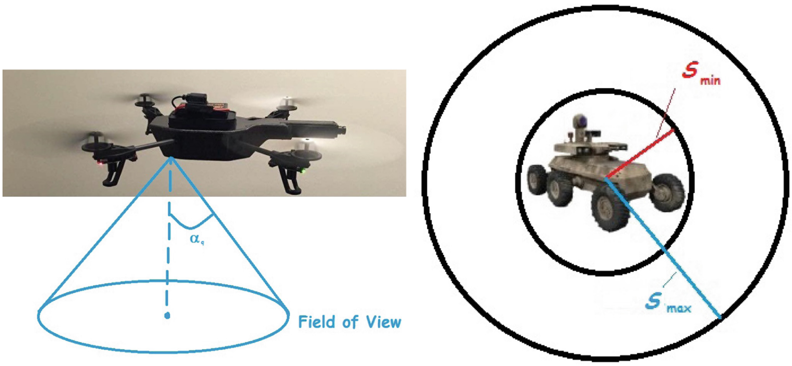

Fig. 2. (a) Field of View of quadrotor bottom camera (Left), (b) Minimum and Maximum sensing range of UGV (Right).

For implementing the above idea we will utilize a function F(h) which will help us insert the above mentioned method to the system. The function F should have the following properties:

1. $F(h) = 0, \qquad \forall 0 < h \leq h_{min}$
2. $0 < F(h) < S_{dif}, \forall h_{min} < h \leq h_{max}$
3. $F(h)$ is smooth and differentiable

Where h_{min} and h_{max} are minimum and maximum heights of the UAV respectively. S_{dif} could be obtained as follow:

$$S_{dif} = S_{max} - S_{min} \tag{14}$$

where S_{max} and S_{min} are maximum and minimum sensing ranges of the ground agents respectively (depicted in Fig. 2 (b)).

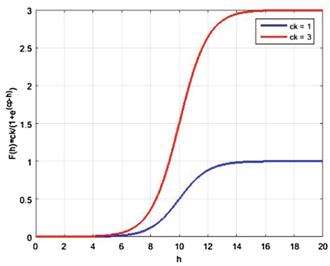

Fig. 3. Function $F(h)$ with two different c_k.

We introduce the following function which satisfies all the above mentioned properties:

$$F(h) = c_k \frac{1}{1 + e^{(c_p - h)}} \tag{15}$$

where c_k and c_p are two positive constants. c_p affects the smoothness of the $F(h)$ (which defines the speed of shrinking or expansion of the agents) while c_k changes the variation range of the $F(h)$. Figure 3 demonstrates the $F(h)$ with two different $c_k = 3$ and $c_k = 1$, and $c_p = 10$. We then rewrite equation (4) to include the effect of quadrotor height changes to the shrinking or expanding of the flocks. Below is the new equation which is employed in our proposed work:

$$\| q_j - q_i \|_\sigma = d_\alpha^{new} + F(h) \quad \forall j \in N_i^\alpha \tag{16}$$

where d_α^{new} is the minimum sensing range of the ground agents.

It could be clearly seen as the height of the quadrotor is changing it affects the sensing range of the ground agents and force them to expand or shrink as they are moving towards their goals. The ground agents still have a capability of avoiding the collision or obstacles by themselves.

4 Simulation Results

In this section two different simulation results from 2-D and 3-D view prospective are presented. The following parameters are used through the simulations: $d_\alpha^{new} = 4$, $r = 1.2 d_\alpha^{new}$, $d = 0.6 d_\alpha^{new}$, $r = 1.2d$, for σ-norm $\epsilon = 0.1$, $a = b = 5$ (for $\phi(z)$), for the bump functions of $\phi_\alpha(z)$ and $\phi_\alpha(z)$, $h = 0.2$ and $h = 0.9$ respectively. Other parameters of the algorithm and the initial positions and velocities of all agents are specified in each case separately.

4.1 3-D View

We employed 10 ground agents which are randomly distributed in the $[-40, 80]^2$. The initial velocities of all agents are equally chosen as zero. We used the obstacles which can be defined from following matrix:

$$OBS_1 = \begin{bmatrix} 220 & 220 & 260 & 260 & 300 & 300 & 340 & 340 \\ -30 & 70 & -30 & 70 & -30 & 70 & -30 & 70 \\ -15 & -15 & -13 & -13 & -15 & -15 & -13 & -13 \\ 20 & 20 & 20 & 20 & 20 & 20 & 20 & 20 \end{bmatrix}.$$

The first three rows of the matrix are the locations (i.e., x-y-z position) of the obstacles, while the last row shows the radius of the obstacles. Figure 4 demonstrates the shrinking of the agents. As the quadrotor senses the obstacles, it flies in a lower attitude and this is forcing the ground agents to reduce the inter-agent distance between them and it results in shrinking. As it could clearly be seen the flock size successfully reduced.

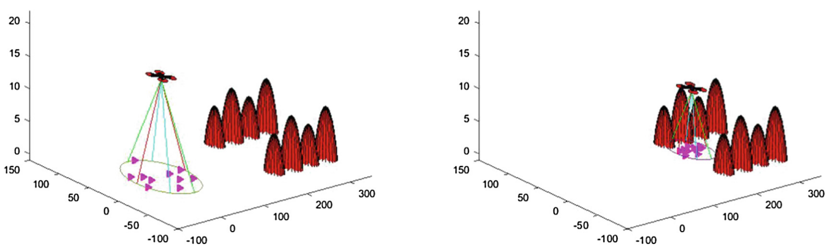

Fig. 4. Shrinking of the agents.

4.2 2-D View

We employed 150 ground agents which are randomly distributed in the $[-40, 80]^2$. The initial velocities of all agents are equally chosen as zero. We used the obstacles which can be defined from following matrix:

$$OBS_2 = \begin{bmatrix} 220 & 220 & 260 & 260 & 300 & 300 & 340 & 340 \\ -20 & 80 & -20 & 80 & -20 & 80 & -20 & 80 \\ 40 & 40 & 40 & 40 & 40 & 40 & 40 & 40 \end{bmatrix}.$$

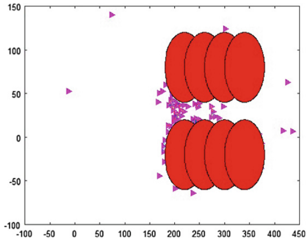

Fig. 5. 2D view of Shrinking of the agents- algorithm from [7].

The first two rows of the matrix are the locations (i.e., x-y position) of the obstacles, while the last row shows the radius of the obstacles. Figure 5 shows the result which is obtained using algorithm in [7]. As it is clearly seen the ground agents get stuck in the narrow path between the obstacles. Figure 6(Left) demonstrates how the ground agents are shrinking and passing the narrow space between obstacles while avoiding any collisions between agents or obstacles. Figure 6(Right) illustrates the closer look of Fig. 6(Left).

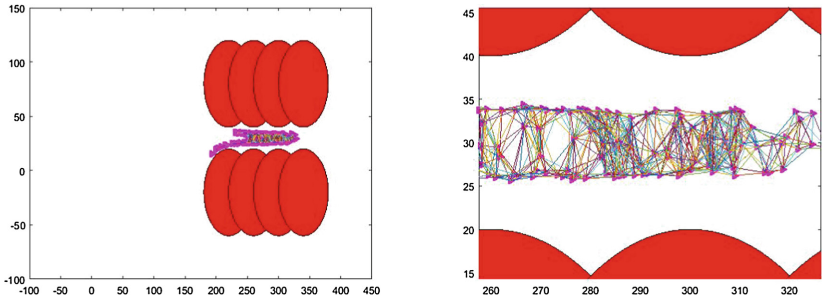

Fig. 6. 2D view of Shrinking of the agents.

All these results are obtained using proposed adaptive flocking method. The results shows the satisfactory performances of the proposed work.

5 Conclusions

Adaptive flocking control of multiple unmanned ground vehicles by using an unmanned aerial vehicle is presented in this paper. We employed a Quadrotor to provide the positions of all agents and also to manage the shrinking or expanding of the agents with respect to the terrain changes. The simulation results show the effectiveness of proposed method.

References

1. Sabattini, L.: Nonlinear Control Strategies for Cooperative Control of Multi-Robot Systems. Ph.D. thesis, Universitá di Bologna (2012)
2. Garg, D.P., Fricke, G.K.: Potential function based formation control of mobile multiple-agent systems. In: 1st International and 16th National Conference on Machines and Mechanisms (iNaCoMM2013) (2013)
3. Dong, W., Guo, Y., Farrell, J.: Formation control of nonholonomic mobile robots. In: American Control Conference, pp. 5602–5607. IEEE (2006)
4. Speranzon, A.: On Control Under Communicaiton Constraints in Autonomous Multi-Robot Systems. KTH, Signals, Sensors and Systems, Stockholm (2004)
5. Arranz, L.B.: Cooperative control design for a fleet of AUVs under communication constraints. Ph.D. thesis, Université de Grenoble (2011)
6. Clark, J.D.: Cooperative hybrid control of robotic sensors for perimeter detection and tracking. Master's thesis, Oklahoma State University (2005)
7. Olfati-Saber, R.: Flocking for multi-agent dynamic systems: algorithms and theory. IEEE Trans. Autom. Control **51**, 401–420 (2006)
8. Coza, C., Nicol, C., Macnab, C., Ramirez-Serrano, A.: Adaptive fuzzy control for a quadrotor helicopter robust to wind buffeting. J. Intell. Fuzzy Syst.: Appl. Eng. Technol. **22**, 267–283 (2011)
9. Zhang, Y., Xu, B., Li, H.: Adaptive neural control of a quadrotor helicopter with extreme learning machine. In: Cao, J., Mao, K., Cambria, E., Man, Z., Toh, K.-A. (eds.) Proceedings of ELM-2014 Volume 2, PALO, vol. 4, pp. 125–134. Springer, Heidelberg (2014)
10. Islam, S., Faraz, M., Ashour, R., Cai, G., Dias, J., Seneviratne, L.: Adaptive sliding mode control design for quadrotor unmanned aerial vehicle. In: 2015 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 34–39. IEEE (2015)
11. Jafari, M., Shahri, A.M., Shouraki, S.B.: Attitude control of a quadrotor using brain emotional learning based intelligent controller. In: 2013 13th Iranian Conference on Fuzzy Systems (IFSC), pp. 1–5. IEEE (2013)
12. Bou-Ammar, H., Voos, H., Ertel, W.: Controller design for quadrotor uavs using reinforcement learning. In: 2010 IEEE International Conference on Control Applications (CCA), pp. 2130–2135. IEEE (2010)
13. Xian, B., Diao, C., Zhao, B., Zhang, Y.: Nonlinear robust output feedback tracking control of a quadrotor UAV using quaternion representation. Nonlinear Dyn. **79**, 2735–2752 (2015)
14. Liu, H., Li, D., Xi, J., Zhong, Y.: Robust attitude controller design for miniature quadrotors. Int. J. Robust Nonlinear Control (2015)
15. Bresciani, T.: Modelling, Identification and Control of a Quadrotor Helicopter. Lund University, Department of Automatic Control, Lund (2008)