Proceedings of the 2007 IEEE
International Conference on Mechatronics and Automation
August 5 - 8, 2007, Harbin, China

# Fuzzy Control of Distributed Flocking System

Zongyao Wang
*Department of Computer Science*
*University of Essex*
*Colchester, UK*
*zwangf@essex.ac.uk*

Dongbing Gu
*Department of Computer Science*
*University of Essex*
*Colchester, UK*
*dgu@essex.ac.uk*

***Abstract*—This paper presents a fuzzy based flocking system. To maintain the distance between agents, we use fuzzy logic controller to design a "force function" which is related to the relative distance between neighbours. And the "force function" is used to control the velocity of agent. To prove the stability of flocking system, we build a Hamilton function which is the kinetic energy of flocking system. Utilizing the LaSalle's invariance principle, we prove that the system is stable. Specially, we developed a local form flocking controller which is derived from the global flocking algorithm. By using the local controller, agents in the flocking system only need to know the local information (relative distances and relative angles between neighbours). To evaluate the performance, we simulate the flocking system with 200 agents under the control of distributed flocking algorithm.**

***Index Terms*—Multiple robot, sensor network, distributed system.**

## I. Introduction

The flocking behaviour of living beings, such as flocks of birds, schools of fish, herds of wildebeest, and colonies of bacteria has certain advantages, including avoiding predators, increasing the chance of finding food, saving energy, etc. Inspired by the collective and cooperation behaviours of biology, robot flocking system has became an active research area in the past few years. The flocking system introduced in [1] is investigated by U.S. military for controlling unmanned vehicles. NASA is planning the mission of swarm robots for planetary exploration [2]. Paper [3] discusses the possibility of using a group of nanorobots to kill the cancer tumours within the body.

Many flocking systems have been developed which utilize fuzzy logic to imitate the flocking behaviour. Using fuzzy controller to implement boids [15] system is introduced in Paper [16]. It proves that an agent's decisions can be based purely on unclear evaluations of its environment and linguistic fuzzy rules. But the fuzzy flocking only concerns the alignment, and the relative distances between neighbours are uncontrollable in this system.

Paper [17] describes a fuzzy-based mobile sensor network for locating the hazardous contaminant in an unknown large-scale area. It uses neighbours' relative angles and contaminant concentration value to implement flocking control. That means flocking march direction and relative distances between agents are related to the contaminant distribution in the unknown area.

Both of the achievements mentioned above use local information and fuzzy logic to implement flocking control. Whereas, the flocking march direction and the relative distances between neighbours are uncertain in these systems.

The control behaviour of flocking system is very complex. To keep the relative distance, a threshold should be set to estimate the distance between agents is too close or too far. If the distance between neighbours is larger than the threshold, the neighbours should attract each other; if their distance is smaller than the threshold, there should be a repulsive force between them. Furthermore, the interaction between agents will disappear if their distance is larger than the sensor range.

To implement the distance control between agents, we design a fuzzy force function. Fuzzy Logic controller is an alternative design method which can be applied in developing both linear and non-linear systems. It uses human knowledge and experience to describe a complex systems with only several simple fuzzy rules. We can use a single fuzzy controller to achieve both separation and cohesion behaviours. Moreover, we can easily build a continuous system function by using fuzzy logic. And the system stability can be proved by using LaSalle's invariance principle.

The agent localization is a big issue in the multi-agent system. In most of the flocking system, agents need to know the state information of themselves and their neighbours. To do so, a global coordinate is needed to enable the agents localize themselves in the global coordinate and send the state information to their neighbours. Namely, the flocking system can not work in the environment where has no localization mechanism. Due to the reason, some researchers have start to investigate the flocking control via local information.

Using local information means agents adjust their motion according to the relative state to their neighbours (relative position or relative velocity between neighbours). And they do not need to know any state information in the global coordinate. The paper [10] applies local information to achieve the Vicsek system [11]. Each agent uses vision sensor to measure the relative distances and relative angles to its neighbours. Depending on the local information, agents gradually update their headings to the same direction. Another experiment achievement introduced in [12] demonstrates a distributed multi-agent cyclic pursuit algorithm which uses the relative angles between neighbours. Both of the [11] and [12] concern the agent headings, and agents forward velocities are defined as constant. Multi-agent system introduced in [13] is capable of controlling both the distances and angles between neighbours by using the local information. But the algorithm is designed for robot formation which means the neighbour relationships must be fixed.

It can be seen that, in most of the flocking systems, agents still need to know the global information. Only several flocking systems utilize the local information to control the robot headings. Although local information can be used to control relative distances and relative angles in robot formation, the agents must know which members are their neighbours. And the neighbours relationships must be fixed in the robot formation.

In this paper, a distributed algorithm will be designed to control the flocking system. Different from other multi-agent systems, the neighbour relationships in the flocking are no longer fixed, instead the agents achieve the flocking behaviour by using the local interactions between adjacent flocking members. In other words, there is no constant neighbours relationship between any two agents, whereas a collective behaviour emerges from the interaction between neighbours. Furthermore, the agents can not get any global information. By using the local information, the relative distance between any neighbours asymptotically reaches a specific value.

The outline of this paper is arranged as: Section II briefly introduces the mobile robot model and the concept of hand position. Section III introduces how to use the fuzzy logic controller to design a non-linear force function. Section IV proves that the force function can stabilize the flocking system. In section V, we develop a local form flocking controller which is derived from the global flocking algorithm. Section VI gives the outcomes of simulation. A brief conclusion is given in section VII.

## II. Mobile Robot Model

### A. Robot kinematic model

In a 2D coordinate space, the agent position can be represented by a position vector $q = [q_x, q_y]^T$ where $q_x$ and $q_y$ is position parameter of $X$-axis and $Y$-axis. The agent velocity can be described as $p = \dot{q} = [\dot{q}_x, \dot{q}_y]^T$, where $\dot{q}$ is the differential of $q$ with respect to time. In our flocking system, $\dot{q}$ is considered as the system output, and $q$ is the system state feedback which is used as the system input.

The kinematic model addresses the relationship between the position and the velocity of robots. The robot output can be represented by a vector $u = [v_f, \omega]^T$ where $v_f$ denotes the forward velocity and $\omega$ is the robot rotation velocity. As mentioned above, the output $p$ is the differential of position, which is described as $p = [\dot{q}_x, \dot{q}_y]^T$. The relationship between $p$ and $u$ is:

$$\begin{cases} \dot{q}_x = v_f cos\theta \\ \dot{q}_y = v_f sin\theta \\ \dot{\theta} = \omega \end{cases} \tag{1}$$

where $\theta$ is the heading of robot in the global coordinate.

As a non-linear parameter, $\omega$ is the differential of robot heading. So the robot kinematics is a non-linear system. In the next subsection, we will introduce a "hand position" concept [18] which can convert $\omega$ into linear velocity.

### B. Hand position

We define the velocity of agent $i$ in the direction of robot heading as forward velocity ($v_f$), and the velocity perpendicular to the heading is defined as translation velocity ($v_t$). Normally, robot kinematic model has two outputs: forward velocity ($v_f$) and rotation velocity ($\omega$). To relate the $v_t$ with $\omega$, we introduce a "hand position" concept to convert rotation velocity $\omega$ into the translation velocity $v_t$.

Hand position is a point located at the heading axis with distance $L$ to the center of robot. Fig.1 is the top view of a double-differential mobile robot. The distance between the center of wheel axle (star) and hand position (circle) is $L$, the rotation velocity of robot is $\omega$.

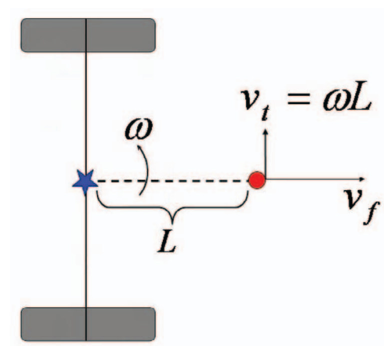


Fig. 1. Top view of a double-differential mobile robot

Taking the hand position as a reference point, the translation velocity of hand position is $v_t = \omega L$. So, the rotation velocity of robot is $\omega = v_t / L$. The output of robot kinematic model can be written as:

$$\begin{aligned} v_f &= \dot{q}_x cos\theta + \dot{q}_y sin\theta \\ v_t &= \omega L = \dot{q}_x cos(\theta + \frac{\pi}{2}) + \dot{q}_y sin(\theta + \frac{\pi}{2}) \end{aligned} \tag{2}$$

where $\dot{q}_x$ and $\dot{q}_y$ are velocities of agent in the direction of $X$-axis and $Y$-axis. It can be seen that, both $v_f$ and $v_t$ are the functions of $\dot{q}_x$, $\dot{q}_y$ and $\theta$. In later part of this paper, we will use the local information to replace $\dot{q}_x$, $\dot{q}_y$, $\theta$. Namely, a local form flocking algorithm will be developed.

## III. Flocking Controller Design

This section introduces the design method for achieving distributed flocking algorithm. To control the relative distances, a force function is used. For the purpose of design and analysis, we still use the global coordinate to describe the 2D space. Later, we will transform our global flocking algorithm into a local form.

Assume that agent $j$ is one of the neighbors of agent $i$. The distance between agent $i$ and agent $j$ is $||q_i - q_j||$. We use $H_s(||q_i - q_j||)$ to denote the potential function between them. The gradient of $H_s(||q_i - q_j||)$ with respect to $||q_i - q_j||$ is called potential force and expressed as $f_s(||q_i - q_j||)$. Because the distance control is a complex behavior, so the fuzzy logic is applied to create this potential force function.

In this fuzzy potential force function, we use 6 input membership functions and 6 output membership functions which are shown in fig.2. The "centroid average" methods is used for defuzzification.

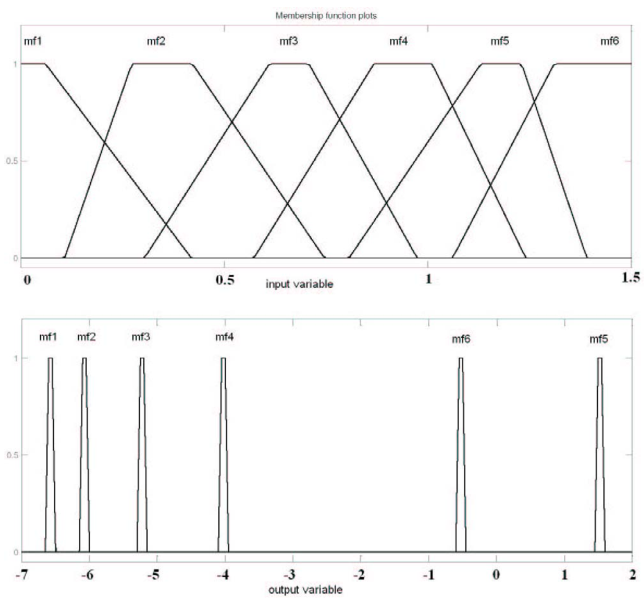


Fig. 2. Fuzzy membership functions

The fig.3 illustrates the force function $f_s(||q_i - q_j||)$ and its corresponding potential function $H_s(||q_i - q_j||)$.

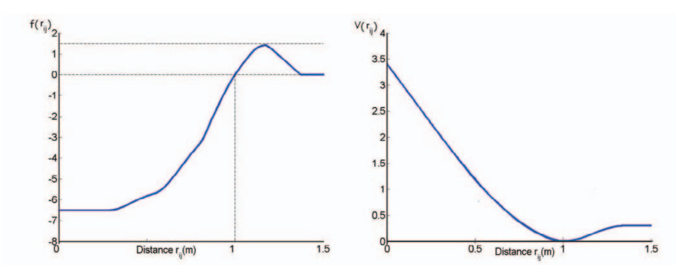

Fig. 3. $f_s(||q_i - q_j||)$ (left) and $H_s(||q_i - q_j||)$ (right)

Furthermore, the $f_s(||q_i - q_j||)$ owns following properties:

- When the distance ($||q_i - q_j||$) between agent $i$ and $j$ is smaller than a specific distance ($d$), $f_s(||q_i - q_j||)$ is negative. Agent $i$ moves away from agent $j$.
- When the distance ($||q_i - q_j||$) between agent $i$ and $j$ is larger than the specific distance ($d$), the $f_s(||q_i - q_j||)$ is positive.
- When the distance ($||q_i - q_j||$)between agent $i$ and $j$ is larger than the communication range ($C$), the $f_s(||q_i - q_j||)$ is zero.
- $f_s(||q_i - q_j||)$ is a continuous function, and $H_s(||q_i - q_j||)$ must be a differentiable function.

We use $\bigtriangledown H_s(||q_i - q_j||)$ to denote the gradient of $H_s(||q_i - q_j||)$ with respect to $q_i - q_j$. The flocking algorithm for the agent $i$ can be written as:

$$\dot{q}_i = - \bigtriangledown H_s(||q_i - q_j||) \tag{3}$$

Because of the mutual relationship between neighbors. The $\bigtriangledown H_s(||q_i - q_j||)$ function satisfies:

$$\bigtriangledown H_s(||q_i - q_j||) = - \bigtriangledown H_s(||q_j - q_i||) \tag{4}$$

Beware that the $\bigtriangledown H_s(||q_i - q_j||)$ is different from $f(||q_i - q_j||)$. Their relationship can be written as:

$$\bigtriangledown H_s(||q_i - q_j||) = f(||q_i - q_j||)\frac{q_i - q_j}{||q_i - q_j||} \tag{5}$$

For the convenience of expression, we use $\bigtriangledown H_s(||q_i - q_j||)$ in the stability analysis, and the $f(||q_i - q_j||)$ will be used to develop a local form flocking algorithm.

We define the set of neighbours of agent $i$ as $N_i$, the flocking algorithm of agent $i$ can be written as:

$$\dot{q}_i = - \sum_{j \in N_i} \bigtriangledown_{ij} H_s(||q_i - q_j||) \tag{6}$$

Because of the mutual relationship between neighbours and (4), we get:

$$\sum_{i=1}^{N} \sum_{j \in N_i} \bigtriangledown_{ij} H_s(||q_i - q_j||) = 0 \tag{7}$$

## IV. System stability analysis

LaSalle's invariance principle can analyze the stability of continuous systems. Comparing with other stability theories, LaSalle's invariance principle is more popular because it focuses on the system behaviours and can be used to analyze non-linear systems. The key of using this stability theory is finding an appropriate Hamiltonian function in the system.

It is clear that, the agent speed depends on the potential function. If the potential energy of flocking system keeps decreasing, the flocking system will asymptotically reach equilibrium. As mentioned in the equation (6), the flocking algorithm is:

$$\dot{q}_i = -\sum_{j \in N_i} \bigtriangledown H_s(||q_i - q_j||) \tag{8}$$

To prove the stability, we must find a proper Hamiltonian function $H(q)$. We use $H(q_i)$ to denote the energy of agent $i$, the relationship between $H(q)$ and $H(q_i)$ is:

$$H(q) = \sum_{i=1}^{N} H(q_i) \tag{9}$$

According to the the definition of potential function which is introduced in the section III, the potential energy of agent $i$ can be written as:

$$H(q_i) = \sum_{j \in N_i} H_s(||q_i - q_j||) \tag{10}$$

We use $\bigtriangledown H(q_i)$ to denote the partial derivative of $H(q_i)$ with respect to $q_i$:

$$\bigtriangledown H(q_i) = \frac{\partial H(q_i)}{\partial q_i} = -\dot{q}_i \tag{11}$$

The differential of $H(q)$ with respect to time is:

$$\dot{H}(q) = \sum_{i=1}^{N} \frac{\partial H(q_i)}{\partial q_i} \cdot \dot{q}_i = \sum_{i=1}^{N} \bigtriangledown H(q_i) \cdot \dot{q}_i = \sum_{i=1}^{N} -\dot{q}_i^2 \tag{12}$$

which means the $\dot{H}(q)$ is non-positive. So all the agents attempt to approach to the stable state:

$$\dot{q}_1 = \dot{q}_2 = \cdots = \dot{q}_n = 0 \tag{13}$$

Namely, the agent velocities equal to zeros at the stable state. In the global coordinate, the agents will stable when:

$$\sum_{j \in N_i} \bigtriangledown H_s(||q_i - q_j||) = 0 \tag{14}$$

which means the distance between any neighbours is equal to the specific distance $d$ where the potential force is zero.

## V. Flocking Control Using Local Information

In this section, we will develope a local form flocking algorithm which enable the agent achieve flocking control by using the local information only.

According to the relationship (5), the flocking algorithm can be written as:

$$\dot{q}_i = -\sum_{j \in N_i} f(||q_i - q_j||) \frac{q_i - q_j}{||q_i - q_j||} \tag{15}$$

Local information, includes relative angle and relative distance between neighbours, has been used in some former achievements ( [10], [13]). Fig.4 illustrates an agent $i$ and one of its neighbours $j$. We use a straight line to link the neighbour. $d_{ij}$ is the length of the link and $\beta_{ij}$ is the angle between the link and the heading of agent $i$. $\theta_i$ is the heading of agent $i$ in the global coordinate.

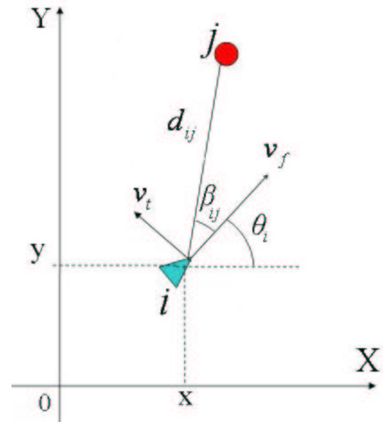


Fig. 4. Local information acquired by agent $i$

The local information can be acquired by agent $i$ is $d_{ij}$ and $\beta_{ij}$. According to the equations (1), (3) and the flocking algorithm (15), the velocity of agent $i$ in the global coordinate can be written as:

$$\begin{aligned} \dot{q}_{ix} &= -\sum_{j \in N_i} f_s(d_{ij}) cos(\beta_{ij} + \theta_i) \\ \dot{q}_{iy} &= -\sum_{j \in N_i} f_s(d_{ij}) sin(\beta_{ij} + \theta_i) \end{aligned} \tag{16}$$

where $\dot{q}_i = [\dot{q}_{ix},\ \dot{q}_{iy}]^T$ is the velocities of agent $i$ in the 2D space. As mentioned in the section II, the velocity of agent $i$ in the direction of heading is forward velocity ($v_f$), and the velocity perpendicular to the heading is defined as translation velocity ($v_t$). Because of (2), the expression of $v_f$ and $v_t$ can be written as:

$$\begin{aligned} v_f &= \dot{q}_{ix} cos\theta_i + \dot{q}_{iy} sin\theta_i \\ &= -\sum_{j \in N_i} f(d_{ij})[cos(\beta_{ij} + \theta_i)cos\theta_i + sin(\beta_{ij} + \theta_i)sin\theta_i] \\ &= -\sum_{j \in N_i} f(d_{ij}) cos\beta_{ij} \\ v_t &= \dot{q}_{ix} cos(\theta_i + \frac{\pi}{2}) + \dot{q}_{iy} sin(\theta_i + \frac{\pi}{2}) \\ &= -\sum_{j \in N_i} f(d_{ij})[-cos(\beta_{ij} + \theta_i)sin\theta_i + sin(\beta_{ij} + \theta_i)cos\theta_i] \\ &= -\sum_{j \in N_i} f(d_{ij})(-sin(-\beta_{ij})) \\ &= -\sum_{j \in N_i} f(d_{ij}) sin(\beta_{ij}) \end{aligned} \tag{17}$$

In the equation (17), both the forward velocity and the translation velocity are functions about the relative distances ($d_{ij}$) and the relative angles ($\beta_{ij}$). Both $d_{ij}$ and $\beta_{ij}$ are local information which can be acquired locally. So this flocking algorithm can be used in any mobile robot which equipped with a local sensor. Because $v_t = \omega L$, we get the local form flocking algorithm:

$$v_f = -\sum_{j \in N_i} f(d_{ij}) cos\beta_{ij}$$
$$\omega = -\frac{1}{L} \sum_{j \in N_i} f(d_{ij}) sin(\beta_{ij}) \quad (18)$$

## VI. Simulation

In the simulation, 200 agents are placed randomly in a 2×2 $m^2$ area at the initial state. We use circles to denote the agents in the flocking and the links describe the neighbour relationships between agents. Because the communication range of each agent is limited, so the links only exist between agents if their relative distance is smaller than a specific distance. In the simulation, the communication range is set as $C = 1.2m$, and the expected distance between agents is $d = 1m$.

Figure 5, 6 and 7 illustrate the network evolution of flocking system under the control of distributed flocking algorithm. The figures show that the flocking network gradually extends. Finally, a stable pattern is formed and all the agents reach a stable state.

figure 8, 9 and 10 give the parameters: connectivity, cohesion radius and the minimal distance among all the neighbours which are used to evaluate the performance of flocking system. It can be seen from the fig.8 that the graph connectivity keeps 1 during the whole process. That means the flocking network keeps connected and the flocking system keeps cohesive. The radius of flocking also reaches a stable level and the minimal distance between neighbours is approaching to $1m$ which is the specific distance.

## VII. Conclusions

This paper introduces a fuzzy based flocking system. A potential function is needed to keep the relative distance between neighbours. And the fuzzy logic controller can be used to design the potential force function. As analyzed in the stability section, the fuzzy logic controller should meet certain stability conditions. The flexibility in the design of fuzzy logic functions provides feasibility to meet the stability conditions. Furthermore, we convert the flocking algorithm into local form which only uses the local information to achieve the flocking control.

In our further work, the flocking algorithm will be tested with real robots (5-6 Pioneer robots or Wifibots). Furthermore, a distributed flocking system which is capable of tracking a dynamic target will be developed. Moreover, the fragmentation phenomena has been observed in the flocking system when the separation force exceeds certain threshold. Our further work will also focus on the connectivity problem of flocking network.

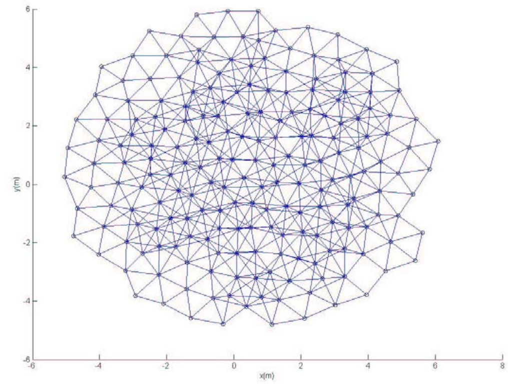

Fig. 5. t=15s

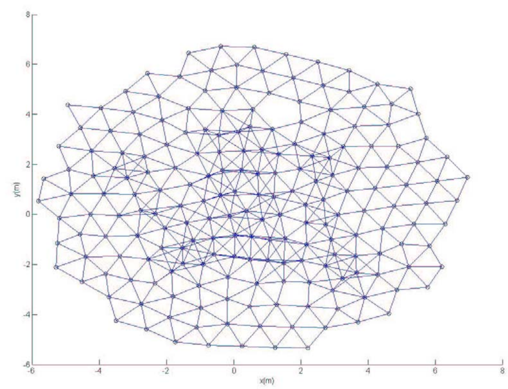

Fig. 6. t=25s

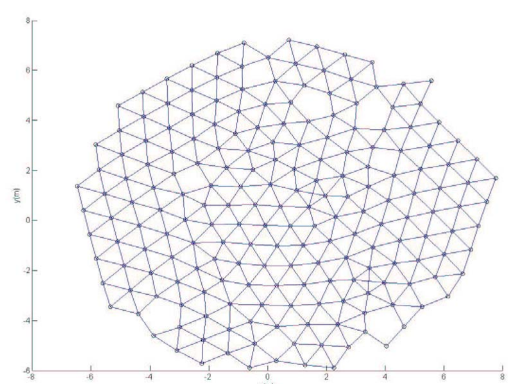

Fig. 7. t=50s

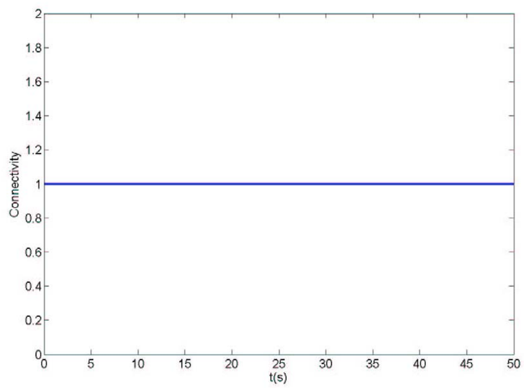


Fig. 8. Network connectivity

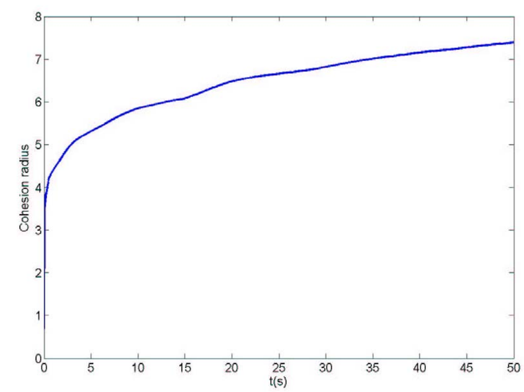


Fig. 9. Cohesion radius

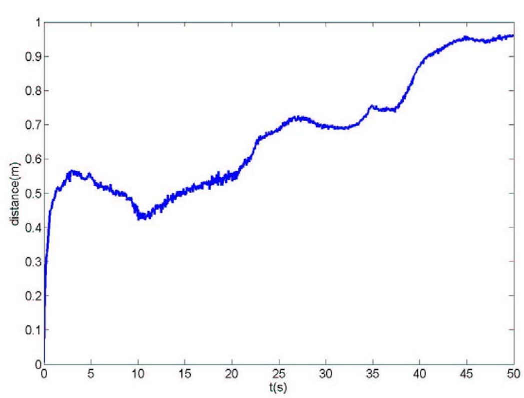


Fig. 10. Minimal distance between agent

# REFERENCES


[1] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," IEEE transactions of robotics and automation, vol.14, pp.926-939, 1998.

[2] M. Ji and M. Egerstedt, "Distributed formation control while preserving connectedness," Proceedings of IEEE conference on decision and control, pp. 5962-5967, 2006.

[3] M. A. Lewis and G. A. Bekey, "The behavioral self-organization of nanorobots using local rules," Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, Vol.2, pp.1333-1338, 1992.

[4] I. D. Couzin, J. Krause, N. R. Franks and S. A. Levin, "Effective leadership and decision-making in animal groups on the move," *Nature*, vol.433, pp.513-516, 2005.

[5] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzer and C. Taylor, "A vision-based formation control framework," IEEE transactions on robotics and automation, vol.18, pp.813-825, 2002.

[6] J. P. Desai, J. P. Ostrowski and V. Kumar, "Modelling and control of formations of nonholonomic mobile robots," IEEE transactions on robotics and automation, vol.17, pp.905-908, 2001.

[7] M. Ji and A. Muhammad and M. Egerstedt, "Leader-based multi-agent coordination: controllability and optimal control," Proceedings of the American control conference, 2006.

[8] W. Wang and J. E. Slotine, "A theoretical study of different leader roles in networks," IEEE transactions on robotics and automation, vol.51, pp.1156-1161, 2006.

[9] D. Dimarogonas, M. Egerstedt and K. J. Kyriakopoulos, "A leader-based containment control strategy for multiple unicycles," Proceedings of the IEEE conference on decision and control, pp. 5968-5973, 2006.

[10] N. Moshtagh, A. Jadbabaie and K. Daniilidis, "Vision-based distributed coordination and flocking of multi-agent systems," Proceedings of conference of robotics: science and systems, Massachusetts Institute of Technology, 2005.

[11] T. Vicsek, A. Czirok, E. Ben Jacob, I. Cohen, and O. Schochet, "Novel type of phase transitions in a system of self-driven particles," Phys. Rev.Lett., vol. 75, pp. 1226-1229, 1995.

[12] J. A. Marshall, T. Fung, M. E. Broucke, G. M. T. D'Eleuterio and B. A. Francis, "Experimental validation of multi-vehicle coordination strategies," Proceedings of the American control conference, vol. 2, pp.1090- 1095, 2005.

[13] M. Karasalo, X. Hu and C. Martin, "Robust formation adaptation for mobile platforms with noisy sensor information", Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, pp.2527-2532, 2006.

[14] M. Egerstedt, X. Hu, and A. Stotsky, "Control of mobile platforms using a virtual vehicle approach," IEEE transactions on automatic control, Vol.46, pp. 1777-1782, 2001.

[15] C. W. Reynolds, "Flocks, herds, and schools: a distributed behavioural model," *Computer Graphics*, vol. 21, pp. 25-34, 1987.

[16] I. L. Bajec, M. Mraz and N. Zimic, "Boids with a fuzzy way of thinking," Proceedings of the ASC international conference, pp.58-62, 2003.

[17] X. Cui, T. Hardin, R. K. Ragade, and A. S. Elmaghraby, "A swarm-based fuzzy logic control mobile sensor network for Hazardous contaminants localization," Proceedings of 2004 IEEE international conference on mobile Ad-hoc and sensor systems, pp.194-203, 2004.

[18] J. R. T. Lawton, R. W. Beard and Brett J. Young , "A decentralized approach to formation maneuvers," IEEE transactions on robotics and automation, vol. 19, pp. 933-941, 2003.