Variable Structure Behavioural Controller for Multi-agent Systems

S.Etemadi [1]
PhD Candidate

H. Kouhi [2]
MSc Student

A. Alasty [3]
Professor

G.R. Vossoughi [4]
Associate Professor

In previous papers authors have considered agents as inertia-less self driven particles and designed a flocking algorithm. Application of this algorithm to agents with considerable inertial characteristics needs a behavioural controller. The controller uses the local information and helps every agent to imitate the desired behaviour as a member of the flocking frame which covers the main issue in this paper. All agents are assumed to possess limited identical influencing/sensing radius. The sliding-mode control technique is used, hence; effect of bounded disturbances and uncertainties can be omitted too. Once inertial agents are equipped with the behavioural controller, the multi-agent system behaves similar to a group of self-driven inertia-less particles which; coordination control algorithms and cohesion analyses are previously designed for.

Keywords: sliding-mode control, behavioural controller, multi-agent, leader following.

1 Introduction

Computer simulations by Brogan [1] showed that the assumption of mutual attractive and repulsive forces between members of a group leads to acceptable social behaviours. Artificial potential fields (APFs) were first introduced for obstacle avoidance by Khatib [2]. Using APFs and through simulations, Reif [3] showed that the quasi static modelling also leads to acceptable social behaviours. Gazi [4] considered a model of quasi-static swarm with massless and dimensionless members in *n* dimensional space and discussed the cohesion and goal convergence. To be more practical Etemadi [5] introduced a gradual decrease in view and later [6] discussed a sharp limitation of sight. Non-holonomic constraints on motion of a group of inertia-less particles are studied by Dimarogonas [7].

Based on the APFs, Kim [8] discussed collision and unreachable goal problems and proposed some frameworks for decentralized control of self-organizing systems. Burger [9], Bıyık [10] and Haddad [11] developed methods for large scale modelling of swarms. Vicsek [12] showed that using the nearest neighbour rule, a group of agents with equal velocities in different directions will finally move in a unified direction. Later Jadbabaie [13] revised this method and strengthened the proof which based on, Olfati-saber [14] combined both graph and energy methods to study the flocking phenomena.

[1] PhD Candidate, School of Mechanical Engineering, Sharif University of Technology, Tehran
[2] MSc Student, School of Mechanical Engineering, Sharif University of Technology, Tehran
[3] Corresponding Author, Professor, School of Mechanical Engineering, Sharif University of Technology, Tehran, Email: *aalasti@sharif.edu*.
[4] Associate Professor, School of Mechanical Engineering, Sharif University of Technology, Tehran

In the area of control and coordination of a multi-agent system (MAS) the variety is even more. Starke [15] used behavioural control systems to produce cohesive or dispersive behaviours and coordinated a group of robots through a field of obstacles. Kim [16] used the PSO method to guide agents. Krishnanand [17] studied clustering of swarms in multiple locations. Olfati saber [14] and Porfiri [18] assumed a virtual leader to coordinate a MAS and discussed the communication stability inside the network. Su [19] generalized the method to multiple virtual leaders.

Leader following is also addressed in literature which is suitable in the absence of central control systems. Liu [20] studied the role of the leader-agent (LA) velocity and members' reaction time delay on cohesion for a line of members. Designing a structure of distributed observers, Hong [21] discussed the active LA following problem in MASs where the leader velocity is not measurable. To guide the MAS and using local information, Etemadi and Vatankhah proposed active LA [22] and fast coordination algorithm [23] respectively.

Large scale flocking and coordination is possible if agents expose quasi-static behavior. Relative distances between agents are supposed to be regulated to their equilibrium values while flocking. Usually every agent in the network uses its distances from neighbors to plan its motion. Many researches include design of flocking algorithms for agents with specific equations of motion like the works by Olfati-saber [14], Porfiri [18], and Su [24]. They have designed flocking algorithms and investigated connectivity of the MAS while flocking. Flocking algorithms of this category are supported by strong mathematical proofs however, their application are restricted to the agents with that specific equations of motion. In a more general point of view, agents can be considered as self driven inertia-less particles. Design of flocking algorithm and connectivity analysis are simplified by this assumption, and a flocking frame is established. Now in order to put every specific kind of agents in the frame, we just have to design a behavioral controller. The behavioral controller helps agents to imitate the quasi-static behavior which is desired while flocking. In this method if the agents are changed then the flocking algorithm is still valid and only the behavioral controller must be redesigned. Gazi [25, 26] is the first who proposed this idea. He designed a sliding-mode controller to overcome the inertial effects for an aggregating swarm with unlimited vision. Later Alasty designed a PID controller [27] and evaluated the results. Etemadi proposed an emotional controller [28] for a same problem (aggregation) where agents possessed limited influencing/sensing radius. In previous papers, authors have proposed control strategies [22] for coordination of MASs where agents were assumed to be self driven inertia-less particles following a LA. They also have discussed cohesion of the MAS beside a LA [6]. These are valid if the agents with higher order dynamics are able to behave as they are expected in a quasi-static frame. The main concern in this paper is to conquer inertial effects of agents with holonomic equation of motion while a MAS is following a LA. Necessity of the behavioral controller can be better understood via the following example. Figures 1 and 2 illustrate motions of three aggregating inertial agents with a badly designed behavioral controller while figure 3 shows performance of a well designed behavioral controller in a similar problem. The agents in this figure perform smooth movements and finally form a regular network. Agents of figure 3 show acceptable behavior as member of the MAS while they are aggregating. In this paper we are going to design a behavioral controller to make them show this behavior as they are following a LA. As a complement and to be more practical, environmental disturbances are also considered which must be overcome by the behavioral controller. Figure 4 shows the schematic location of the behavioral controller inside the behavioral control loop of every agent.

The articulation of this paper is as the following:

In section two we introduce the desired behavior of agents. In section three the problem is analytically discussed and defined in details. In section four we design the controller and in section five illustrative examples are provided.

2 Desired Quasi-static Behavior

A homogenous MAS with at least two agents is considered (M>1). Agents are assumed to be dimensionless with no time delay, which is a normal assumption in this field of studies [3, 4, 14, 24]. Agents move in planar space and their behaviors are affected by two different kinds of objects: other agents and the LA [6]. To be closer to real conditions, here, the range of the agents' vision is limited. For example for the i^{th} agent ($i\in[1,M]$) if an object is farther than a specified distance (named v), then it cannot be detected and have no effect on the motion of the i^{th} agent. The effect of the object is assumed to be a vector parallel to the line connecting its position to the i^{th} agent position. It is named $\Gamma(.)$ and can be calculated through eq. (1):

$$\Gamma(x^i,W,\lambda,r,v)=\begin{cases}(x^i-W).g_r & \left\|x^i-W\right\|<r\\ -(x^i-W)\lambda & r<\left\|x^i-W\right\|<v\\ 0 & \left\|x^i-W\right\|>v\end{cases} \quad (1)$$

where x^i and W represent the position vectors of the i^{th} agent and the object respectively. λ is a positive constant coefficient that attracts the agent to the object and g_r is an even positive definite scalar function which prevents the agent from getting too close to the object. v is the range of view of the agent and r is the radius of the private area. $||x^i-W||$ plays a significant role. If the object is farther than v from the i^{th} agent ($||x^i-W||>v$), then it has no effect on the i^{th} agent. If $||x^i-W||< r$ then the effect of the object is repulsive which prevents collision, and it is attractive if $r <||x^i-W||< v$.

Velocity of the i^{th} agent is supposed to follow eq. (2) [6]:

$$\dot{x}_d^i=\sigma_i+\sum_{j=1}^{M}G_{ij}+\sum_{q=1}^{N}\Delta_q=\Omega\left(C,x^i,\{x^j\},\{z^q\}\right)=\Omega_i \qquad i,j\in[1,M] \quad (2)$$

where $\sigma_i=\Gamma(x^i, C, \beta, r_a, f)$ is a vector function that expresses the effect of the LA located in C, and $G_{ij}=\Gamma(x^i, x^j, \mu, r_r, f_v)$ is the effect of another agent located in x^j. The (repellent) effect of an obstacle located in z^q is expressed by $\Delta_q=(x^i- z^q).g_r$. β and μ are positive constant coefficients that attract the i^{th} agent to the LA and to the j^{th} agent respectively. r_a and r_r respectively represent the radius of the private areas around the LA and agents of the MAS. The i^{th} agent can detect another agent if it is not farther than f_v while it can detect the LA if it is not farther than f. Via proper choose of function g_r we gain control over the inter-agent equilibrium distances which are desired to be approximately equal to r_r in this case. Note that all corresponding symbols ($\{\beta, \mu\}$, $\{r_r, r_a\}$, and $\{f_v, f\}$) may posses same values.

Eq. (2) presents a quasi-static equation of motion that produces acceptable social behaviors [3]. This behavior is similar to natural flocks where viscosity is considerably high and inertia is negligible. It may be doubted whether the use of eq. (2) for a group of autonomous agents is justifiable or not. When agents do not show such behaviors, it is possible to design and equip every agent with a behavioral controller to force it follow the quasi-static behavior which is our main concern in this paper.

3 - Problem Definition

Based on the eq. (2) which simulates the desired behavior of agents, authors have designed a coordination controller. By this controller and using local information, the LA guides the MAS to track a path or toward a specified location [22]. Additionally a real-time Particle Swarm Optimization algorithm is added to this controller to move the MAS as fast as possible [23]. Cohesion of the MAS also is studied [6] which guarantee that the MAS totally follows the LA. Validity of all these algorithms and discussions depends on the quasi-static behavior of agents. Usually a real agent with considerable inertial effects may behave different if it is subjected to the same inputs as in eq. (2). Consider real autonomous agents with inertial equation of motion:

$$I\,\ddot{x}^i + H(x^i, \dot{x}^i) = D^i + U^i \tag{3}$$

where I is inertia matrix, D is disturbance vector, U is actuation vector, and H comprises other terms of the dynamic equation of motion. $i \in [1, M]$ and x^i represents the position of the i^{th} agent. If we set $U^i = \Omega_i$ which is defined in eq. (2) then even in the absence of any disturbances, drastic vibrating behaviors similar to figures 1 and 2 are observed which are not acceptable.

Proper design of U can help autonomous agents behave as they are expected. In other words it is possible to design and equip every agent with a behavioral controller which uses local information and gives it the ability to move and behave as a quasi-static agent. Design of such controller is previously discussed by Gazi [25, 26] and Etemadi [27] for aggregating swarms and here we are going to solve the problem for the leader following case.

4 - Behavioral Controller Design

To have the same form of equation as eq. (2), eq. (3) can be rewritten as:

$$\ddot{x}^i + h^i = d^i + u^i \tag{4}$$

where $h^i = h(x^i, \dot{x}^i) = I^{-1}H(x^i, \dot{x}^i)$, $d^i = I^{-1}D^i$ and $u^i = I^{-1}U^i$. Disturbance $d=[d_x, d_y]^t$ contains uncertainty and the certain term is $\hat{d} = [\hat{d}_x, \hat{d}_y]^t$. The uncertainty is bounded and its maximum value is δ. In other words $\|d-\hat{d}\|<\delta$, or $|d_x-\hat{d}_x|<\delta$ and $|d_y-\hat{d}_y|<\delta$.

We are going to design a controller to make autonomous agents with the above equation of motion follow behaviors produced from the model in eq. (2). This controller is named *behavioral controller*. So the behavioral controller uses local information around an agent; which has an equation of motion similar to eq. (3), and forces it to behave similar to $\dot{x}_d^i = \Omega_i$. Sliding mode control method is used. Therefore we are able to overcome disturbances too. Sliding mode variable of interest $s=[s_x, s_y]^t$ is defined in eq. (5). In definition of s we are inspired by Gazi [25].

$$s = \dot{x}^i - \dot{x}_d^i = \dot{x}^i - \Omega\left(C, x^i, \{x^j\}\right) = \dot{x}^i - \Omega_i \tag{5}$$

where Ω_i is defined in eq. (2). Consider a Lyapunov candidate function V in terms of s:

$$V = \frac{1}{2}\|s\|^2 = \frac{1}{2}s^T s \tag{6}$$

The control law should be defined to make $\dot{V}$ definitely negative.

Derivative of s is:

$$\dot{s} = \ddot{x}^i - \dot{\Omega}_i = -h^i + d^i + u^i - \dot{\Omega}_i = -h + d + u - \dot{\Omega} \tag{7}$$

For simplicity index i is ignored. While there is no uncertainty ($d=\hat{d}$), s=0 corresponds to the movement of the system on sliding surface. Therefore $\hat{u}$ can be obtained as:

$$\hat{u} = h - \hat{d} + \dot{\Omega} \tag{8}$$

Uncertainty may cause the system to deviate from the sliding surface. So the control law needs an additional compensating term which brings the system back to sliding surface:

$$u = \hat{u} - k\left[sign(s_x), sign(s_y)\right]^t \tag{9}$$

where k>0 is the controller gain.

In order to prove that V is a Lyapunov function we must analyze its derivative. Using eqs. (7), (8) and (9) derivative of V is obtained as the following:

$$\begin{aligned}\dot{V} &= \dot{s}^t s = \left\{-h + d + u - \dot{\Omega}\right\}^t s = \left\{-h + d + \hat{u} - k\left[sign(s_x), sign(s_y)\right]^t - \dot{\Omega}\right\}^t s \\ &= \left[d - \hat{d}\right]^t s - k\left\{sign(s_x)s_x + sign(s_y)s_y\right\} = \left\{\left(d_x - \hat{d}_x\right)^t s_x + \left(d_y - \hat{d}_y\right)^t s_y\right\} - k\left\{|s_x| + |s_y|\right\} \\ &\le \left\{\left|d_x - \hat{d}_x\right| - k\right\}|s_x| + \left\{\left|d_y - \hat{d}_y\right| - k\right\}|s_y| \quad \le \{\delta - k\}\left(|s_x| + |s_y|\right)\end{aligned} \tag{10}$$

Derivative of V is supposed to satisfy relation (11):

$$\dot{V} \le -\eta\left(|s_x| + |s_y|\right) \tag{11}$$

which yields the lower limit of k as:

$$k \ge \delta + \eta \tag{12}$$

δ is introduced in equation (4). η is a positive constant that determines the change rate of V. k comprises two parts: δ and η. Value of δ is dictated by the physical conditions of agents and the environment while the value of η can be adjusted to obtain acceptable control performance.

Note that the sliding mode behavioral controller which is presented in eq. (9) needs no global information. Indices x and y indicate two local orthogonal axes.

5 - Simulation Examples

To verify performance of the behavioral controller three simulation examples are presented. A MAS including 14 agents are following a LA. Parameters of the MAS are shown in Table 1. Every agent is assumed to posses a 10*Kg* inertia and no energy dissipating term (h=0). Range of view of every agent covers a circular area with 1.8*m* radius and they are desired to move with approximately 1.5*m* inter-member distance. It means that agents have an identical 1.8*m* sensing/influencing radius and they will be repelled from each other if they get closer than 1.5*m*. The LA is moving with constant velocity parallel to X axis. Every agent is assumed to have 2nd order equation of motion. MATLAB/SIMULINK toolbox is used to produce simulations. A block-diagram similar to Fig. 4 is built inside the SIMULINK environment. The popular numerical simulation method of dynamic systems is used. Using initial values of positions and velocities in every simulation step, the control signal and subsequently acceleration vector of every agent is calculated. Integration of the acceleration vector yields velocity and position vectors of the agent which can be used as initial values for the next simulation step. Simulation step time is set to 0.01 sec in all simulations.

In the first simulation agents are initially distributed over the area. They are supposed to form a regular network and simultaneously follow the LA. From the motion paths of agents in Fig. 5 no bouncing movements can be observed which means that agents have successfully imitated the quasi-static behavior. The best option for evaluating the results is a previous work by Etemadi [28] where an emotional learning based intelligent controller is used for behavioural control of six aggregating agents. Motion path of agents are shown in Fig. 3 which seem similar to the behaviour of agents of our simulation in Fig. 5. Since the main discussions in these two cases are different (aggregation and leader following) results can only be qualitatively compared. Similarity of agents aggregating motions in these cases is a verification of the performance of the sliding mode behavioural controller. Control signal of the sliding mode behavioral controller is shown in Fig. 6 for one of the agents. The signal illustrates acceptable control performance. According to the figure it takes approximately ten seconds for every agent to move to its equilibrium position in the network. Agents have formed a completely regular network in this simulation and the maximum control signal value is about 20N which upon we conclude that controller gains are well adjusted.

In the second simulation initial conditions of agents and motion of the LA are similar to the first simulation. But in this case agents are supposed to move through obstacles. Two obstacles are assumed. Fig. 7 shows that the agents form a network which is not as regular as Fig. 5. This is not unacceptable. Agents only regulate their relative distances and the hexagonal pattern in the network topology is the structure with the lowest total energy. There are other patterns which may locally minimize the network energy. There is a pentagonal pattern in the network topology in Fig. 7 which corresponds to such a local minima. Inter-member distances are shown in Fig. 8. Since agents have to move through and over the obstacles they form their network with a delay. The obstacles are blocking the MAS so the LA moves slower in this simulation to give agents more time to follow. Therefore the control signals of this simulation which are shown in Fig. 9 are weaker in this simulation. Fig. 8 shows that inter-member distances have not exactly converged to the same value. Since the MAS is moving, inter-member distances are a little larger in the direction of the LA movement and they are a little smaller in the lateral direction.

The third simulation is totally similar to the second simulation except that in the third simulation disturbance is imposed on agents. The disturbance term is assumed to be randomly dependent on velocity, i.e. $D^i = \left[\alpha_x, \alpha_y\right]^T \dot{x}^i$, while α_x and α_y are real random digits and $\alpha_x, \alpha_y \in [1,5]$. Figures 10 and 11 show the successful performance of the controller in presence of disturbance. Fig. 10 shows that the agents form a regular network while following the LA. Inter-member distances are shown in Fig. 11. Similar to the second simulation (Fig. 8) inter-member distances in the direction of motion are little above their equilibrium value while in the lateral direction they are a little less than their equilibrium value. Value of the disturbance is shown in Fig. 12. Controller tries to omit the effect of the disturbance in agents' behavior. So in this case the control signal which is shown in Fig. 12 is not as smooth as the control signal of the second simulation (Fig. 9). In this case disturbance makes the controller produce considerable outputs even after the network of agents has obtained its regular topology. Comparison of figures 9 and 11 shows that in the absence of disturbance and after the network topology is formed and fixed, the control signal value considerably decreases. Therefore disturbance may cause considerable loss of energy and decrease the power-source life time.

6 - Conclusion

Behavioural control of a group of autonomous agents is discussed analytically in this paper. Agents are assumed to have second order dynamics with considerable inertial effects. Agents are supposed to access their local information. A sliding mode controller is designed which uses the local information. Although the flocking algorithm is designed for a group of inertia-less self driven members, however; simulations show that by using the behavioral controller, while flocking every agent successfully behaves as it is expected.

Contrary to similar works by Gazi [25, 26], sharp limitation of the sensing/influencing radius is considered in this paper, and results for aggregation and leader following are acceptable. Even during moving through obstacles, the agents do not show any violent movements and maintain their network connectivity which means that they have successfully imitated the desired behaviour as a member of a flocking MAS. Due to the robustness of the sliding mode control technique, disturbance is easily manipulated by the behavioural controller. Results will be better and control signals will be smother if energy dissipating terms of the agents' equation of motion are considered too.

Olfati-saber [14], Porfiri [18], and Su [24] provided stable flocking algorithms for second order inertial agents too, but inter-member interactions in those algorithms are different from ours and we cannot quantitatively compare results. Differences in interactions and models lead to differences in time responses and no quantitative comparison will be possible. However; as the main contribution of the paper we proved that it is possible to ignore dynamics of agents while designing the flocking algorithms and qualitatively we obtained the same performance as Olfati-saber [14], Porfiri [18], and Su [24]. Considering agents' dynamics complicates the design of the flocking algorithm and stability investigations. We proved that agents' dynamics can be considered at the final stage where their behaviours can be changed to the desired flocking behaviour. Gazi [25, 26] followed the same idea for an aggregating swarm but he assumed that agents are able to communicate within large distances. We proved the possibility of using a behavioural controller for the leader following flocking case.

Second order holonomic equation of motion represents a simple form of agents' dynamics. The importance of the behavioural controller will be better demonstrated if it is applied to agents with more complicated equations of motion. Future works include non-holonomic constraints on motion of inertial agents which is under active study by authors. Another potential application of this method is coordination of heterogeneous MASs by a predesigned flocking algorithm.

References

[1] Brogan, D.C., and Hodgins, J.K., "Group Behaviors for Systems with Significant Dynamics," Autonomous Robots, Vol. 4, pp. 137-153, (1997).

[2] Khatib, O., "Real Time Obstacle Avoidance for Manipulators and Mobile Robots", International Journal of Robotics Research, Vol. 5, No. 1, pp. 90- 99, (1986).

[3] Reif, J.H., and Wang, H., "Social Potential Fields: A Distributed Behavioral Control for Autonomous Robots", Robotics and Autonomous Systems, Vol. 27, pp. 171-194, (1999).

[4] Gazi, V., and Passino, K.M., "Stability Analysis of Social Foraging Swarms", IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 34, No. 1, pp. 539-557, (2004).

[5] Etemadi, S., Alasty, A., and Vossoughi, G.R., "Stability Analysis of Robotic Swarm with Limited Field of View", Proceeding of the ASME International Mechanical Engineering Congress and Exposition, Washington, USA, pp. 175-184, (2007).

[6] Etemadi, S., Alasti, A., and Vossoughi G.R., “Stability Investigation of a Robotic Swarm with Limited Field of View”, Proceedings of IFAC World Congress, Seoul, Korea, Vol. 17, No. 1, pp. 10794-10799, (2008).

[7] Dimarogonas, D.V., Kyriakopoulos, K.J., “A Connection Between Formation Infeasibility and Velocity Alignment in Kinematic Multi-agent Systems”, Automatica, Vol. 44, No. 10, pp. 2648-2654, (2008).

[8] Kim, D.H., Wang, H., and Shin, S., “Decentralized Control of Autonomous Swarm Systems Using Artificial Potential Functions: Analytical Design Guidelines”, Journal of Intelligent and Robotic Systems, Vol. 45, pp. 369–394, (2006).

[9] Burger, M., Capasso, V., and Morale, D., “On an Aggregation Model with Long and Short Range Interactions”, Nonlinear Analysis: Real World Applications, Vol. 8, pp. 939–958, (2007).

[10] Bıyık, E., and Arcak, M., “Area Aggregation and Time-scale Modeling for Sparse Nonlinear Networks”, Systems and Control Letters, Vol. 57, pp. 142–149, (2008).

[11] Haddad, W.M., and Hui1, Q., “Complexity, Robustness, Self-organization, Swarms, and System Thermodynamics”, Nonlinear Analysis: Real World Applications, Vol. 10, pp. 531–543, (2009).

[12] Vicsek, T., Czirok, A., Jacob, E. B., Cohen, I., and Schochet, O., “Novel Type of Phase Transitions in a System of Self-driven Particles”, Physical Review Letters, Vol. 75, pp. 1226-1229, (1995).

[13] Jadbabaie, A., Lin, J., and Morse, A.S., “Coordination of Groups of Mobile Autonomous Agents using Nearest Neighbor Rules”, IEEE Transactions on Automatic Control, Vol. 48, No. 6, pp. 988-1001, (2003).

[14] Olfati Saber, R., “Flocking for Multi Agent Dynamic Systems: Algorithms and Theory”, IEEE Transactions on Automatic Control, Vol. 51, No. 3, pp. 401-420, (2006).

[15] Starke, J., Kaga, T., Schanz, M., and Fukuda, T., “Experimental Study on Self Organized and Error Resistant Control of Distributed Autonomous Robotic Systems”, International Journal of Robotic Research, Vol. 24, No. 6, pp. 465-486, (2005).

[16] Kim, D.H., and Shin, S., “Self-organization of Decentralized Swarm Agents Based on Modified Particle Swarm Algorithm”, Journal of Intelligent and Robotic Systems, Vol. 46, pp. 129–149, (2006).

[17] Krishnanand, K.N., and Ghose, D., “Theoretical Foundations for Rendezvous of Glowworm-inspired Agent Swarms at Multiple Locations”, Robotics and Autonomous Systems, Vol. 56, pp. 549–569, (2008).

[18] Porfiri, M., Roberson, D.G., and Stilwell, D.J., “Tracking and Formation Control of Multiple Autonomous Agents: A Two-level Consensus Approach”, Automatica, Vol. 43, pp. 1318–1328, (2007).

[19] Su, H., Wang, X., and Yang, W., “Flocking in Multi-agent Systems with Multiple Virtual Leaders”, Asian Journal of Control, Vol. 10, No. 2, pp. 238-245, (2008).

[20] Liu, Y., Passino, K.M., and Polycarpou, M., “Stability Analysis of M Dimensional Asynchronous Swarms with a Fixed Communication Topology”, IEEE Transactions on Automatic Control, Vol. 48, No. 1, pp. 76-95, (2003).

[21] Hong, Y., Chen, G., and Bushnell, L., “Distributed Observers Design for Leader-following Control of Multi-agent Networks”, Automatica, Vol. 44, pp. 846–850, (2008).

[22] Etemadi, S., Alasty, A., and Vossoughi, G.R., “Nonlinear Tracking Control of Robotic Swarm,” Proceedings of ISME International Mechanical Engineering Conference (CD), Kerman Iran, Paper No. 1678, (2008).

[23] Vatankhah, R., Etemadi, S., Honarvar, M., Alasty, A., Boroushaki, M., and Vossoughi, G.R., “Online Velocity Optimization of Robotic Swarm Flocking using Particle Swarm Optimization (PSO) Method,” Proceedings of ISMA09 Conference (CD), Sharjah, UAE, Paper No. 56-14717, DOI. 10.1109/ISMA.2009.5164776, (2009).

[24] Su, H., Wang, X., and Chen, G., “A Connectivity-preserving Flocking Algorithm for Multi-agent Systems Based only on Position Measurements,”. International Journal of Control, Vol. 82, No. 7, pp. 1334-1343, (2009).

[25] Gazi, V., “Swarm Aggregations using Artificial Potentials and Sliding Mode Control,” IEEE Transactions on Robotics, Vol. 21, No. 6, pp. 1208-1214, (2005).

[26] Gazi, V., Fidan, B., Hanay, Y. S., and Koksal, M. I., “Aggregation, Foraging, and Formation Control of Swarms with Non-holonomic Agents using Potential Functions and Sliding Mode Techniques,” Turkish Journal of Electrical Engineering, Vol. 15, No. 2, pp. 149-168, (2007).

[27] Alasty, A., Etemadi, S., and Roshan-Ghalb, F., "Behavioral Control of Autonomous Swarms," Proceedings of ISME International Mechanical Engineering Conference (CD), Kerman Iran, Paper No. 2758, (2008).

[28] Etemadi, S., Vatankhah, R., Alasty, A., and Vossoughi, G.R., ”Swarm Aggregation Using Emotional Learning Based Intelligent Controller,” Proceedings of ISMA09 Conference (CD), Sharjah, UAE, Paper No. 56-69446, DOI. 10.1109/ ISMA, 5164827, (2009).

Nomenclature

C :	Position of the LA
D :	Disturbance matrix
d:	Mapped disturbance matrix
$\hat{d}$:	Certain part of disturbance matrix
f:	The range the LA can be detected
f_v:	The range an agent can be detected
G_{ij}:	Effect of an object on an agent velocity
g_r:	Repellent effect of an object
H:	Nonlinear terms of an agent
h:	Mapped nonlinear terms of an agent

I :	Inertia matrix of an agent
k :	Controller gain
M:	Number of agents
r:	Radius of private area around an object
r_a:	Radius of private area around the LA
r_r:	Radius of private area around an agent
s :	Sliding-mode variable of interest
U :	Control signal
u^i:	Mapped control signal
$\hat{u}$:	A part of u^i
V:	Lyapunov function
W:	Position of an object
x^i	Position of an agent
z^q	Position of an obstacle

Greek Symbols

β:	Attraction coefficient for the LA
Δ:	Repellent effect of an obstacle
δ :	Maximum uncertainty of d_i
Γ:	Effect of an object on an agent velocity
η :	Controller gain
λ :	Attraction coefficient
μ :	Attraction coefficient for an agent
ν :	Range an object can be detected
σ_i :	Effect the LA on an agent velocity
Ω_i :	Desired quasi-static behaviour

Table

Table 1 Values of Parameters used in Simulations

μ	β	f_v	f	r_r	r_a	η	I
3.2 /s	*3/s*	*1.8 m*	*1.8m*	*1.5m*	*1.5m*	1	*10Kg*

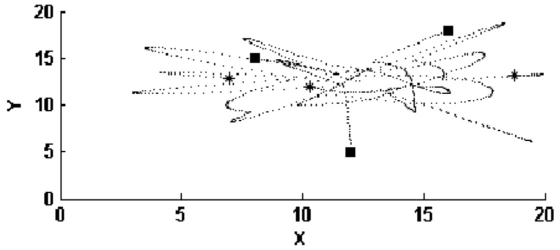

Figure 1- 300 seconds simulation of motion of 3 inertial agents with badly designed behavioral controller in XY plane. Inertial effects of agents produce vibrating behaviors which are totally different from the desired quasi-static behavior. (■: initial positions, *: agents' positions after 300s, dotted trajectories: agents' motion path) [27].

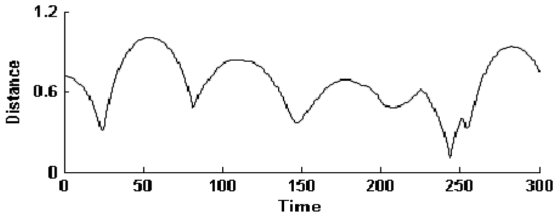

Figure 2 Distance from the geometrical center vs. time for one of the agents of the simulation in Fig.1 [27].

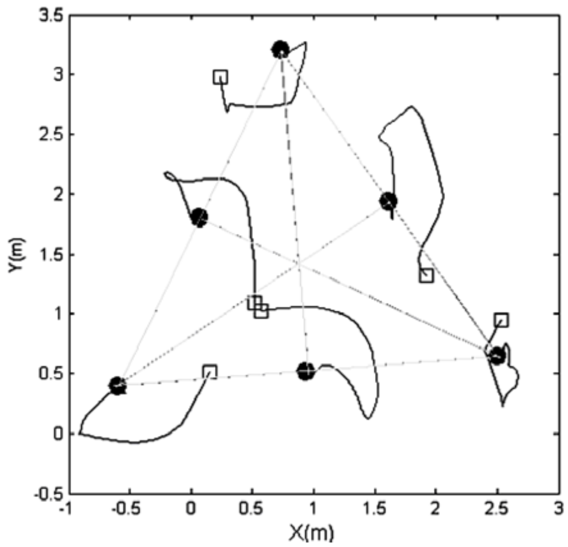

Figure 3 Six inertial agents have been able to behave as desired quasi-static inertia-less members. (□: initial agents' position, ●: final position of agents, solid line: motion path of every agent) [28].

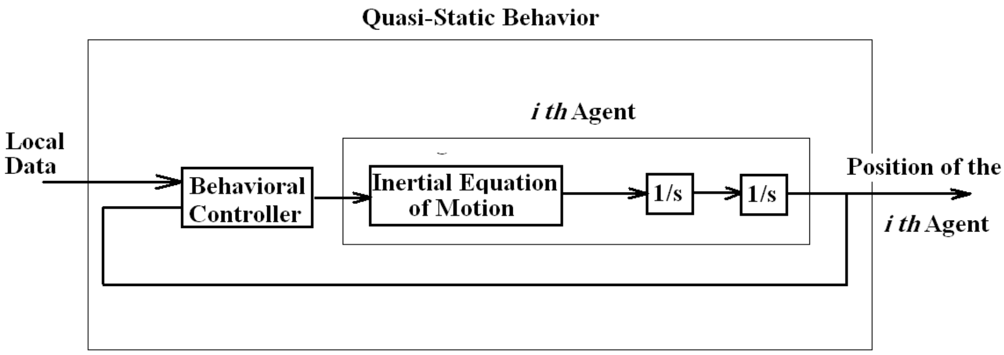

Figure 4 Schematic block-diagram of the closed loop control system on every agent. The same model is used for simulation in MATLAB/SIMULINK environment.

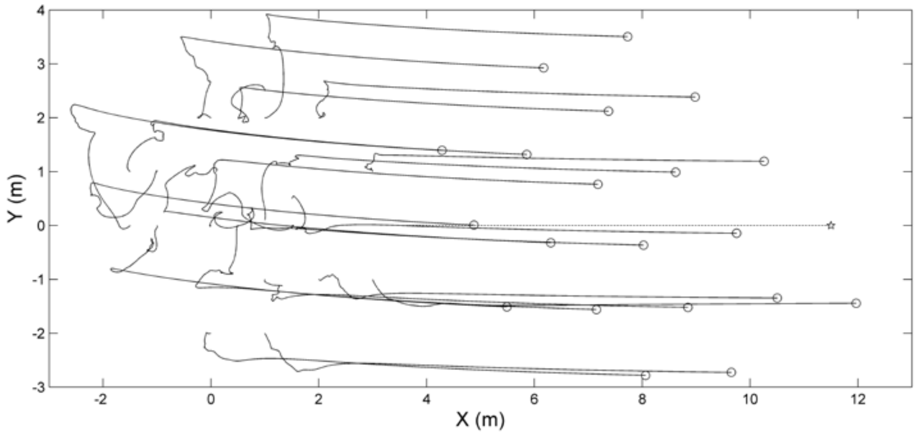

Figure 5 Aggregation of a MAS and following the LA. Inertial agents form a regular network while they are following the LA. (star: LA, dotted line: motion path of the LA, circles: positions of agents at the moment, continues lines: motion path of agents)

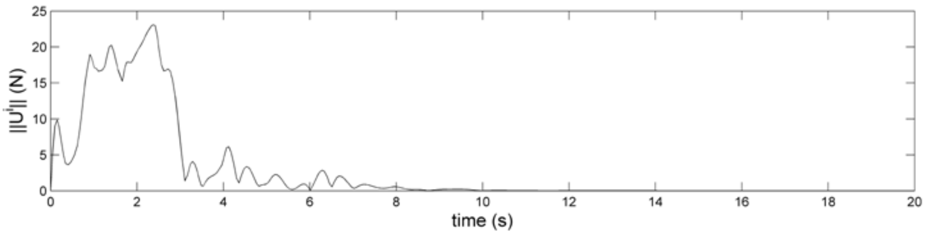

Figure 6 Control signal value generated by the sliding mode controller for one of the agents. The graph corresponds to the simulation shown in Fig. 5.

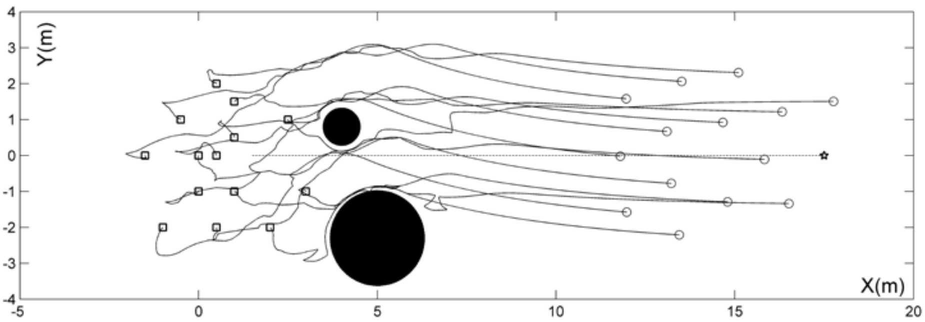

Figure 7 While moving through obstacles and following the LA, inertial agents form a regular network. (dark circles: obstacles, squares: initial positions of agents. See descriptions of Fig. 5 about other symbols)

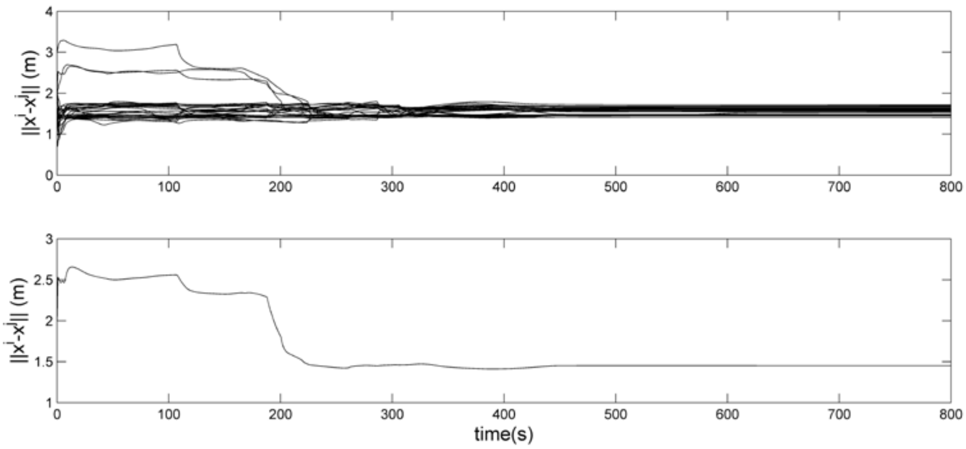

Figure 8 Inter-member distances between agents. One of the graphs is duplicated to help clear observation. The graph corresponds to the simulation shown in Fig. 7.

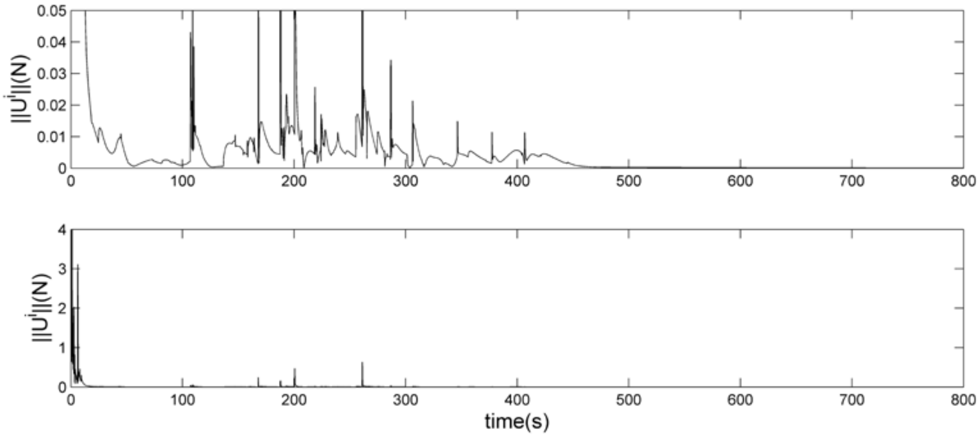

Figure 9 Control signal value generated by the sliding mode controller for one of the agents. The graph is shown in two different scales to help clear observation. The graph corresponds to the simulation shown in Fig. 7.

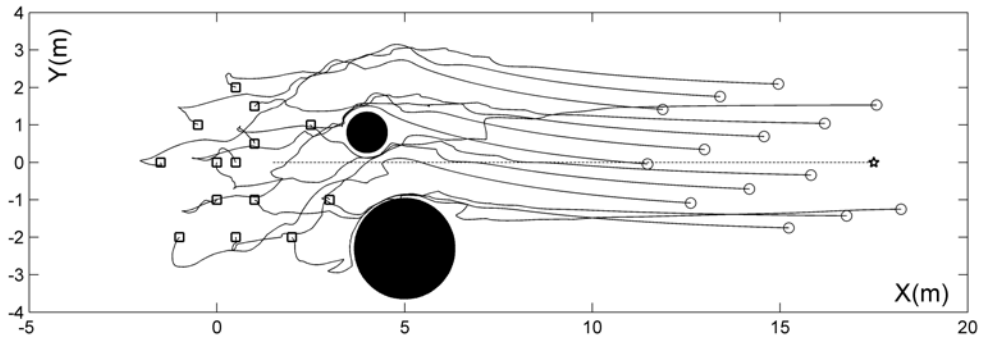

Figure 10 While moving through obstacles and following the LA, inertial agents form a regular network.
In this simulation disturbance is imposed on agents (See descriptions of Fig. 7 about symbols)

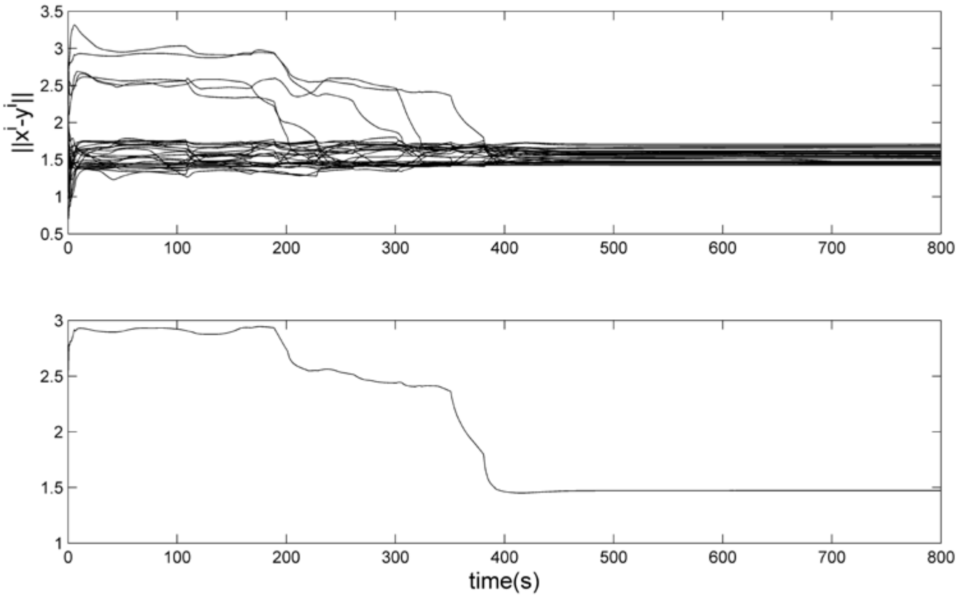

Figure 11 Inter-member distances between agents. One of the graphs is duplicated to help clear observation. The graph corresponds to the simulation shown in Fig. 10.

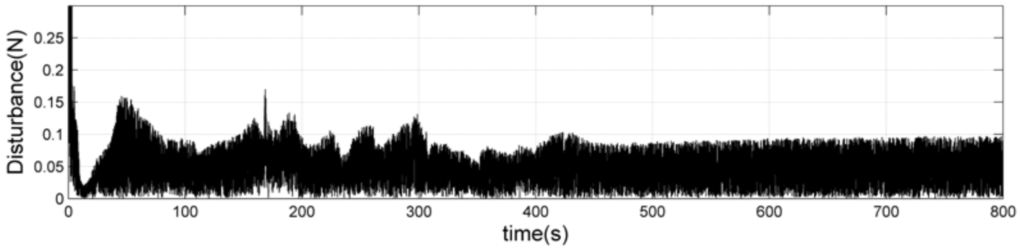

Figure 12 Value of the disturbance imposed on one of the agents (for the simulation shown in Fig. 10).

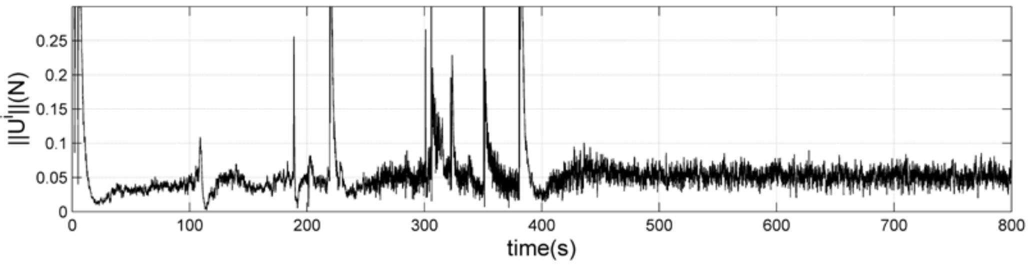

Figure 13 Control signal value generated by the sliding mode controller for one of the agents while disturbance is imposed on agents. The graph corresponds to the simulation shown in Fig. 10.

چکیده

در مقالات قبلی با در نظر گرفتن هر عضو به صورت یک ذره خود-محرک بدون اینرسی, الگوریتم حرکت جمعی یک مجموعه چند رباتی طراحی شد. یک کنترلر رفتاری می تواند اعضای دارای اینرسی قابل توجه را به پیروی از این الگوریتم حرکت جمعی وادار کند. کنترلر از اطلاعات محلی هر عضو استفاده کرده و او را قادر می سازد تا رفتار مطلوب در حرکت جمعی را تقلید کند که بحث اصلی این مقاله را شامل می شود. تمام اعضا دارای حوزه تأثیر و تشخیص یکسان و محدود هستند. استفاده از روش کنترل مود لغزشی امکان حذف اثر اغتشاشات کرانه دار را نیز میسر کرده است. در صورت تجهیز هر عضو دارای اینرسی به کنترلر رفتاری، مجموعه اعضا، رفتاری مشابه یک مجموعه از ذرات خود- محرک بدون اینرسی از خود بروز خواهند داد که الگوریتم حرکت جمعی آنها قبلاً طراحی و پایداری آن اثبات شده است.