The Eighteenth International Symposium on Artificial Life and Robotics 2013 (AROB 18th '13),
Daejeon Convention Center, Daejeon, Korea, January 30-February 1, 2013

Flocking control of multi-agent systems in a limited space

Sharu Jiang, Yingmin Jia, Shichen Long

The Seventh Research Division and the Department of Systems and Control, Beihang University (BUAA), Beijing 100191, China (e-mail: jsrshark@126.com, ymjia@buaa.edu.cn, inuyashalsc@hotmail.com).

Abstract: In this paper, flocking control in a limited space is considered. First, a new conception of safety-value is proposed to evaluate the safety between agents and obstacles in a limited space. Then, a new distributed flocking control protocol called the limited space flocking (LSF) algorithm is designed so as to extend the Olfati-Saber's control protocol to the case of a limited space. The algorithm utilizes control protocols corresponding to the safety-value, so the multi-agent system can automatically change its velocity and structure to pass the limited space both quickly and safely. Finally, simulation results show that the proposed algorithm can greatly improve the average velocity of systems and enhance the safety-value in a limited space.

Keywords: flocking, limited space, LSF algorithm, multi-agent, safety-value

1 INTRODUCTION

Recently, multi-agent systems have drawn increasing attention and many good results on the control protocol design have been obtained in a series of works [1]-[6]. An important issue in the control of multi-agent systems is to design a control protocol in order to achieve flocking behavior of systems. A model introduced by Reynolds in [4] plays an important role in the study of flocking. Reynolds' three heuristic rules (flock centering, collision avoidance and velocity matching [7]) led to the first computer animation of flocking. Since then, there emerge lots of works dealing with the flocking problems [8]-[13]. Particularly, among these works the algorithm proposed by Olfati, Saber [8] had a long-term influence on the later development of the flocking control protocol design because they provided a theoretical and computational framework for design and analysis of scalable flocking algorithms.

Flocking algorithms have wide applications such as self-assembly of connected mobile networks; massive distributed sensing using mobile sensor networks; performing military missions and so on. In some practical cases, for example the control of the air traffic, the space where a system forms flocking is limited. In these situations, by using most algorithms [8]-[13] which may work well in free space, system usually couldn't achieve effective obstacle avoidance and would greatly slow down the average speed. Under the consideration that both the obstacle avoidance and the speed of a system are important standards, in this paper a safety-value is raised to evaluate the safety between agents and obstacles in a limited space and a LSF (Limited Space Flocking) algorithm, which is developed from Olfati-Saber's algorithm [8], is proposed to make the system to pass through a limited space both safely and quickly.

2 PRELIMINARIES

In this section, we provide some basic concepts in graph theory [14]-[16], algebraic graph theory [17], spatially induced graphs (or proximity nets) [18] and make the preparation for us to introduce the LSF algorithm which will be further discussed in the following section.

Consider the following "boids" model of N agents:

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

where q_i is the displacement of agent i, p_i is its velocity, u_i is the behavior rule of agent i

Definition 1 (detection shell [19]). *The detection shell Ω_i of agent i is a region, within which the agent can sense the relative location of neighboring agents.*

we assume that all the Ω_i have the same radius R. Denote the set of agents contained in Ω_i by $N_i(t)$, which can be given as

$$N_i(t) = \{j : \|q_i - q_j\| \leq R\}, \tag{2}$$

where $\|q_i - q_j\|, i, j = 1, 2, \ldots, N$ is the relative position vectors.

Definition 2 (neighboring graph [19]). *The neighboring graph, $G = (V, E)$, is an undirected graph consisting of:*

1. *A set of vertices (nodes), $V = \{1, 2, \ldots, n\}$, indexed by the agents in the group, and*
2. *A set of edges, $E = \{(i, j) \in V \times V | i \in N_j(t) \text{ or } j \in N_i(t)\}$*

Definition 3 (adjacency matrix). *The Adjacency matrix is a matrix $A = [a_{ij}]$ with nonzero elements satisfying the property $a_{ij} \neq 0 \Leftrightarrow (i, j) \in E$, with its elements:*

$$a_{ij} = \rho_k(\|q_j - q_i\|/R) \in [0, 1], \quad j \neq i. \tag{3}$$

where $\rho_k(z)$ is a scalar function which smoothly varies between 0 and 1 [20], and $h \in (0, 1)$ is a constant parameter, the fomula of $\rho_k(z)$ is shown below:

$$\rho_k(z) = \begin{cases} 1 & , z \in [0, h) \\ \frac{1}{2}[1 + cos(\pi\frac{(z-h)}{(1-h)})] & , z \in [h, 1] \\ 0 & , \text{otherwise} \end{cases} \tag{4}$$

© ISAROB 2013

The use of an indicator bump function leads to an adjacency matrix with 0-1 position-dependent elements.

Definition 4 (α-lattice [1]). *An α-lattice is a configuration satisfying the set of constraints in*

$$\|q_j - q_i\| = d, \quad \forall j \in N_i(t). \tag{5}$$

In reference [8], the conception of virtual agents and a classification were first proposed. Corresponding controls can be given according to different kinds of agents. The physical agent with dynamics $\ddot{q}_i = u_i$ is called an α-agent. The primary objective of α-agent in a flock is to form α-lattice with its neighboring α-agents. In nature, α-agents correspond to birds, bees, fish, and ants. Later, virtual agents β-agents and γ-agents are also introduced, which model the effect of "obstacles" and "collective objective" of a group, respectively. γ-agent can be also called the virtual leader of the system.

The distributed control given by Olfati-Saber [8] is denoted as u_i^{OS} and shown in (6):

$$u_i^{OS} = u_i^{\alpha} + u_i^{\beta} + u_i^{\gamma} \tag{6}$$

where the u_i^{α}, u_i^{β} and u_i^{γ} are the controls related to α-agents, β-agents and γ-agents respectively. The stability of the Olfati-Saber algorithm is proved in [8]. Extended from its theoretical frame, a LSF control protocol is designed to deal with the flocking in a limited space.

3 LSF CONTROL PROTOCOL

In this section we present the LSF control protocol in order to enhance the safety-value and the average speed of systems in limited space. First of all, an index is needed to quantitively describe the safety status of the system. Therefore a new conception called the safety-value should be proposed to evaluate the saftey of agents and obstacles in a limited space.

3.1 Safety-value of system in limited space

When passing through a limited space, a multi-agent system will get greatly compressed by the repulsive force from the obstacles. To get efficient control, we introduce two kinds of safety-values to quantify the safety status of an agent as below.

Definition 5 (obstacle safety). *The Obstacle safety $S_k^o = [s_{k1}^o, s_{k2}^o, \ldots, s_{kN}^o] \in R^N$ is a vector with each element varying between 0 and 1, where N is the number of the agents and $s_{ki}^o < 1$ denote the Obstacle safety between agent i and obstacle k, where i, k are the indices of the agent and the obstacle, respectively. The element s_{ki}^o can be calculated as follows:*

$$s_{ki}^o = f_{thr}\left(\frac{\|\hat{q}_{i,k} - q_i\|}{D_\beta}\right), \tag{7}$$

where the $\hat{q}_{i,k}$ is the location of the β-agent, $\|\hat{q}_{i,k} - q_i\|$ is the real distance between agent i and its β-agent, D_β is the set distance between agent i and its β-agent.

$$f_{thr}(x) = \begin{cases} x & x < 1 \\ 1 & x \geq 1. \end{cases} \tag{8}$$

$s_{ki}^o = 1$ means that agent i is safe with obstacle k. $s_{ki}^o = 0$ means that agent i crashes the obstacle k. Analogously, we can also define the Agent safety to quantify the safety situation among agents:

Definition 6 (agent safety). *The agent safety $S^\alpha = \{s_{ij}^\alpha\} \in R^{N\times N}$ is an array with each element varying between 0 and 1, s_{ij}^α denote the agent safety between agent i and agent j and can be calculated as follows:*

$$S^\alpha = \{s_{ij}^\alpha\}, \quad s_{ij}^\alpha = f_{thr}\left(\frac{\|q_j - q_i\|}{D_\alpha}\right). \tag{9}$$

Here we have to mention that because of the cohesive force from the γ-agent (virtual leader), even in a free space, the system would not form a strict α-lattice, which means the distance between agents may shorter than D_α even when a stable flocking behavior of the system is formed. So there is an agent safety diminishment caused by the cohesive force from the virtual leader (γ-agent) and the S_k^o will stabilize in a matrix with elements slightly smaller than 1.

By using the defined and , we can design the LSF control protocol of system in a limited space.

3.2 LSF algorithm in a limited space

Based on the Olfati-Saber model, we propose an algorithm to improve safety-value and the average speed of a multi-agent system in limited space. Noticed that when a multi-agent system meets a limited space, its structure just can be slightly changed by Olfati-Saber algorithm, accordingly, the key point of the LSF algorithm is to add corresponding control in order to automatically change the formation to greatly enhance the safety-value and speed up the system at the same time. When an agent i get dangerous with obstacle k, namely $s_{ki}^o < 1$, we can introduce two kinds of special α-agents to get more precise control.

Definition 7 (ε_k-agent). *δ_k-agent is a virtual agent, and for obstacle k, agent i of the multi-agent system is a ε-agent when it satisfies:*

1. *$s_{ki}^o < 1$ and $(s_{ki}^o)' > 0$*
2. *There does not exist any j s.t. $s_{kj}^o \geq s_{ki}^o$, $a_{ij} > \lambda_{min}$ and $(p_j \cdot v_\gamma)/\|v_\gamma\| > (p_i \cdot v_\gamma)/\|v_\gamma\|$*

where $(s_{ki}^o)'$ is the derivative of s_{ki}^o with respect to time.

Definition 8 (δ_k-agent). *δ_k-agent is a virtual agent and for obstacle k, agent i of the multi-agent system is a δ_k-agent when it satisfies:*

1. *$s_{ki}^o < 1$*
2. *exists j s.t. $s_{kj}^o \geq s_{ki}^o$, $a_{ij} > \lambda_{min}$ and $p_j \cdot v_\gamma/\|v_\gamma\| > p_i \cdot v_\gamma/\|v_\gamma\|$*

© ISAROB 2013

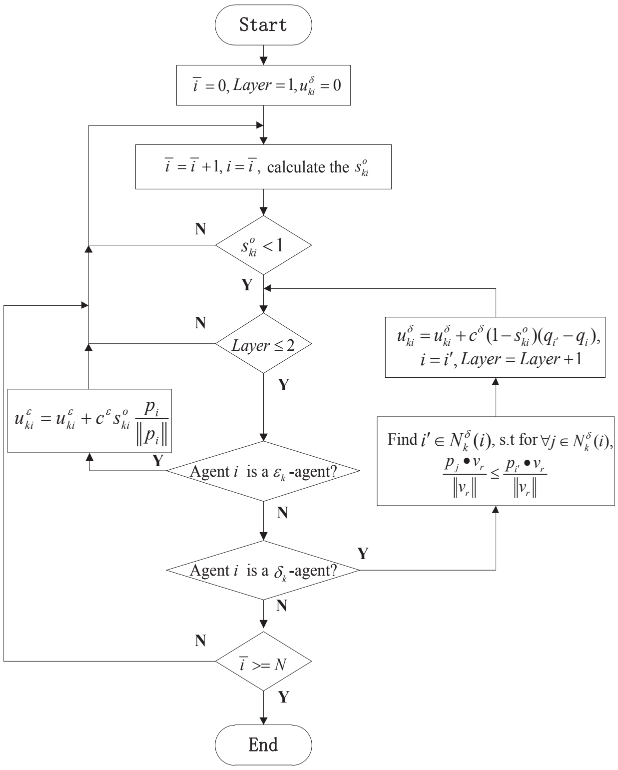

Fig. 1. Flow chart of LSF algorithm 1

where $\lambda_{min}\in\mathbb{R}, 0<\lambda_{min}\le 1$, v_γ *is the velocity of the* γ*-agent (virtual leader).*

The collection of j here is denoted by $N^{\delta}_{k}(i)$. Notice that an agent can't be ε_k-agent and δ_k-agent at the same time. We present an algorithm to give control to these two kinds of agents respectively. The flow chart of LSF algorithm 1 is displayed in Fig. 1.

Algorithm 1:

It can be inferred from the definition of the ε_k-agent that the ε_k-agents are on the forth front of the system and are heading toward a safer place with an increasing obstacle safety, which makes the effective control to simply be the propulsion along with the same direction. The control u^{ε}_{ki} is obtained in (10):

$$u^{\varepsilon}_{ki}=c^{\varepsilon}s^{o}_{ki}\frac{p_i}{\|p_i\|} \tag{10}$$

where the c^{ε} is a constant coefficient for u^{ε}_{ki}. As to the δ_k-agent, compared to the ε_k-agent, it functions like the follower to the pioneer in the system. δ_k-agent always has agents with higher obstacle safety before it. In order to maximum the effect of changing the structure, enhance the obstacle safety, and securely speed up the velocity, we propel the δ_k-agent to a forefront safer neighboring agent. The specific process is to find $i'\in N^{\delta}_{k}(i)$, s.t for $\forall j\in N^{\delta}_{k}(i)$, $p_j\cdot v_\gamma/\|v_\gamma\|\le p_{i'}\cdot v_\gamma\|v_\gamma\|$, then the corresponding control u^{δ}_{ki} is given in (11)

$$u^{\delta}_{ki}=c^{\delta}(1-s^{o}_{ki})(q_{i'}-q_i) \tag{11}$$

where the c^{δ} is a constant coefficient for u^{δ}_{ki}.

After adding up the control with respect to the obstacles, the result can be expressed in the formulation below:

$$u^{\delta}_{i}=\sum_{k}u^{\delta}_{ki}, \quad u^{\varepsilon}_{i}=\sum_{k}u^{\varepsilon}_{ki}. \tag{12}$$

The algorithm 1 works well in obstacle avoidance, however, the improvement of the obstacle safety is gained on a little sacrifices of the agent safety, namely the agent safety may drop a little. In order to reduce such sacrifices and make a complement to algorithm 1, we use a complementary algorithm to make some amendment.

Algorithm 2:

When the system is passing through a limited place, the u^{θ}_{i} can be added to the Algorithm and be gained as follows:

$$u^{\theta}_{i}=c^{\theta}\sum_{j}\frac{a(i,j)\Delta s^{\alpha}_{i,j}}{s^{\alpha}_{i,j}}\frac{(q_j-q_i)}{\|q_j-q_i\|}, \tag{13}$$

where $\Delta s^{\alpha}_{i,j}=s^{\alpha}_{i,j}(t)-s^{\alpha}_{i,j}(t-\Delta t)$ is the change of $s^{\alpha}_{i,j}$ from $t-\Delta t$ to t, t is the temporary moment.

The function of u^{θ}_{i} is like a buffer in the system, when reduction of the ,namely the $\Delta s_{i,j}$ is huge and the $s^{\alpha}_{i,j}$ is low, the u^{θ}_{i} is fairly considerable. So by adding u^{θ}_{i}, the reduction of the agent safety can be greatly decreased and then the Algorithm 2 can make complement to the Algorithm 1. In conclusion, the LSF distributed control protocol can be given by the following formula (14):

$$u^{LSF}_{i}=u^{OS}_{i}+u^{\varepsilon}_{i}+u^{\theta}_{i} \tag{14}$$

where the u^{OS}_{i} is the control of Olfati-Saber algorithm.

4 SIMULATION RESULT

In this section, we use some simulations to demonstrate the effectiveness of our design method - LSF algorithm. Without loss of generality, we can specify the limited space into a channel, the number of the system N is 25, the width of the channel is 8, and length is 20. The parameters for each kind of agent are set as follows:

$$\begin{array}{lcl} c^{\alpha}_{1}=1 & , & c^{\alpha}_{2}=1, \\ c^{\gamma}_{1}=1 & , & c^{\gamma}_{2}=1, \\ c^{\beta}_{1}=4.5 & , & c^{\beta}_{2}=5.8, \\ c^{\delta}=6.55 & , & c^{\theta}=7.7. \end{array}$$

and $c^{\varepsilon}=100.215$. To contrast the control performances between the LSF algorithm and the Olfati-Saver algorithm of multi-agent systems in a limited space, three main qualities of the process of proceeding though the channel are calculated to get further analysis of the two algorithms. $s^{o}=\min_{k,i}(s^{o}_{ki})$ and $s^{\alpha}=\min_{i,j}(s^{\alpha}_{ij})$ can show the safety-value of the system in a limited space, $V_{average}$ is the average velocity of all the agents of the system.

In figure 2 , the blue solid line represents the results of Olfati-Saber algorithm, and the red dotted line represent the results of LSF algorithm. Looking at the Fig. 2, we can see, the LSF algorithm can accelerate the average speed, greatly enhance the s^{o} and remain the s^{α}.

© ISAROB 2013

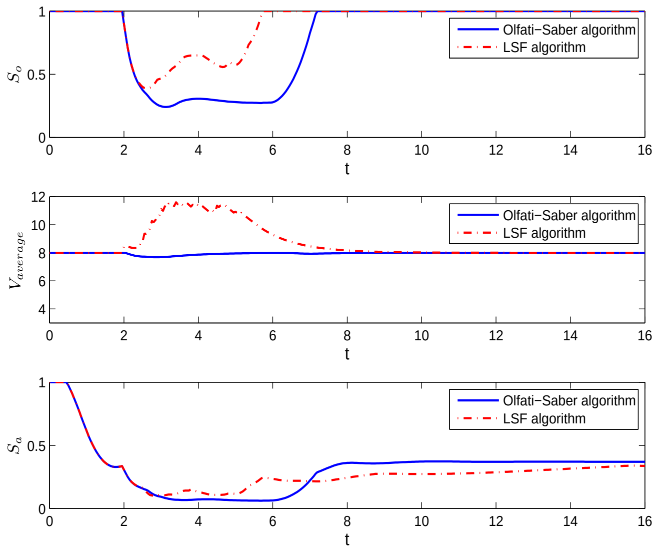

Fig. 2. Contrast figure between two algorithms about $s^o, s^\alpha, V_{average}$

It has been shown that, under the LSF flocking control protocols, the stable flocking behaviors can be achieved and the average speed of the system in a limited space is improved. What's more, the obstacle safety is greatly enhanced and the agent safety basically stays the same. Study of simulation reveals that the LSF control algorithm can speed up the system and improve its safety as the same time.

5 CONCLUSION

In this paper, we study the flocking control for multi-agent systems in a limited space. A new coefficient called the safety-value is introduced to describe the safety between agents and obstacles, On the basis of the safety-value and Olfati-Saber algorithm, a distributed flocking control protocol - LSF algorithm is designed to control the multi-agent system in a limited space. By using local information of each agent, the system can automatically and flexibly change its velocity and structure in order to pass the limited space both quickly and safely. Simulation reveals that the proposed LSF control protocol can greatly improve the velocity of the multi-agent system with even enhanced safety-value.

6 ACKNOWLEDGMENTS

This work was supported by the National 973 Program (2012CB821200, 2012CB821201) and the NSFC (61134005, 60921001, 90916024, 91116016).

REFERENCES

[1] Li Q, Jiang ZP (2009), Global Analysis of Multi-Agent Systems Based on Vicsek's Model. IEEE Trans. Automa. Control, 54(12): 2876-2881.

[2] Lin ZY, Broucke M, Francis B (2004), Local Control Strategies for Groups of Mobile Autonomous Agents. IEEE Trans. Automa. Control 49(4):622-629

[3] Ren W, Beard RW (2005), Consensus Seeking in Multi-agent Systems Under Dynamically Changing Interaction Topologies. IEEE Trans. Automa. Control 50(5):655-661

[4] Reynolds CW (1987), Flocks,Herds,and Schools: A Distributed Behavioral Model. Computer Graphics 21(4): 25-34

[5] Shi H, Wang L, Chu TG, et al (2007), Flocking of Multi-Agent Systems with a Virtual Leader. In: Proceedings of the 2007 IEEE Symposium on Artificial Life, pp. 287-294

[6] Carpin S, Parker LE (2002), Cooperative Leader Following in a Distributed Multi- Robot System. In: Proceedings of IEEE International Conference on Robotics Automation, pp. 2994-3001

[7] Tanner HG (2004), Flocking with Obstacle Avoidance in Switching Networks of Interconnected Vehicles. In: Conference of Robotics and Automation, pp. 3006-3011

[8] Olfati-Saber R (2006), Flocking for multi-agent dynamic systems: Algorithms and theory. IEEE Trans. Automat. Control 51(3): 401-420

[9] Hong Y, Gao L, Cheng D, et al (2007), Lyapunov-based approach to multi-agent systems with switching jointly-connected interconnection. IEEE Trans. Autom. Control 52(5):943-948

[10] Gazi V, Passino KM (2004), A class of attraction/repulsion functions for stable swarm aggregations. Int. J. Control 77(18): 1567-1579

[11] Su H, Wang X, Lin Z (2009), Flocking of multi-agents with a virtual leader. IEEE Trans. Autom. Control 54(2): 293-306

[12] Rimon E, Koditschek DE (1990), Robot navigation functions on manifolds with boundary. Adv. Appl. Math 11(4): 412-442

[13] Gazi V, Passino K (2004), Stability analysis of social foraging swarms. IEEE Trans. Syst., Man, Cybern. B: Cybern. 34(1): 539-557

[14] Bollob s B (1998), Modern Graph Theory. Vol. 184 of Graduate Texts in Mathematics. New York: Springer-Verlag

[15] Diestel R (2000), Graph Theory. Vol. 173 of Graduate Texts in Mathematics. New York: Springer-Verlag

[16] Horn RA, Johnson CR (1987), Matrix Analysis. Cambridge Univ. Press, Cambridge

[17] Godsil C, Royle G (2001), Algebraic Graph Theory, Vol. 207 of Graduate Texts in Mathematics. New York: Springer-Verlag

[18] Olfati-Saber R (2003), Flocking with obstacle avoidance. California Inst. Technol., Control Dyna. Syst., Pasadena, CA, Tech. Rep. 2003-006, Feb

[19] Tanner HG (2004), Flocking with obstacle avoidance in switching networks of interconnected vehicles. In: Conference Robotics and Automation, pp. 3006-3011.

[20] Saber RO, Murray RM (2003), Flocking with obstacle avoidance: cooperation with limited communication in mobile networks. In: Proceedings 42nd IEEE Conf. Decision and Control. Dec, pp. 2022-2028.

© ISAROB 2013