

具有复杂障碍物的多智能体蜂拥控制算法

高晶英¹, 徐 旭², 何斯日古楞¹

¹呼和浩特民族学院数学与大数据学院, 内蒙古 呼和浩特

²吉林大学数学学院, 吉林 长春

Email: minzugjy@sina.com

收稿日期: 2021年8月22日; 录用日期: 2021年9月12日; 发布日期: 2021年9月24日

摘 要

本文研究具有切换拓扑网络的一般二阶多智能体系统蜂拥控制算法, 当多智能体系统在遇到复杂障碍物时, 引入了一种新的虚拟智能体产生一个切向排斥力, 使得多智能体系统能够更光滑的通过障碍物。同时, 采用动态牵制策略控制整个网络的连通性, 确保多智能体系统能够达到蜂拥状态且智能体之间不发生碰撞。最后给出了一些数值模拟结果, 证实了算法的有效性。

关键词

多智能体, 蜂拥控制, 复杂障碍物

Flocking Control Algorithm of Multi-Agent System with Complex Obstacles

Jingying Gao¹, Xu Xu², Siriguleng He¹

¹School of Mathematics and Big Data, Hohhot Minzu College, Hohhot Inner Mongolia

²College of Mathematics, Jilin University, Changchun Jilin

Email: minzugjy@sina.com

Received: Aug. 22nd, 2021; accepted: Sep. 12th, 2021; published: Sep. 24th, 2021

Abstract

The flocking control algorithm for general second-order multi-agent systems with switching topology networks is investigated in this paper. When the multi-agent system encounters complex obstacles, a new virtual agent is introduced to produce a tangential repulsion force, so that the multi-agent system can pass through obstacles more smoothly. At the same time, the dynamic pinning control strategy is adopted to control the connectivity of the whole network, so as to en-

sure that the multi-agent system can reach the flocking state and there is no collision between agents. Finally, some numerical simulation results are given to verify the effectiveness of the algorithm.

Keywords

Multi-Agent System, Flocking Control, Complex Obstacles

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

多智能体系统(Multi-agent System, MAS)是由大量的智能体构成,它是将大而复杂的系统转化成多个、简单的并且彼此相互通信的系统。受到人工智能理论的启发,多智能体系统最初的目标是弥补单一复杂系统在解决大型复杂实际应用中的不足之处。随着计算机技术的突飞猛进,同时受到大自然生物群体协调运动的启发,多智能体协调控制问题的研究取得了非常丰富的成果。

目前大部分多智能体蜂拥控制算法考虑的是双重积分模型,由于该系统的简单性,针对该系统的蜂拥控制算法研究已经有很多成果[1] [2] [3] [4]。但是在实际应用当中不可能把所有系统都用二阶模型表示,因此有必要把二阶模型推广到更一般的模型[5] [6]。此外,多智能体系统在运动过程中难免遇到障碍物情况,如何有效的绕过障碍物成为了多智能体协调控制问题的另一个热点研究方向。Olfati-Saber [7]提供了一种多智能体能够有效绕过障碍物的方法。文献[8]在 Olfati-Saber 算法的基础上提出了一种更有效绕过障碍物的方法,增加了转向机制,当智能体遇到障碍物时根据其前进方向与障碍物之间夹角决定智能体应从障碍物的哪一侧绕过,大大提高了避障效率。文献[9]中给出了一种人工势函数结合流函数的方法,智能体根据流函数决定如何绕过障碍物,而势函数能保证智能体与障碍物之间不会发生碰撞。

本文研究一般二阶系统多智能体蜂拥控制算法,使其能够解决每个智能体具有更一般的二阶系统多智能体通过复杂障碍物时的蜂拥控制问题。考虑具有切换拓扑结构的多智能体网络,假设多智能体网络有一个虚拟领导者且有一部分智能体能够收到虚拟领导者的反馈信息。为了解决网络不连通的问题,采用动态牵制策略实现整个智能体网络蜂拥控制。为了能更光滑的绕过障碍物,引入了一种新的虚拟智能体叫做 β_2 -智能体,使其对 α -智能体产生一个切向排斥力。因此本文具有障碍物的蜂拥控制算法当中有两种 β -智能体,它们产生的排斥力大小相同、方向垂直。最后给出了一些模拟结果,证实了算法的有效性。

2. 问题阐述

考虑 n 个智能体在 m 维欧式空间上移动,每个智能体的运动方程表示为

$$\begin{bmatrix} \dot{q}_i \\ \dot{p}_i \end{bmatrix} = (X \otimes I_m) \begin{bmatrix} q_i \\ p_i \end{bmatrix} + (Y \otimes I_m) u_i, \quad i = 1, 2, \dots, n \quad (1)$$

其中 $q_i \in \mathbf{R}^m$ 是智能体 i 的位置向量, $p_i \in \mathbf{R}^m$ 是智能体 i 的速度向量, $u_i \in \mathbf{R}^m$ 智能体 i 的控制输入向量。 X 是 Y 是二阶常量矩阵, I_m 是 m 阶单位矩阵, \otimes 表示 Kronecker 积, 标记虚拟领导者的下标为 0。

为了使虚拟领导者以常速运动,它的控制输入 u_i 定义为

$$X = \begin{bmatrix} \xi_{11} & \xi_{12} \\ \xi_{21} & \xi_{22} \end{bmatrix}, Y = \begin{bmatrix} \zeta_1 \\ \zeta_2 \end{bmatrix}. \quad (2)$$

虚拟领导者可以看作是提前给定的预定路线，蜂拥控制的目的是多智能体不仅要达到蜂拥状态而且还要按照预定路线移动，如来自地面指挥员的命令。本文考虑了切换拓扑结构，因此把每个智能体的邻域可以看作是时间 t 的函数。两个智能体之间的距离超过感应半径时，它们之间的连边会断开，有可能整个多智能体网络会不连通。本文采用动态牵制策略达到多智能体网络蜂拥控制目标。动态牵制策略的具体步骤是每次拓扑结构发生变化的时候，把整个智能体分为若干个连通子网络，然后从每个子网络当中选择度最大的智能体作为牵制节点(能收到虚拟领导者的反馈信息)。此时，每个智能体在每一个时刻都能直接或者间接的接收到虚拟领导者的位置和速度信息，从而既不用每个节点都进行牵制控制，也不需要整个智能体网络时刻连通的情况下，达到对多智能体网络蜂拥控制目的，见图 1。

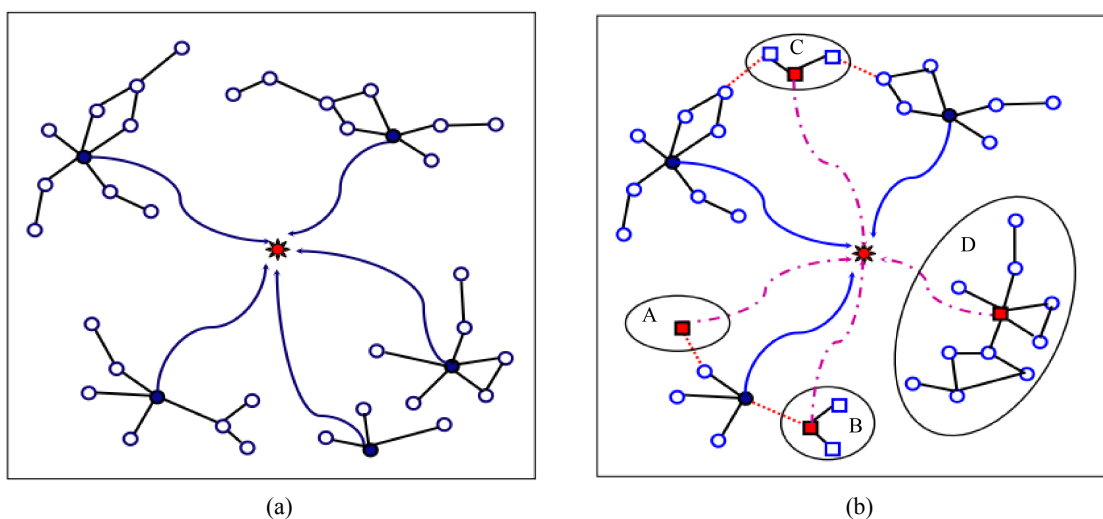


Figure 1. Dynamic pinning strategy
图 1. 动态牵制策略

3. 具有复杂障碍物的多智能体蜂拥控制算法

本文考虑了具有复杂障碍物的动态牵制蜂拥控制算法，即多智能体系统在蜂拥过程中遇到复杂障碍物的情况，见图 2，其中白色区域是智能体可以通过的空间，其它区域是障碍物，如楼房，墙体等等。由于本文算法是基于文献[7]中第三个算法得到的，所以首先介绍一下在文中出现的相关概念。多智能体网络中实际存在的智能体称为 α -智能体，而障碍物的影响用另一种虚拟智能体表示，该智能体称为 β -智能体。 β -智能体可以通过在障碍物表面上映射 α -智能体的位置而得到，目的是使 α -智能体不会与障碍物发生碰撞。 β -智能体会随着 α -智能体的移动而在障碍物边缘上移动。当 α -智能体感知障碍物的时候，根据自己的位置可以在障碍物表面上得到一个与自己对应的 β -智能体，之后 α -智能体和 β -智能体之间会产生排斥力，从而 α -智能体能够有效的避免与障碍物碰撞。为了使 α -智能体在遇到障碍物的时候能更光滑的绕过障碍物，本文引入了另一种虚拟智能体，但是该虚拟智能体不在障碍物表面上并且能给 α -智能体一个切向的排斥力。因此在本文算法当中有两种不同的 β -智能体，为了避免发生混淆，在障碍物表面上的虚拟智能体称为 β_1 -智能体，而不在障碍物表面上的虚拟智能体称为 β_2 -智能体，见图 3(a)。

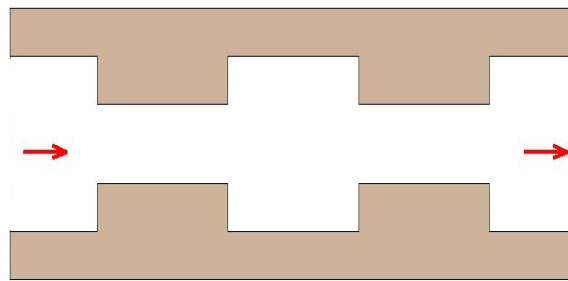


Figure 2. Geometric characteristics of complex obstacle

图 2. 复杂障碍物几何特征

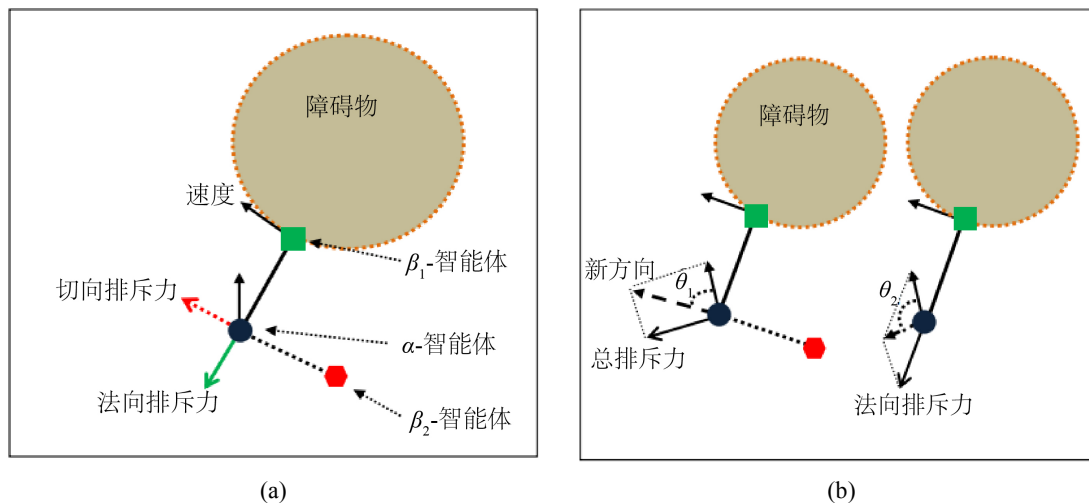


Figure 3. Schematic diagram of α -agent, β_1 -agent, β_2 -agent, normal repulsion force and tangential repulsion force

图 3. α -智能体, β_1 -智能体, β_2 -智能体, 法向排斥力, 切向排斥力示意图

β_2 -智能体依然是虚拟智能体, 它用于产生对 α -智能体一个切向排斥力, 排斥力的方向与 β_1 -智能体的速度方向一致, 且 α -智能体会向 β_1 -智能体的速度方向从障碍物的一侧绕过。由 β_1 -智能体产生的排斥称为法向排斥力, 而由 β_2 -智能体产生的排斥力称为切向排斥力, 两种排斥力是垂直的。因此, α -智能体与障碍物之间的排斥力是这两种排斥力的总和, 见图 3(a)。从图 3(b)可以看出, 由于增加了切向排斥力, α -智能体新的速度方向与原速度方向之间的夹角 θ_1 总是小于在没有切向力时的 α -智能体新的速度方向与原速度方向之间的夹角 θ_2 。这说明本文算法中的智能体更能光滑的绕过障碍物, 并且不会与障碍物发生碰撞。每个智能体的控制输入分为三个部分, 它们分别是每个 α -智能体与它周围智能体之间的作用力, β_1 -智能体与 β_2 -智能体之间的排斥力, β_2 -智能体与 α -智能体之间的排斥力。

本文提出的具有复杂障碍物蜂拥控制算法可以总结为如下几个步骤:

- (1) 首先得到 β_1 -智能体在障碍物 O_k 边缘上的位置 $q_{i,k}$ 和速度 $p_{i,k}$, O_k 表示第 k 个障碍物。
- (2) 根据 β_1 -智能体的位置和速度确定 β_2 -智能体的位置 $q_{i,l}$ 。 β_2 -智能体位置满足: $(q_i - q_{i,l}) \cdot p_i \geq 0$, $\langle (q_{i,l} - q_i) \cdot (q_{i,k} - q_i) \rangle = \pi/2$ 和 $\|q_{i,l} - q_i\|_\sigma = \|q_{i,k} - q_i\|_\sigma$, 其中 $\langle \cdot \rangle$ 表示两个向量的夹角。
- (3) 增加对 α -智能体的两个排斥力 $\psi_\beta(\|q_{i,k} - q_i\|_\sigma)$ 和 $\psi_\beta(\|q_{i,l} - q_i\|_\sigma)$, 其中 $\psi_\beta(\cdot)$ 是 α -智能体与 β -智能体之间的势函数。

在步骤 2 中, $(q_i - q_{i,l}) \cdot p_i \geq 0$ 表示向量 $q_i - q_{i,l}$ 和速度向量 p_i 的夹角总是锐角, $\langle (q_{i,l} - q_i) \cdot (q_{i,k} - q_i) \rangle = \pi/2$ 表示向量 $q_{i,l} - q_i$ 垂直于向量 $q_{i,k} - q_i$, $\|q_{i,l} - q_i\|_\sigma = \|q_{i,k} - q_i\|_\sigma$ 表示从 β_1 -智能体和 β_2 -智能体到 α -智能体的距离相等, 这意味着 β_1 -智能体和 β_2 -智能体产生的排斥力相等。

文献[7]中当 α -智能体感应到障碍物的时候不是立刻对障碍物产生排斥作用的, 而是障碍物和 α -智能体的距离小于某个临界值时才产生作用。本文提出的有障碍物蜂拥控制算法中对人工势函数做了一点改动, 当 α -智能体感应到障碍物时马上在 α -智能体和 β_1 -智能体之间产生排斥力。

α -智能体和 β -智能体的邻域分别定义如下

$$N_i^\alpha = \{j \in V \mid \|q_j - q_i\|_\sigma < r\}, \tag{3}$$

和

$$N_i^\beta = \left\{ k \mid \|q_{i,k} - q_i\|_\sigma < r' \right\} \cup \left\{ l \mid (q_i - q_{i,l}) \cdot p_i \geq 0, \langle (q_{i,l} - q_i) \cdot (q_{i,k} - q_i) \rangle = \pi/2 \text{ and } \|q_{i,l} - q_i\|_\sigma = \|q_{i,k} - q_i\|_\sigma \right\}. \tag{4}$$

其中 $\|\cdot\|_\sigma$ 表示 sigma 范数, r, r' 分别是 α -智能体和 β -智能体的感应半径且可以选择 $r' \leq r$ 。

下面给出具有障碍物的动态牵制蜂拥控制算法。对于每个智能体 i 的控制输入定义如下

$$u_i = (w_i + v_i) / \zeta_2, \quad i = 1, 2, \dots, n. \tag{5}$$

其中第一项 w_i 为智能体的状态反馈项, 第二项 v_i 为智能体 i 的协作控制项。设计控制输入的目的是使所有智能体的速度达到一致且智能体之间不会发生碰撞。为了达到该目的, 第一项 w_i 可以考虑如下形式

$$w_i = -\xi_{21}q_i - \xi_{22}p_i, \tag{6}$$

第二项 v_i 可以考虑如下形式

$$v_i = - \sum_{j \in N_i^\alpha(t)} \nabla_{q_i} \psi_\alpha(\|q_i - q_j\|_\sigma) - \sum_{j \in N_i^\alpha(t)} a_{ij}(t)(p_i - p_j) - h_i(t) [c_1(q_i - q_0) + c_2(p_i - p_0)] - \sum_{l \in N_i^\beta(t)} \nabla_{q_i} \psi_\beta(\|q_i - q_{i,l}\|_\sigma) - \sum_{k \in N_i^\beta(t)} \nabla_{q_i} \psi_\beta(\|q_i - q_{i,k}\|_\sigma). \tag{7}$$

其中, $c_1, c_2 > 0$, q_0 和 p_0 分别是虚拟领导者的位置和速度向量。如果智能体 i 是牵制节点, 则 $h_i(t) = 1$, 否则 $h_i(t) = 0$ 。 $\psi_\beta(z)$ 如文献[7]中定义的势函数。公式(7)中最后两项分别是 β_1 -智能体和 β_2 -智能体与 α -智能体之间的排斥力。

4. 模拟结果与讨论

本文模拟了 $n = 30$ 个智能体在控制输入(7)的影响下在 2 维平面上运动, 验证当多智能体网络穿过复杂障碍物时的动态牵制蜂拥控制算法的有效性。30 个智能体的初始位置和初始速度分别由区间 $[0, 20] \times [0, 20]$ 和 $[-1, 1] \times [-1, 1]$ 随机生成。每个智能体的感应半径为 $r = 4$, 感应障碍物的半径 $r' = 2, c_1 = 0.3, c_2 = 6$, 虚拟领导者的初始位置和初始速度分别设为 $q_0(0) = [160, 12]^T$ 和 $p_0 = [0.3, 0]^T$, 并且虚拟领导者的速度是不变的。动力学方程(1)中矩阵 X 和 Y 设置为

$$X = \begin{bmatrix} 2 & 2 \\ 2 & -1 \end{bmatrix}, Y = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

在这个部分考虑了两种不同几何特征的复杂障碍物，如图 4 和图 5，障碍物外部的虚线表示危险区域，智能体一旦进入危险区域，立刻得到与之相对应的虚拟智能体，进而能顺利的通过复杂障碍物。首先，30 个多智能体需要通过两个不同宽度的过道，第一个过道比较宽，第二个过道相比于第一个比较窄。在遇到第二种障碍物的时候多智能体需要通过一条蛇形通道，这种障碍物的几何特征相比第一个障碍物要复杂一点，两个狭窄过道宽度相同，选择这种复杂障碍物的可以更好的验证本文动态牵制蜂拥控制算法通过复杂障碍物的能力。图 4 和图 5 显示了 30 个智能体在通过两种复杂障碍物时的运动轨迹，从图可以清楚的看出所有智能体顺利通过障碍物且没有与障碍物发生碰撞。

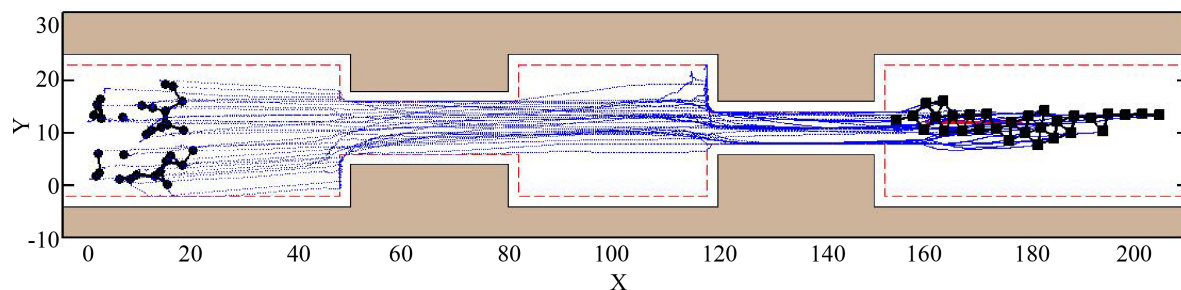


Figure 4. Trajectory of 30 agents passing through the first complex obstacle

图 4. 30 个智能体通过第一种复杂障碍物时的运动轨迹

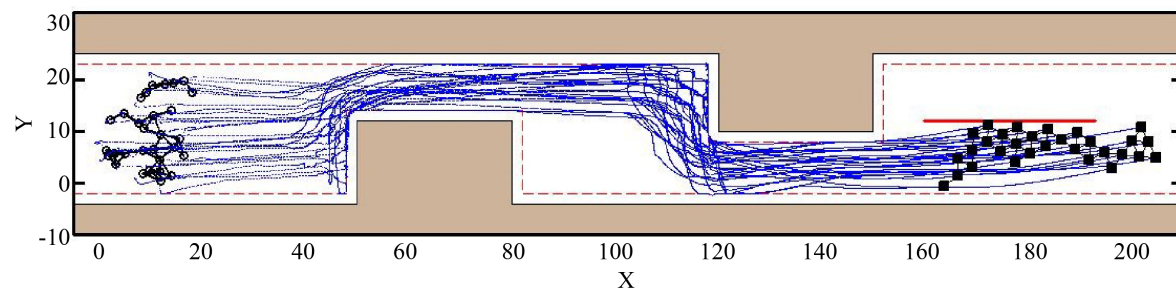


Figure 5. Trajectory of 30 agents passing through the second complex obstacle

图 5. 30 个智能体通过第二种复杂障碍物时的运动轨迹

5. 小结

本文提出了一种具有一般二阶线性系统的多智能体动态牵制蜂拥控制算法。基于动态牵制策略，提出了一种新的控制协议，在多智能体网络不需要假设多智能体网络连通或者使用无穷大势函数保持网络连通的情况下，有效的解决了多智能体系统蜂拥控制问题。为了能更光滑的绕过障碍物，引入了一种新的虚拟智能体叫作 β_2 -智能体对 α -智能体产生一个切向排斥力。最后，给出了一些模拟结果，进一步证实了算法的有效性。

基金项目

内蒙古自治区高等学校科学研究项目(NJZY19174); 呼和浩特民族学院博士基金(HMBS1903); 国家自然科学基金(11961022); 呼和浩特民族学院科研创新团队建设计划项目(HMTD202005); 呼和浩特民族学院校级项目。

参考文献

- [1] Yu, W.W., Chen, G.R. and Cao, M. (2010) Distributed Leader-Follower Flocking Control for Multi-Agent Dynamical

Systems with Time-Varying Velocities. *Systems and Control Letters*, **59**, 543-552.

<https://doi.org/10.1016/j.sysconle.2010.06.014>

- [2] Gao, J., Xu, X., Ding, N., *et al.* (2017) Flocking Motion of Multi-Agent System by Dynamic Pinning Control. *IET Control Theory & Applications*, **11**, 714-722. <https://doi.org/10.1049/iet-cta.2016.1150>
- [3] 刘佳, 陈增强, 刘忠信. 多智能体系统及其协同控制研究进展[J]. 智能系统学报, 2010, 5(1): 1-9.
- [4] 谢光强, 章云. 多智能体系统协调控制一致性问题研究综述[J]. 计算机应用研究, 2011, 28(6): 2035-2039.
- [5] Yazdani, S. and Haeri, M. (2015) Position Convergence of Informed Agents in Flocking Problem with General Linear Dynamic Agents. *IET Control Theory & Applications*, **9**, 392-398. <https://doi.org/10.1049/iet-cta.2014.0478>
- [6] Yazdani, S. and Haeri, M. (2016) Flocking of Multi-Agent Systems with Multiple Second-Order Uncoupled Linear Dynamics and Virtual Leader. *IET Control Theory & Applications*, **10**, 853-860. <https://doi.org/10.1049/iet-cta.2015.1194>
- [7] Olfati-Saber, R. (2006) Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory. *IEEE Transactions on Automatic Control*, **51**, 401-420. <https://doi.org/10.1109/TAC.2005.864190>
- [8] Wang, J., Zhao, H., Bi, Y., *et al.* (2014) An Improved Fast Flocking Algorithm with Obstacle Avoidance for Multi-agent Dynamic Systems. *Journal of Applied Mathematics*, **2014**, Article ID: 659805. <https://doi.org/10.1155/2014/659805>
- [9] Wang, Q., Chen, J., Fang, H., *et al.* (2014) Flocking Control for Multi-Agent Systems with Stream-Based Obstacle Avoidance. *Transactions of the Institute of Measurement and Control*, **36**, 391-398. <https://doi.org/10.1177/0142331213503864>