

基于网络特征重数的多智能体牵制蜂拥控制

陈世明¹ 梅艳红¹ 赖强¹ 方华京²

(1 华东交通大学电气与电子工程学院, 江西 南昌 330013;

2 华中科技大学自动化学院, 湖北 武汉 430074)

摘要 针对多智能体系统蜂拥控制过程中存在效率不高的现象,融合图论和控制理论的分析方法,提出一种基于网络特征重数的牵制蜂拥控制算法.该算法通过网络拓扑结构的邻接矩阵,根据其特征值的最大重数,得出网络可控时所需牵制节点的数目,随后进一步利用控制理论中的PBH判据确定具体的牵制节点,并为之设计牵制蜂拥控制律.与随机牵制策略和度最大牵制策略进行的仿真实验统计结果对比证实了所提算法在提升整个系统控制效率方面的有效性.

关键词 多智能体;蜂拥控制;特征重数;PBH判据;牵制策略

中图分类号 TP273 **文献标志码** A **文章编号** 1671-4512(2016)05-0019-06

Flocking control for multi-agent systems via characteristics multiplicity of network and pinning

Chen Shiming¹ Mei Yanhong¹ Lai Qiang¹ Fang Huajing²

(1 School of Electrical and Electronic Engineering, East China Jiaotong University, Nanchang

330013, China; 2 School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract An effective pinning strategy was proposed to improve the control efficiency of multi-agent systems based on characteristics multiplicity of network of flocking control, and combining the theory of graph and analytical of control theory method. The algorithm was based on adjacency matrix of the multi-agent network topology structure and according to maximum multiplicity of its eigenvalue, the number of pinned nodes was obtained when the network controllable, and then by using the control theory of PBH (Popov-Belevitch-Hautus) criterion to identify the specific pinned nodes, the pinning flocking control law was designed. With randomly pinning strategy and the maximum degree pinning strategy as a comparative study, the simulation results demonstrate that the proposed algorithm is effective in improving the control efficiency of the entire system.

Key words multi-agent; flocking control; characteristics multiplicity; PBH criterion; pinning strategy

Reynolds 提出了模拟鸟群的 Boid 模型^[1],要求其满足避碰、向中心靠近和速度一致的规则,并给出了经典的蜂拥控制算法.在此基础上,随着智能技术的蓬勃发展,多智能体系统协调控制已成为复杂性科学领域的研究热点并取得了大量成果^[2-15].文献[2]提出了同时具有蜂拥和反蜂拥特性的半蜂拥算法,文献[3]提出了一种保持连通性

策略和自适应蜂拥控制算法.考虑经典蜂拥算法可能会产生分裂现象,文献[4]根据智能个体视野的变化提出一种局部蜂拥控制策略,所有智能体在蜂拥追踪和反蜂拥搜索模式下实现了多目标的追踪.文献[5]提出了带虚拟领导者的人工势函数的蜂拥控制算法,并假设虚拟领导者的状态信息是全局的.随着复杂系统的牵制控制^[6-7]概念的引

收稿日期 2015-09-17.

作者简介 陈世明(1977-),男,教授,E-mail: c1977318@hotmail.com.

基金项目 国家自然科学基金资助项目(61364017).

入,文献[8]在Saber的基础上进一步研究网络中只有少部分智能体可以获得虚拟领导者的状态信息,并验证了改进算法的有效性,但未对牵制节点的选取策略进行研究.文献[9]通过牵制的思想研究时变拓扑结构的多智能体系统,但牵制节点是依据随机选取的方式确定.文献[10]利用图的拉普拉斯矩阵特征值和所需稳定增益之间的相互作用设计同步控制器,研究离散时间有向多智能体系统的同步控制问题.文献[11]研究多智能体网络中少部分智能体可获得领导者信息的脉冲一致性算法,并获得了一致性的充分必要条件.

牵制的思想在很大程度上降低了控制成本,然而随机牵制具有不确定性,系统的控制效率不高.随着研究的深入,特定牵制控制的研究也在不断发展,文献[12]通过对各子网络间有联系的节点施加牵制控制,实现整体系统的一致行为.文献[13-15]根据有向网络节点的入度和出度确定具体的牵制节点,但只适用于有向网络.

鉴于当前多智能体牵制控制研究成果大多没有考虑牵制节点选取数目多少且控制效率不高的问题,本研究结合图论和控制理论的分析方法,对多智能体网络的蜂拥问题进行研究,提出一种基于网络特征重数的牵制蜂拥控制算法.首先针对任意给定拓扑结构的网络,求出其邻接矩阵和所有特征值的最大重数,利用矩阵的线性相关/无关特性确定网络中具体的牵制节点集,然后设计牵制蜂拥控制律,与随机牵制策略和度最大牵制策略相比,将控制的相关理论引入到蜂拥控制问题中,研究任意给定拓扑结构蜂拥控制如何有效选取牵制智能体,最后实现蜂拥行为,因此更加具有实际意义.

1 相关知识

定义1 图 $G=(V,E)$ 为任意一个图,其中 $V=\{1,2,\dots,N\}$ 是非空的节点(顶点)集合, N 为图 G 的节点数, $E=\{e_{ij}=(i,j):i,j\in V\}$ 是边的集合.若任意点对 (i,j) 与 (j,i) 对应同一条边,则该图称为无向图,否则称为有向图.

定义2 对于图 G ,其邻接矩阵为 $A=(a_{ij})_{N\times N}$,其中 $a_{ij}=l_{ij}$ 表示节点 i 与 j 间有权值为 l_{ij} 的连边,否则 $a_{ij}=0$.图 G 的拉普拉斯矩阵可表示为 $L=D-A$,其中 $D=\text{diag}(d_i)$ 为度矩阵, $d_i=\sum_{j=1}^N a_{ij}$.

定义3 L 的第二小特征值 λ_2 用来表征系统

的连通程度^[16-17]. λ_2 值越大,网络的连通性越好;反之, λ_2 越小,网络的连通性能越差.

定义4 假设 $r(r>0)$ 表示智能体间交互作用的感知半径,智能体 i 的邻域为

$$R_i(t)=\{j\in V:\|q_i-q_j\|<r\}.$$

假设系统中不存在孤立的智能体,智能体 i 在二维平面上运动的动力学方程为

$$\begin{cases} \dot{q}_i(t)=p_i(t); \\ \dot{p}_i(t)=u_i(t), \end{cases} \quad (1)$$

式中: q_i 和 p_i 分别为智能体 i 的位置和速度; u_i 为控制输入; $q_i,p_i,u_i\in\mathbb{R}^2$ ($i=1,2,\dots,N$).虚拟领导者在平面上运动,其运动方程为

$$\begin{cases} \dot{q}_\gamma(t)=p_\gamma(t); \\ \dot{p}_\gamma(t)=u_\gamma(q_\gamma(t),p_\gamma(t)), \end{cases} \quad (2)$$

式中: q_γ 和 p_γ 分别为虚拟领导者的位置和速度; u_γ 为虚拟领导者的控制输入.

2 多智能体牵制蜂拥控制算法

2.1 牵制策略——网络特征重数

本文的牵制策略只依据网络拓扑结构的邻接矩阵来确定,具体的牵制节点的选取方法如下.

步骤1 依据智能体的感知半径 r 和智能体 i 和 j 间的权值 l_{ij} ,求出图 G 的邻接矩阵 A ;

步骤2 求出矩阵 A 的所有特征值 $\lambda_1,\lambda_2,\dots,\lambda_N$,并找出所有特征值中重数最多的特征值 λ^M 和相应的重数 M ,且当网络为无向时, M 对应代数重数 M_1 ($M=M_1$),当网络为有向时, M 对应几何重数 M_2 ($M=M_2$);

步骤3 对矩阵行进行初等变换,找出 $A-\lambda^M I_N$ 最大线性无关的列;

步骤4 矩阵 $A-\lambda^M I_N$ 线性相关组所对应的列即为网络可控的牵制节点,其个数等于邻接矩阵 A 所有特征值中重数最多的特征值对应的重数 M .

算法的复杂度为 $O(N^2(\log N)^2)$,其中 N 为网络的节点数.当 $\text{rank}(A-\lambda_i I_N)=N$ 时,牵制节点数为1,即为网络中的任意一个节点.

图1以简单的无向网络和有向网络为例,描述了牵制节点的求解过程,其中实心黑色节点为牵制节点.图1(a)和(b)的邻接矩阵的特征值分别为:

$$\begin{aligned} \lambda_a &= [-4.146\ 2, -2.620\ 2, -1.000\ 0, \\ &\quad -0.257\ 5, 0.323\ 4, 2.600\ 2, 5.100\ 2]^T; \\ \lambda_b &= [0, 0, 0, 0, 0, 0, 0]^T, \end{aligned}$$

得出图1(a)所示图的邻接矩阵特征值均不同,代

数重数 $M_1 = 1$ 和图 1(b) 所示图的 $\lambda_b^{M_2} = 0$, 几何重数 $M_2 = 3$.

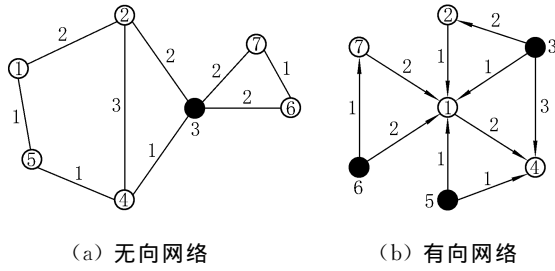


图 1 牵制节点的选取

由此根据步骤 2, 图 1(a) 的牵制节点数量为 1, 图 1(b) 的牵制节点数量为 3, 根据步骤 3 得出图 1(a) 的牵制节点为网络中的任意一个节点, 图 1(b) 的牵制节点为 3, 5, 6.

确定牵制节点个数后, 系统的状态方程可表示为

$$\dot{q}(t) = Aq(t) + Bu(t) \quad (q \in \mathbf{R}^N, u \in \mathbf{R}^M),$$

式中: $q(t)$ 为智能体在 t 时刻的状态; A 为邻接矩阵; $B = (b_{ij})_{N \times M}$ 为控制输入矩阵, 表示节点与外部输入信号间的联系; $u(t) = (u_1(t), u_2(t), \dots, u_M(t))^T$ 表示牵制节点在 t 时刻的控制输入.

2.2 多智能体牵制蜂拥控制算法

智能体 i 在二维空间中运动的控制输入为

$$u_i(t) = \alpha_i + \beta_i + \gamma_i, \quad (3)$$

式中: $\alpha_i = -\sum_{j \in N_i(t)} (p_i - p_j)$ 为速度一致项, 用以保证智能体之间的速度匹配; $\beta_i = -\sum_{j \in N_i(t)} \nabla_{q_i} V_{ij}$ 为人工势函数 V_{ij} 对位置 q_i 的梯度项, 用以实现智能体之间的分离和聚合; $\gamma_i = -k_i [c_1 (q_i - q_\gamma) + c_2 (p_i - p_\gamma)]$ 为引导反馈项, 其中 c_1, c_2 为反馈系数, 若智能体 i 能够获取虚拟领导者的信息, 则 $k_i = 1$, 否则 $k_i = 0$, 不能获取虚拟领导者信息的智能体可通过与邻居智能体的信息交互来更新自身状态, 最终实现蜂拥行为. 对于势函数 $\phi_a^{[5]}$, 当 $\|q_{ij}\| \rightarrow 0$ 时, ϕ_a 有最大值; 当 $\|q_{ij}\| \rightarrow d_a$ 时, ϕ_a 有最小值且为极小值, 当 $\|q_{ij}\| \geq r_a$ 时, ϕ_a 为常数 V_q . 综上知

$$V_{ij} = \sum_{j=1, j \neq i}^N \phi_a(\|q_{ij}\|_\sigma) = \sum_{j \notin N_i(t), j \neq i} \phi_a(r_\sigma) + \sum_{j \in N_i(t)} \phi_a(\|q_{ij}\|_\sigma). \quad (4)$$

2.3 稳定性分析

定理 1 考虑由动力学系统(1)和(2)组成的二阶系统, 初始分布是随机的, 且初始能量 H_0 为有限值. 在控制协议式(3)作用下, 所有智能体会渐近一致收敛到 p_γ , 最终实现稳定的蜂拥行为.

证明 令 $\tilde{q}_i = q_i - q_\gamma, \tilde{p}_i = p_i - p_\gamma$ 分别表示

智能体与虚拟领导者的位置和速度之间的差值, 则

$$\begin{cases} \dot{\tilde{q}}_i(t) = \tilde{p}_i(t); \\ \dot{\tilde{p}}_i(t) = \tilde{p}_i(t) = u_i(t). \end{cases} \quad (5)$$

将式(5)代入式(3)可得

$$u_i(t) = - \sum_{j \in N_i(t)} \nabla_{\tilde{q}_i} \phi_a(\|\tilde{q}_{ij}\|_\sigma) - \sum_{j \in N_i(t)} (\tilde{p}_i - \tilde{p}_j) - k_i (c_1 \tilde{q}_i + c_2 \tilde{p}_i). \quad (6)$$

李雅普诺夫函数为系统的总能量, 即系统智能体之间总的势能和智能体与虚拟领导者之间相对势能和动能之和, 即:

$$H = \frac{1}{2} \sum_{i=1}^N (V_i + k_i c_1 \tilde{q}_i^T \tilde{q}_i) + \frac{1}{2} \sum_{i=1}^N \tilde{p}_i^T \tilde{p}_i; \quad (7)$$

$$V_i = \sum_{j \notin N_i(t), j \neq i} \phi_a(r_\sigma) + \sum_{j \in N_i(t)} \phi_a(\|\tilde{q}_{ij}\|_\sigma). \quad (8)$$

对式(7)求导可得

$$\dot{H} = \frac{1}{2} \sum_{i=1}^N \dot{V}_i + k_i c_1 \sum_{i=1}^N \tilde{q}_i^T \dot{\tilde{q}}_i + \sum_{i=1}^N \tilde{p}_i^T \dot{\tilde{p}}_i. \quad (9)$$

由于人工势函数 ϕ_a 和邻接矩阵 A 的对称性, 有

$$\frac{1}{2} \sum_{i=1}^N \dot{V}_i = \sum_{i=1}^N \left[\sum_{j \in N_i(t)} \tilde{p}_i^T \nabla_{\tilde{q}_i} \phi_a(\|\tilde{q}_{ij}\|_\sigma) \right]. \quad (10)$$

将式(6)和式(10)代入式(9), 并进一步化简得到

$$\dot{H} = \sum_{j \in N_i(t)} \tilde{p}_i^T \left[- \sum_{j \in N_i(t)} (\tilde{p}_i - \tilde{p}_j) - k_i c_2 \tilde{p}_i \right]. \quad (11)$$

由于 $\sum_{j \in N_i(t)} (\tilde{p}_i - \tilde{p}_j) = \sum_{j=1}^N a_{ij} (\tilde{p}_i - \tilde{p}_j) = \sum_{j=1}^N l_{ij} \tilde{p}_i$, 因此

$$\dot{H} = - \sum_{i=1}^N \tilde{p}_i^T \left(\sum_{j=1}^N l_{ij} + k_i c_2 \right) \tilde{p}_i = - \tilde{p}^T [(L(t) + c_2 K(t)) \otimes I_{2 \times 2}] \tilde{p}, \quad (12)$$

式中 $\tilde{p} = \text{col}(\tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_N)$; $K(t) = \text{diag}(k_1, k_2, \dots, k_N)$. 由于 $L(t)$ 和 $K(t)$ 都是正半定矩阵, $L(t) + c_2 K(t)$ 也是正半定矩阵, 因此 $\dot{H} \leq 0$, 即 $H(t) \leq H_0$, 故多智能体系统在控制算法式(3)的作用下是渐近稳定的, 满足 $H(t) \leq H_0$ 集合 $\{\tilde{q}, \tilde{p}\}$, 即 $\Omega = \{(\tilde{q}^T, \tilde{p}^T)^T \in \mathbf{R}^{2 \times 2N} : H \leq H_0\}$ 是一个不变集. 根据 LaSalle 不变原理, 得到始于集合 Ω 内的每个解都会趋于最大的不变集 $\Omega_1 = \{(\tilde{q}^T, \tilde{p}^T)^T \in \mathbf{R}^{2 \times 2N} : \dot{H} = 0\}$. 由式(12)知: $\dot{H} = 0$ 等价于 $\tilde{p} = \text{col}(\tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_N) \equiv \text{col}(0, 0, \dots, 0)$, 因此, $\tilde{p}_1 = \tilde{p}_2 = \dots = \tilde{p}_N$ 或 $p_1 = p_2 = \dots = p_N = p_\gamma$, 这意味着所有智能体渐近收敛到虚拟领导者的速度 p_γ .

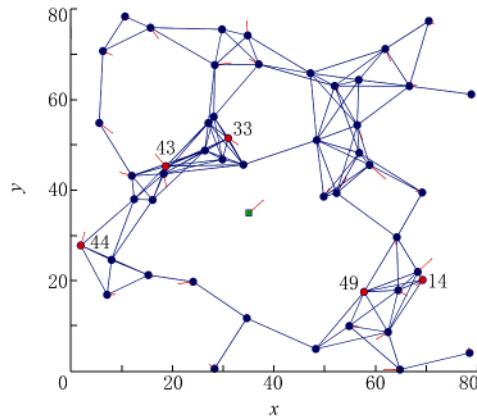
3 仿真结果与分析

3.1 无向拓扑结构的仿真

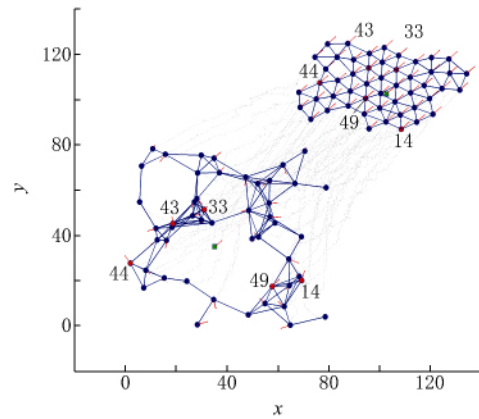
用 Matlab 对多智能体牵制蜂拥控制算法进行仿真,具体参数设置如下. 设多智能体总数为 50,智能体的位置在矩形区域 $[0, 80] \times [0, 80]$ 内随机生成,其速度和方向角分别在 $[1, 2] \times [1, 2]$ 和 $(-\pi, \pi)$ 的范围内随机生成;虚拟领导者的初始位置 $q_r(0) = [35, 35]$,速度 $p_r(0) = 0.35 \times [1,$

$\sqrt{3}]$. 智能体间的期望距离 $d = 8.75$,感知半径 $r = 1.2d$,反馈系数 $c_1 = 0.6, c_2 = 0.4, \epsilon = 0.1, h = 0.9$.

图 2(a) 为多智能体网络随机初始化的分布图. 蓝色实心圆圈表示普通智能体,红色实心圆圈表示牵制智能体,绿色正方形表示虚拟领导者. 本文牵制策略确定的牵制节点为 14, 33, 43, 44, 49. 图 2(b) 所示为多智能体的蜂拥运动轨迹,随着时间的增加,在本文提出的牵制策略下,最终智能体间达到稳定状态.



(a) $t=0$ s, 网络的初始分布

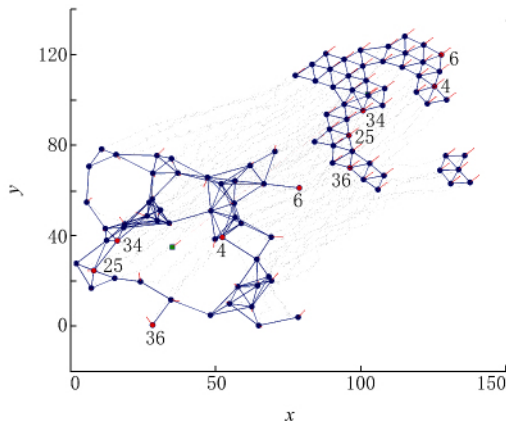


(b) $0 \leq t \leq 150$ s, 群体的运动轨迹

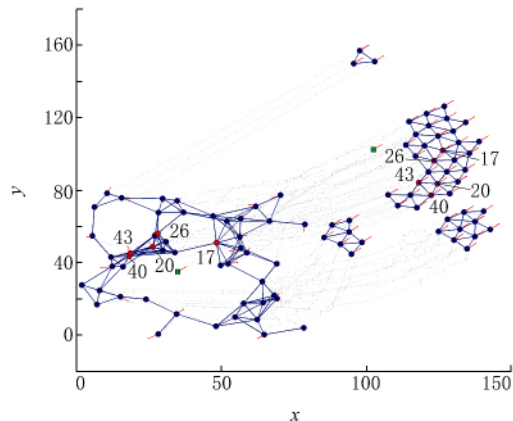
图 2 本文牵制策略下多智能体的蜂拥过程

图 3(a) 和 (b) 分别是随机牵制策略和度最大牵制策略的多智能体蜂拥运动轨迹,最终系统出现蜂拥控制失败的情形,不能实现速度一致,与图

2(b) 相比,在相同的初始条件下,本文提出的牵制策略的蜂拥控制效果要优于随机牵制策略和度最大牵制策略.



(a) 随机牵制策略



(b) 度最大牵制策略

图 3 $0 \leq t \leq 150$ s, 其他牵制策略下的多智能体蜂拥运动轨迹

在相同规模下,针对不同的初始网络,牵制节点也有所不同,分别采用随机牵制策略(策略 1)、度最大牵制策略(策略 2)和本文牵制策略(策略 3)进行蜂拥控制仿真,并统计分析 100 次独立仿真实验,结果如表 1 和表 2 所示,其中: η 为控制成功率; N 为网络的节点数; L 为初始网络的边数; d_N 为牵制节点比例,且有 $d_N = M/N$; T 为实现速度一致的平均时间.

表 1 表明:当网络规模相同时,随着网络紧密程度增加,控制成功率上升,平均一致时间也随之缩短. 从表 2 可知:不同的牵制策略对蜂拥一致的效果影响很大,随机牵制策略和度最大牵制策略虽然最终都实现了速度一致,但最终实现一致的平均时间比本文牵制策略大得多. 当 N 相同时, d_N 与网络的拓扑结构有关, L 一定时,不同的拓扑关系对应的 d_N 不同,且随着 d_N 的增加,实现

表 1 $N=50, d_N=0.1$ 时控制成功率统计表

牵制策略		策略 1	策略 2	策略 3
$\eta/\%$	$L=103$	80.3	90.8	100.0
	$L=133$	92.5	98.5	100.0
	$L=173$	99.5	100.0	100.0
T/s	$L=103$	143.6	126.0	78.9
	$L=133$	125.8	110.6	65.2
	$L=173$	112.2	94.5	51.5

表 2 $N=50$ 时各牵制策略的比较

L	d_N	T/s		
		策略 1	策略 2	策略 3
103	0.10	143.6	126.0	78.9
	0.12	140.0	123.2	75.6
	0.14	137.3	119.8	74.3
133	0.10	125.8	110.6	65.2
	0.12	121.5	109.2	63.1
	0.14	113.6	104.5	60.0
173	0.10	112.2	94.5	51.5
	0.12	110.5	89.0	50.8
	0.14	105.0	86.3	49.3

一致的平均时间略微减少,但减少的幅度很小,同时牵制节点比例增加在一定程度上增加了网络的控制成本;当 N 和 d_N 相同时,随着 L 增加,实现一致的平均时间减少,即致密的网络能更快地实现一致。

不同规模下多智能体系统运动规律的统计分析如下:为了进一步研究网络的规模 N 对系统运行效果的影响,假设智能体的位置在矩形区域 $[0, S] \times [0, S]$ 内随机生成, S 按分布密度 $\rho = N/S^2 = 0.004$ 确定,通过 100 次独立仿真实验,统计得出 N 与牵制节点比例平均值 \bar{d}_N 之间的关系如图 4 所示。

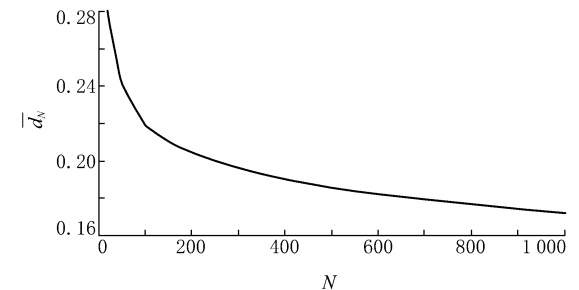


图 4 牵制节点比例平均值与网络规模的关系

图 4 表明:随着网络规模的增大,牵制节点比例的平均值逐渐减少。定义参数 $\delta = L/C_N^2$, 其中 $C_N^2 = 2/[N(N-1)]$ 其表征网络连接的紧密程度, δ 越大,连接越紧密。考虑 N 为 50, 100 和 500 时,对三种牵制策略的蜂拥效果进行比较并进行 100 次独立仿真实验,统计结果见表 3 和表 4。

表 3 表明:当网络规模不同时,随着网络紧密

程度增加,控制成功率上升,平均一致时间也随之缩短。50/56 表示网络规模为 50, 边数为 56 时,本文策略的控制成功率为 98.6%, 出现了控制失败的情况,这是由于稀疏、不连通的网络必定会影响牵制控制效率。

表 3 N 不同时控制成功率统计表

	N/L	策略 1	策略 2	策略 3
$\eta/\%$	50/56	70.6	80.5	98.6
	100/268	88.5	92.3	100.0
	500/7 582	97.5	100.0	100.0
T/s	50/56	203.1	139.3	85.3
	100/268	176.5	127.6	68.6
	500/7 582	141.3	110.2	43.8

表 4 N 不同时各牵制策略的比较

N/L	δ	\bar{d}_N	T/s		
			策略 1	策略 2	策略 3
50/56	0.045 7	0.24	203.1	139.3	85.3
100/268	0.054 1	0.22	176.5	127.6	68.6
500/7 582	0.060 8	0.186	141.3	110.2	43.8

表 4 给出了在不同网络规模下,不同牵制策略实现一致的平均时间。通过多次仿真实验发现:当初始网络拓扑不连通时,随着网络紧密程度的增加,速度收敛一致的平均时间减小。

3.2 有向拓扑结构的仿真

针对二阶有向多智能体网络进行仿真分析,由于有向网络的信息传递是单向的,随机牵制策略和度最大牵制策略未必能实现蜂拥行为,因此只考虑本文提出的牵制策略,具体的控制参数如下:智能体个数为 20, 初始拓扑结构如图 5 所示,其中智能体的位置在矩形区域 $[0, 25] \times [0, 25]$ 内随机生成,虚拟领导者的初始位置 $q_f(0) = [15,$

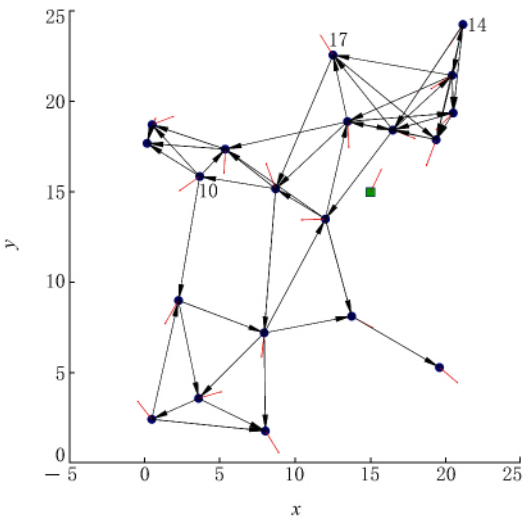


图 5 有向多智能体系统初始分布

15], 智能体间的期望距离 $d=7$, 感知半径 $r=1.2d$, 其余参数的设置与无向网络相同。

针对图5所示的拓扑结构,得出的牵制节点数为3,牵制节点为14,17,10.网络在演化过程中智能体之间最终达到稳定的状态,并在 $T=21.6$ s实现了收敛一致,如图6所示,图中 Δv 为速度差值.

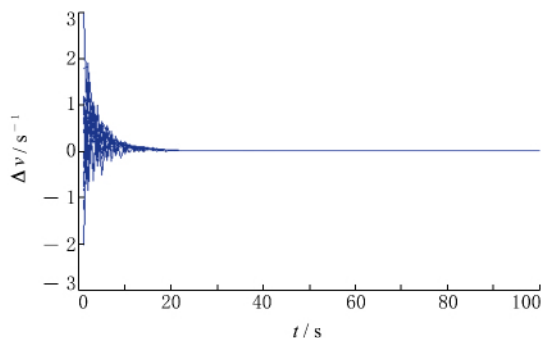


图6 智能体与虚拟领导者之间的速度差值

研究蜂拥控制的关键问题是如何提高控制效率以及牵制个体数目对蜂拥行为的影响.本文融合图论和控制理论的思想,提出了一种基于网络特征重数的牵制蜂拥控制算法,同时对比仿真随机牵制策略和度最大牵制策略.仿真实验发现:在相同条件下,本文的牵制策略能够更有效地提高多智能体牵制蜂拥控制的控制效率.

参 考 文 献

- [1] Reynolds C W. Flocks, herds, and schools; a distributed behavioral model [J]. Computer Graphics (ACM), 1987, 21(4): 25-34.
- [2] Semnani S H, Basir O A. Semi-flocking algorithm for motion control of mobile sensors in large-scale surveillance systems[J]. IEEE Transactions on Cybernetics, 2015, 45(1): 129-137.
- [3] Su H S, Zhang N Z, Chen M Z Q, et al. Adaptive flocking with a virtual leader of multiple agents governed by locally Lipschitz nonlinearity[J]. Nonlinear Analysis: Real World Applications, 2013, 14(1): 798-806.
- [4] Pei H Q, Chen S M, Lai Q. A local flocking algorithm of multi-agent dynamic systems[J]. International Journal of Control, 2015, 88(11): 2242-2249.
- [5] Olfati-Saber R. Flocking for multi-agent dynamic systems: algorithms and theory[J]. IEEE Transactions on Automatic Control, 2006, 51(3): 401-420.
- [6] Liu J G. Generalized projective synchronization of fractional-order complex networks with nonidentical nodes[J]. Chinese Physics: B, 2012, 21(12): 120506.
- [7] Tang Y, Wong W K, Fang J A, et al. Pinning impulsive synchronization of stochastic delayed coupled networks[J]. Chinese Physics: B, 2011, 20(4): 040513.
- [8] Su H S, Wang X F, Lin Z L. Flocking of multi-agents with a virtual leader[J]. IEEE Transactions on Automatic Control, 2009, 54(2): 293-307.
- [9] Wang X F, Li X, Lu J H. Control and flocking of networked systems via pinning[J]. IEEE Circuits and Systems Magazine, 2010, 10(3): 83-91.
- [10] Hengster-Movric K, You K Y, Lewis F L, et al. Synchronization of discrete-time multi-agent systems on graphs using Riccati design[J]. Automatica, 2013, 49(2): 414-423.
- [11] 刘智伟. 基于混杂控制的复杂多智能体网络同步一致性[D]. 武汉: 华中科技大学自动化学院, 2011.
- [12] Lu X Q, Francis A, Chen S H. Cluster consensus of second-order multi-agent systems via pinning control [J]. Chinese Physics: B, 2010, 19(12): 120506.
- [13] Hu H X, Yu W W, Xuan Q, et al. Consensus for second-order agent dynamics with velocity estimators via pinning control[J]. IET Control Theory and Applications, 2013, 7(9): 1196-1205.
- [14] Song Q, Cao J D, Yu W W. Second-order leader-following consensus of nonlinear multi-agent systems via pinning control[J]. Systems & Control Letters, 2010, 59(9): 553-562.
- [15] Song Q, Liu F, Cao J D, et al. M-matrix strategies for pinning-controlled leader-following consensus in multi-agent[J]. IEEE Transactions on Cybernetics, 2013, 43(6): 1688-1697.
- [16] Fiedler M. Algebraic Connectivity of graphs[J]. Czechoslovak Mathematical Journal, 1973, 23(2): 298-305.
- [17] Kim Y, Mesbahi M. On maximizing the second smallest eigenvalue of a state-dependent graph laplacian[J]. IEEE Transactions on Automatic Control, 2006, 51(1): 116-120.