

多智能体系统一致性跟踪的遗忘因子迭代学习方法

郑境易, 何熊熊, 伍益明, 秦贞华

信息工程学院, 浙江工业大学, 杭州310023
E-mail: hxx@zjut.edu.cn

摘要: 近些年来, 迭代学习控制被应用到多智能体系统编队和一致性跟踪等问题中. 本文对含有领导者的多智能体系统一致性跟踪建立模型, 提出了一种含有遗忘因子的迭代学习控制算法, 并给出了系统在所提算法下实现一致性跟踪的充分条件. 本文最后通过一个仿真实例, 验证了该算法的有效性.

关键词: 多智能体系统, 一致性跟踪, 迭代学习控制, 初值问题, 遗忘因子

Iterative Learning Control with a Forgetting Factor for Consensus Tracking in Multi-Agent Systems

Jingyi Zheng, Xiongiong He, Yiming Wu, Zhenhua Qin

College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China
E-mail: hxx@zjut.edu.cn

Abstract: Recently, iterative learning control (ILC) has been used in the field of the multi-agent systems. In this paper, we first propose a model for the leader-following multi-agent system consensus tracking problem from the perspective of ILC. ILC is used to ensure that all following agents perfectly track the only leader's trajectory in a finite time interval as the iterative number goes to infinity. Then, the forgetting factor is introduced into ILC to deal with the initial state shift. A sufficient condition is proposed to guarantee the forgetting factor ILC consensus tracking achievement. Finally a simulation example is given to show the effectiveness of the forgetting factor ILC.

Key Words: Multi-Agent System, Consensus Tracking, Iterative Learning Control, Initial Shift Problem, Forgetting Factor

1. 引言(Introduction)

近些年来, 多智能体系统的协同控制问题得到了国内外学者的广泛关注, 学者们对此问题进行了深入地研究, 并取得相当可观的研究成果[1-5]. 多智能体系统协同控制在无人机(UAV)协调合作、多机器人编队、无线传感器网络信息融合和卫星姿势调整等领域中有着广泛的应用. 一致性问题作为实现协同控制的基础, 是指在多个智能体的协调合作过程中, 协作所需要的某类信息(如智能体的状态、位移、速度、加速度等), 在智能体之间传递, 并按照一致性协议进行更新, 最终达成一致.

自然界中, 领导者(leader)对整个集群(种群)的运动有着重要的作用, 它往往能够引导集群的运动

方向, 并确定移动速度的大小. 受此种现象启发, 广大学者研究了多智能体系统为领导者-跟随者(leader-follower)结构下, 各跟随者对领导者的跟踪情况[6-8]. 研究表明, 满足一定的拓扑条件, 每个跟随者都能渐近地跟踪领导者的状态.

近些年来, 迭代学习控制被应用到多智能体系统编队^[9]和一致性跟踪^[10]等问题中. 迭代学习控制^[11-12]的优点在于其可以不断利用上一次的输入和输出偏差信息来调整此次输入, 使得最终的输出曲线与期望曲线相同, 从而实现在有限时间区间 $[0, T]$ 上对期望曲线的完美跟踪.

2009年, 韩国学者Ahn^[9]针对重复运行的多智能体系统的编队问题, 首次在卫星编队中引入迭代学习控制实现轨迹的完全跟踪. 随后, Liu^[13]考虑了在一般拓扑情形下的多智能体编队问题. Yang^[14]则在多智能体系统一致性迭代学习控制中, 引入最优增益

此项工作得到国家自然科学基金资助, 项目批准号: 61473262, 得到浙江省社会公益性项目资助, 项目批准号: 2014C31097.

的概念, 并提出最优增益的计算方法, 从而加快了跟踪速度, 减少了跟踪到目标轨迹的迭代次数.

ILC需要初值的完全重置, 当存在初值偏移时, 将不能实现对目标轨迹的完美跟踪, 甚至会导致实际轨迹的发散. 然而, 在实际情况下, 多智能体系统的独立性特点导致了初值问题的存在, 每个智能体的初值通常不会都等于一个确定值. Meng^[15]在连续系统中应用初值整形技术, 从而能够在任意短时间 τ 后, 开始跟踪到目标轨迹. Yang^[16]在多智能体系统中应用带有初值学习的D型学习律, 选择适当的学习增益, 有效地克服了初值偏移问题. 伍凤山^[17]考虑了多智能体一致性迭代学习控制的初值问题, 在存在系统噪声和量测噪声的情况下, 使用带有初值学习的PD型学习律, 并将之应用到机械臂中, 展示了算法的有效性和可行性. 然而以上迭代学习律需要对初始时刻的状态进行学习, 在实际运行中, 可能无法直接对系统状态的初值进行更改. 针对此种情况, Yang^[18]在存在初值偏移的情况下, 使用D型和PD型学习律, 各跟随者的轨迹最终会收敛到一条新的轨迹上, 然而新轨迹不同于目标智能体的期望轨迹, 两者存在差异. Meng^[19]在存在系统噪声和量测噪声的情况下使用二维模型对系统进行建模, 利用LMI技术设计了鲁棒学习律, 实现了多智能体系统的一致性跟踪.

针对多智能体系统迭代学习控制中的初值问题, 本文对领导者-跟随者的多智能体系统一致性跟踪问题建立模型, 只有部分跟随智能体能够获得领导者的信息, 提出含有遗忘因子的迭代学习控制算法^[20], 使得所有跟随智能体都能在固定时间区间内, 跟踪领导智能体的运动轨迹. 该算法不要求状态的初值等于期望轨迹的初值, 并且加快了跟踪的速度. 在初始输入接近目标输入的情况下, 该算法能够防止过大的跟踪误差出现. 该算法在系统存在初值误差的情况下, 仍能保证系统的输出轨迹收敛到一条固定轨迹上. 本文最后通过一个仿真来验证算法的可行性.

2. 预备知识

2.1 图论

图论是描述网络中信息传播的有效工具. 可以用图中的节点代表智能体, 用点跟点之间的边来表示两个智能体之间的通信情况, 边的权值表示为信息的影响力.

图 $\mathcal{G}=(\mathcal{V}, \mathcal{E}, A_{\mathcal{G}})$ 表示为 N 个智能体之间的加权有向图, 其中 $\mathcal{V}$ 为节点集, $\mathcal{E}$ 为边集, $A_{\mathcal{G}}$ 为邻接矩阵. 图的每个顶点 $v_i \in \mathcal{V}$, $i=1, 2, \dots, N$ 表示为每个智能体, 图的每条边表示为 $e_{ij}=\langle v_i, v_j \rangle$ 由顶点 v_i 指向 v_j , 表示 v_j 收到 v_i 的信息. 称 v_i 为 v_j 的父(parent)节点.

$A_{\mathcal{G}}=[a_{ij}]$ 为邻接矩阵, 其中 $a_{ji}>0 \Leftrightarrow \langle v_i, v_j \rangle \in \mathcal{E}$, $a_{ji}=0 \Leftrightarrow \langle v_i, v_j \rangle \notin \mathcal{E}$, $i \neq j$. 本文中, 图不存在自环, 即 $a_{ii}=0$, $\forall i=1, \dots, N$. 节点 v_i 的邻居集 $\mathcal{N}_i=\{j: \langle v_j, v_i \rangle \in \mathcal{E}\}$ 是由节点 v_i 的所有父节点所组成的集合. $\Delta_i=\sum_{j=1, \dots, N} a_{ij}$ 称为 v_i 的入度, $\Delta=\text{diag}(\Delta_1, \Delta_2, \dots, \Delta_N)$, 图的Laplacian矩阵定义为 $L=\Delta-A_{\mathcal{G}}$. 若在节点 v_i 与节点 v_j 之间, 能够由多个相连的边按顺序 $p: \langle v_i, v_k \rangle \langle v_k, v_m \rangle \dots \langle v_s, v_j \rangle$ 连接, 则称 v_i 存在一条路径(path)到达 v_j .

生成树(spanning tree)是一种特殊的有向图, 图中除了一个根(root)节点外, 其他的节点都只有1个父节点, 并且根节点不存在父节点. 若在图中能用边集 $\mathcal{E}$ 的子集将所有顶点连接起来, 构成一生成树, 则称图 $\mathcal{G}=(\mathcal{V}, \mathcal{E}, A_{\mathcal{G}})$ 包含有一生成树.

2.2 代数知识

$\mathbf{R}$, $\mathbf{R}^n$, $\mathbf{R}^{m \times n}$ 分别表示为实数, 行数为 n 的列向量, 和 m 行 n 列的矩阵. A^T 表示矩阵 A 的转置. I_n 为 $n \times n$ 的单位矩阵, $\mathbf{1}_n$ ($\mathbf{0}_n$)表示为行数为 n , 数值都为1(0)的列向量, $\otimes$ 表示kronecker内积.

给定一个向量 $\mathbf{x}=[x_1, x_2, \dots, x_n]^T \in \mathbf{R}^n$, $\|\mathbf{x}\|$ 为相应范数, 其中 $\|\mathbf{x}\|_1=\sum_{k=1}^n |x_k|$, $\|\mathbf{x}\|_2=\sqrt{\mathbf{x}^T \mathbf{x}}$, $\|\mathbf{x}\|_{\infty}=\max_{k=1, 2, \dots, n} |x_k|$.

给定一个矩阵 $A=[a_{ij}] \in \mathbf{R}^{m \times n}$, $\|A\|$ 为相应的诱导矩阵范数, $\|A\|_2=\sqrt{A^T A}$ 为矩阵的谱范数, $\|A\|_{\infty}=\max_{1 \leq i \leq m} \sum_{j=1}^n |a_{ij}|$. $\rho(A)$ 为 A 的谱半径.

向量函数 $f(\cdot)$ 的sup范数为 $\|f(t)\|=\max_{t \in [0, T]} \|f(t)\|$, 令 λ 为一个正数, 定义函数的 λ 范数为 $\|f(t)\|_{\lambda}=\max_{t \in [0, T]} e^{-\lambda t} \|f(t)\|$. 有

$$\|f(t)\|_{\lambda} \leq \|f(t)\| \leq e^{\lambda t} \|f(t)\|_{\lambda}.$$

3. 问题描述

本文考虑的多智能体系统由1个领导智能体(以下简称领导者)与 N 个跟随者智能体(以下简称跟随者)组成, 并且只有部分跟随者得到领导者的输出信息.

令跟随者编号为 $1, 2, \dots, N$, 由所有跟随者组成的图表示为 $\mathcal{G}=(\mathcal{V}, \mathcal{E}, A_{\mathcal{G}})$. 令领导者编号为0, 用 d_i 表示智能体 v_i 获得领导者 v_0 状态的权重值. 若智能体 v_i 能够获得领导者状态, 则 $d_i>0$, 否则 $d_i=0$.

令包含领导者和所有跟随者的新图表示为 $\bar{\mathcal{G}} = (\mathcal{V} \cup v_0, \bar{\mathcal{E}}, \bar{A}_{\mathcal{G}})$, 其中 $\bar{\mathcal{E}} = \mathcal{E} \cup \langle v_0, v_i \rangle$, $\forall i \in 1, 2, \dots, N$, $\bar{A}_{\mathcal{G}} = \begin{bmatrix} 0 & \mathbf{0}_N^T \\ d & A_{\mathcal{G}} \end{bmatrix}$, $d = (d_1, d_2, \dots, d_N)^T$, $D = \text{diag}(d_1, d_2, \dots, d_N)$.

假定1: 图 $\bar{\mathcal{G}} = (\mathcal{V} \cup 0, \bar{\mathcal{E}})$ 中包含生成树, 且0号节点是生成树的根.

假定1等同于在图 $\bar{\mathcal{G}} = (\mathcal{V} \cup 0, \bar{\mathcal{E}})$ 中, 0号节点全局可达.

令领导者的更新方程为

$$\begin{cases} \dot{x}_0(t) = f(t, x_0(t)) + Bu_0(t) \\ y_0(t) = Cx_0(t) \end{cases}, \quad (1)$$

其中, $x_0(t) \in \mathbf{R}^s$ 表示领导者的内部状态, $u_0(t) \in \mathbf{R}^n$ 表示为领导者的输入, $y_0(t) \in \mathbf{R}^m$ 表示领导者的输出.

类似地, 令跟随者的更新方程为

$$\begin{cases} \dot{x}_i(t) = f(t, x_i(t)) + Bu_i(t) \\ y_i(t) = Cx_i(t) \end{cases}, i \in 1, 2, \dots, N. \quad (2)$$

假定2: $f(t, x_i(t))$ 满足全局Lipschitz连续条件, 即 $\|f(t, x_1(t)) - f(t, x_2(t))\| \leq b_f \|x_1(t) - x_2(t)\|$.

假定3: CB 列满秩.

假定4: 跟随者的状态初值能够重置, 然而跟随者的初值与领导者的初值不尽相同, 即完美初值条件 (IIC, Identical Initial Condition) $x_i(0) = x_0(0), \forall i \in 1, 2, \dots, N$ 不能满足.

在多智能体系统中, 给定目标智能体的期望输出为 $y_0(t)$, 若从 t_0 时刻 (不失一般性, 令 $t_0 = 0$) 开始到 $t = T$ 时刻结束, 在这段时间范围内, 能够完全跟踪到 y_0 的数值, 即

$$y_i(t) = y_0(t), \forall t \in [0, T], i = 1, 2, \dots, N. \quad (3)$$

则称为所有跟随智能体能够在固定时间区间内, 实现对领导智能体的完全一致性跟踪.

在多智能体系统中, 通常只有部分跟随智能体能够得到领导者的值 $y_0(t)$. 对于编号为 i 的智能体, 收到的数据只有邻居的输出 $y_j(t)$, 其中 $v_j \in \mathcal{N}_i$. 智能体把自身的输出和接收到的邻居智能体的输出按照下式进行运算

$$\xi_i(t) = \sum_{j \in \mathcal{N}_i} a_{ij}(y_j(t) - y_i(t)) + \sum d_i(y_0(t) - y_i(t)) \quad (4)$$

令

$$e_i(t) = y_0(t) - y_i(t). \quad (5)$$

把式(4)与式(5)写成紧凑模式,

$$\begin{aligned} \xi(t) &= [\xi_1^T(t), \xi_2^T(t), \dots, \xi_N^T(t)]^T, \\ E(t) &= [e_1^T(t), e_2^T(t), \dots, e_N^T(t)]^T, \end{aligned}$$

由式(4)和式(5)可得

$$\xi_i(t) = \sum_{j \in \mathcal{N}_i} a_{ij}(e_i(t) - e_j(t)) + \sum d_i e_i(t). \quad (6)$$

把式(6)写成紧凑模式, 则

$$\begin{aligned} \xi(t) &= (L \otimes I_m)E(t) + (D \otimes I_m)E(t) \\ &= [(L + D) \otimes I_m]E(t) \end{aligned} \quad (7)$$

引入迭代轴 k , 采用遗忘因子迭代学习律对输入信号进行更新, 则

$$u_{k+1,i}(t) = (1 - \alpha_i)u_{k,i}(t) + \alpha_i u_{0,i}(t) + \Gamma_i \dot{\xi}_{k,i}(t) \quad (8)$$

其中, $\Gamma_i \in \mathbf{R}^{n \times m}$ 为待定的学习增益, $0 \leq \alpha_i < 1$ 为遗忘因子, $u_{0,i}(t)$ 为第 i 个跟随智能体的初始输入.

令

$$\begin{aligned} U_k(t) &= [u_{k,1}^T(t), u_{k,2}^T(t), \dots, u_{k,N}^T(t)]^T, \\ X_k(t) &= [x_{k,1}^T(t), x_{k,2}^T(t), \dots, x_{k,N}^T(t)]^T, \\ Y_k(t) &= [y_{k,1}^T(t), y_{k,2}^T(t), \dots, y_{k,N}^T(t)]^T, \\ \Phi &= \text{diag}[\alpha_1, \alpha_2, \dots, \alpha_N], \Gamma = \text{diag}[\Gamma_1, \Gamma_2, \dots, \Gamma_N], \end{aligned}$$

则式(2)写成紧凑形式,

$$\begin{cases} \dot{X}_k(t) = F(t, x(t)) + (I_N \otimes B)U_k(t) \\ Y_k(t) = (I_N \otimes C)X_k(t) \end{cases} \quad (9)$$

其中, $F(t, x(t)) = \text{diag}(f(t, x_1(t)), \dots, f(t, x_N(t)))$.

令 $U_0(t) = [u_{0,1}^T(t), u_{0,2}^T(t), \dots, u_{0,N}^T(t)]^T$ 为初次输入, 学习律(8)写为

$$\begin{aligned} U_{k+1}(t) &= [(I_N - \Phi) \otimes I_n]U_k(t) + [\Phi \otimes I_n]U_0(t) \\ &\quad + \Gamma[(L + D) \otimes I_m]\dot{E}_k(t) \end{aligned} \quad (10)$$

4. 主要内容

定理1: 考虑式 (2) 的系统, 和式 (8) 的学习律, 在满足假定1-4的情况下, 如果存在合适的遗忘因子矩阵 Φ , 和增益矩阵 Γ , 满足

$$\rho((I_N - \Phi) \otimes I_n - \Gamma[(L + D) \otimes I_m](I_N \otimes CB)) < 1 \quad (11)$$

则 $\|u_0 - u_i\|$, $\|y_0 - y_i\|$ 有界. 并且界的范围跟 α_i 和初值误差的大小相关.

证明: 第 $k+1$ 次的输入误差为

$$\begin{aligned} u_0(t) - u_{i+1}(t) &= (1 - \alpha_i)(u_0(t) - u_{k,i}(t)) \\ &\quad + \alpha_i(u_0(t) - u_{0,i}(t)) - \Gamma_i \dot{\xi}_{k,i}(t) \end{aligned} \quad (12)$$

把式(15)写成紧凑模式,

$$\begin{aligned} \mathbf{1} \otimes u_0(t) - U_{k+1}(t) &= [(I_N - \Phi) \otimes I_n](\mathbf{1} \otimes u_0(t) - U_k(t)) \\ &\quad + [\Phi \otimes I_n](\mathbf{1} \otimes u_0(t) - U_0(t)) \\ &\quad - \Gamma[(L + D) \otimes I_m]\dot{E}_k(t) \end{aligned}$$

令

$$\Delta U_k(t) = \mathbf{1}_N \otimes u_0(t) - U_k(t). \quad (13)$$

为了书写和阅读简便, 以下证明略去时间符号 t .
令

$$\tilde{\Gamma} = \Gamma[(L + D) \otimes I_m]$$

则

$$\begin{aligned} \Delta U_{k+1} &= [(I_N - \Phi) \otimes I_n] \Delta U_k + \\ &\quad [\Phi \otimes I_n] \Delta U_0 - \Gamma[(L + D) \otimes I_m] \dot{E}_k \\ &= [(I_N - \Phi) \otimes I_n] \Delta U_k + \\ &\quad [\Phi \otimes I_n] \Delta U_0 - \tilde{\Gamma} \dot{E}_k \\ \dot{E}_k &= \mathbf{1} \otimes \dot{y}_0 - \dot{Y}_k \\ &= \mathbf{1} \otimes (Cf(x_0) + CBu_0) - \{(I_N \otimes C)[f(X_k) + \\ &\quad (I_N \otimes B)U_k]\} \\ &= \mathbf{1} \otimes (Cf(x_0) + CBu_0) - [(I_N \otimes C)f(X_k) + \\ &\quad (I_N \otimes CB)U_k] \\ &= (I_N \otimes C)[(\mathbf{1} \otimes f(x_0)) - f(X_k)] + \\ &\quad (I_N \otimes CB)[\mathbf{1} \otimes u_0 - U_k] \end{aligned} \quad (14)$$

由式(13)与式(14)可得

$$\begin{aligned} \Delta U_{k+1} &= [(I_N - \Phi) \otimes I_n - \tilde{\Gamma}(I_N \otimes CB)] \Delta U_k \\ &\quad + [\Phi \otimes I_n] \Delta U_0 \\ &\quad - \tilde{\Gamma}(I_N \otimes C)[(\mathbf{1} \otimes f(x_0)) - f(X_k)] \end{aligned} \quad (15)$$

对上式左右两边取范数, 可得

$$\begin{aligned} \|\Delta U_{k+1}\| &\leq \|(I_N - \Phi) \otimes I_n - \tilde{\Gamma}(I_N \otimes CB)\| \|\Delta U_k\| \\ &\quad + \|\Phi \otimes I_n\| \|\Delta U_0\| \\ &\quad + \|\tilde{\Gamma}\| \left\{ \|I_N \otimes C\| \left\| [\mathbf{1} \otimes f(x_0) - f(X_k)] \right\| \right\} \end{aligned} \quad (16)$$

$$\text{令 } \rho_K = \|(I_N - \Phi) \otimes I_n - \tilde{\Gamma}(I_N \otimes CB)\|,$$

则式(16)可写为

$$\begin{aligned} \|\Delta U_{k+1}\| &\leq \rho_K \|\Delta U_k\| + \|\Phi \otimes I_n\| \|\Delta U_0\| \\ &\quad + b_\Gamma \left\{ b_c \left[k_f \|\mathbf{1} \otimes x_0 - X_k\| \right] \right\} \\ &\leq \rho_K \|\Delta U_k\| + \|\Phi \otimes I_n\| \|\Delta U_0\| \\ &\quad + k_1 \|\mathbf{1} \otimes x_0 - X_k\| \end{aligned} \quad (17)$$

其中 $k_1 = b_\Gamma b_c b_f$.

接着计算 $\|\mathbf{1} \otimes x_0 - X_k\|$.

$$\begin{aligned} \|\mathbf{1} \otimes x_0 - X_k\| &\leq \left\| \mathbf{1} \otimes x_0(0) - X_k(0) + \int_0^t \begin{pmatrix} (f(\mathbf{1} \otimes x_0) + B(\mathbf{1} \otimes u_0)) \\ -(f(X_k) + BU_k) \end{pmatrix} d\tau \right\| \\ &\leq \|\mathbf{1} \otimes x_0(0) - X_k(0)\| + \int_0^t \left(\|f(\mathbf{1} \otimes x_0) - f(X_k)\| \right. \\ &\quad \left. + \|B\| \|\mathbf{1} \otimes u_0 - U_k\| \right) d\tau \\ &\leq \|\mathbf{1} \otimes x_0(0) - X_k(0)\| + \int_0^t \left(b_f \|\mathbf{1} \otimes x_0 - X_k\| \right. \\ &\quad \left. + b_B \|\mathbf{1} \otimes u_0 - U_k\| \right) d\tau \\ &\leq \|\mathbf{1} \otimes x_0(0) - X_k(0)\| e^{k_f t} + \int_0^t e^{k_f(t-\tau)} (b_B \|\mathbf{1} \otimes u_0 - U_k\|) d\tau \end{aligned} \quad (18)$$

由式(17)与式(18), 可得

$$\begin{aligned} \|\Delta U_{k+1}\| &\leq \rho_K \|\Delta U_k\| + b_\Phi \|\Delta U_0\| \\ &\quad + k_1 \left\{ \|\mathbf{1} \otimes x_0(0) - X_k(0)\| e^{k_f t} \right. \\ &\quad \left. + \int_0^t e^{k_f(t-\tau)} (b_B \|\mathbf{1} \otimes u_0(\tau) - U_k(\tau)\|) d\tau \right\} \\ &\leq \rho_K \|\Delta U_k\| + b_\Phi \|\Delta U_0\| + k_1 \|\Delta X_k(0)\| e^{k_f t} \\ &\quad + k_1 b_B \int_0^t e^{k_f(t-\tau)} \Delta U_k(\tau) d\tau \end{aligned} \quad (19)$$

令 $k = \max(k_1, k_f)$, 上式两边同时乘以 $e^{-\lambda t}$, 则

$$\begin{aligned} e^{-\lambda t} \|\Delta U_{k+1}\| &\leq \rho_K e^{-\lambda t} \|\Delta U_k\| + b_\Phi e^{-\lambda t} \|\Delta U_0\| \\ &\quad + k_1 \|\Delta X_k(0)\| e^{(k_f - \lambda)t} \\ &\quad + k b_B \int_0^t e^{-\lambda t} e^{k(t-\tau)} \Delta U_k(\tau) d\tau \end{aligned} \quad (20)$$

根据 λ 范数的定义, 式(20)可写为

$$\begin{aligned} \|\Delta U_{k+1}\|_\lambda &\leq \rho_K \|\Delta U_k\|_\lambda + b_\Phi \|\Delta U_0\|_\lambda \\ &\quad + k_1 \|\Delta X_k(0)\| + \frac{k}{\lambda - k} (1 - e^{(k - \lambda)T}) \|\Delta U_k\|_\lambda \\ &= \left(\rho_K + \frac{k}{\lambda - k} (1 - e^{(k - \lambda)T}) \right) \|\Delta U_k\|_\lambda \\ &\quad + b_\Phi \|\Delta U_0\|_\lambda + k_1 \|\Delta X_k(0)\| \end{aligned} \quad (21)$$

$$\text{令 } \tilde{\rho} = \left(\rho_K + \frac{k}{\lambda - k} (1 - e^{(k - \lambda)T}) \right),$$

则

$$\|\Delta U_{k+1}\|_\lambda \leq \tilde{\rho} \|\Delta U_k\|_\lambda + b_\Phi \|\Delta U_0\|_\lambda + k_1 \|\Delta X_k(0)\| \quad (22)$$

取一个足够大的正数 M , 使得

$$M \geq b_\Phi \|\Delta U_0\|_\lambda + k_1 \|\Delta X_k(0)\|$$

则式(22)可写成

$$\|\Delta U_{k+1}\|_\lambda \leq \tilde{\rho} \|\Delta U_k\|_\lambda + M \quad (23)$$

由于 $\rho_K < 1$, 总可以找到一个足够大的 λ , 满足 $\tilde{\rho} < 1$. 因此

$$\lim_{k \rightarrow \infty} \|\Delta U_k\|_\lambda \leq \left(\frac{1}{1 - \tilde{\rho}} \right) M \quad (24)$$

即 $\lim_{k \rightarrow \infty} U_k \rightarrow \mathbf{1} \otimes u_0 + \left(\frac{1}{1 - \tilde{\rho}} \right) M$, 表明最终输入会收敛到期望输入的邻域内.

同理, 可以由

$$\begin{aligned} \|\Delta X_k\|_\lambda &= \|\mathbf{1} \otimes x_0 - X_k\|_\lambda \\ &\leq \|\Delta X_k(0)\| + \int_0^t e^{(k_f - \lambda)(t-\tau)} \|\Delta U_k\|_\lambda d\tau \\ &\leq \|\Delta X_k(0)\| + \frac{1}{\lambda - k_f} (1 - e^{(k_f - \lambda)T}) \|\Delta U_k\|_\lambda \end{aligned} \quad (25)$$

得到 $\|\Delta X_k\|_\lambda$ 是有界的.

同理, 可以由

$$\begin{aligned}\|\Delta Y_k\|_\lambda &= \|\mathbf{1} \otimes y_0 - Y_k\|_\lambda \\ &\leq \|(I \otimes C)(\mathbf{1} \otimes x_0 - X_k)\|_\lambda \\ &\leq \|(I \otimes C)\| \|\Delta X_k\|_\lambda\end{aligned}\quad (26)$$

得到 $\|\Delta Y_k\|_\lambda$ 有界。

证毕■

特别地, 当 $\|\Delta X_k(0)\| = 0$, $\alpha = 0$ 时, 即所有初值误差都为0, 且遗忘因子为0时, 有

$$\begin{aligned}\|\Delta U_{k+1}\|_\lambda &\leq \tilde{\rho} \|\Delta U_k\|_\lambda \\ \lim_{k \rightarrow \infty} \|\Delta U_k\|_\lambda &\rightarrow 0, \\ \lim_{k \rightarrow \infty} \|\Delta Y_k\|_\lambda &\rightarrow 0.\end{aligned}$$

说明, 此时输入即为领导者的输入, 输出轨迹能完全重合。

5. 仿真

本文采用一个简单例子来验证本文所提算法的可行性. 考虑一个由1个领导者和5个跟随者所组成的多智能体系统, 领导者和跟随者之间的通信拓扑图如图1所示,

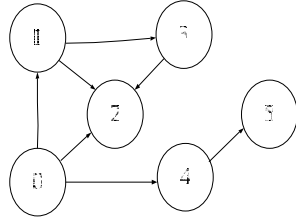


图1 通信拓扑图

为了计算简便, 在仿真中只取0-1权值, 相应的邻接矩阵和Laplacian矩阵如下所示

$$A_G = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

取 $D = \text{diag}(d_1, d_2, d_3, d_4, d_5) = \text{diag}(1, 1, 0, 1, 0)$

跟随者的更新方程为

$$\begin{cases} \begin{bmatrix} x_{k,i_1}(t) \\ x_{k,i_2}(t) \end{bmatrix} = \begin{bmatrix} x_{k,i_1}(t) + x_{k,i_2}(t) \\ \cos(x_{k,i_1}(t)) \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} u_{k,i_1}(t) \\ u_{k,i_2}(t) \end{bmatrix} \\ y_{k,i}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k,i_1}(t) \\ x_{k,i_2}(t) \end{bmatrix} \end{cases} \quad (27)$$

其中 $i = 1, 2, 3, 4, 5$ 为跟随智能体的编号, $k \in \mathbf{Z}$ 为迭代次数。

$$\text{目标轨迹 } y_0(t) = \begin{bmatrix} t^2 - t \\ t \end{bmatrix}, \quad t \in [0, 1]$$

遗忘因子的值分别为

$$\begin{aligned}\Phi &= \text{diag}(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) \\ &= \text{diag}(0.02, 0.04, 0.06, 0.08, 0.1)\end{aligned}\quad (28)$$

学习增益由文[14]的方法计算得到, 为

$$\begin{aligned}\Gamma &= \frac{2}{\lambda_{1(L+D)} + \lambda_{5(L+D)}} (CB)^{-1} \\ \Gamma_1 &= \Gamma_2 = \Gamma_3 = \Gamma_4 = \Gamma_5 = \begin{pmatrix} -0.3 & 0.2 \\ 0.4 & -0.1 \end{pmatrix}\end{aligned}\quad (29)$$

首次输入为

$$\begin{bmatrix} u_{0,1}(t) & u_{0,2}(t) & u_{0,3}(t) & u_{0,4}(t) & u_{0,5}(t) \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

初值点

$$\begin{bmatrix} x_1(0) & x_2(0) & x_3(0) & x_4(0) & x_5(0) \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

此时,

$$\begin{aligned}\rho((I_N - \Phi) \otimes I_{n \times n} - \Gamma[(L + D) \otimes I_{m \times m}](I_{N \times N} \otimes CB)) \\ = 0.54\end{aligned}$$

满足收敛条件式(11)。

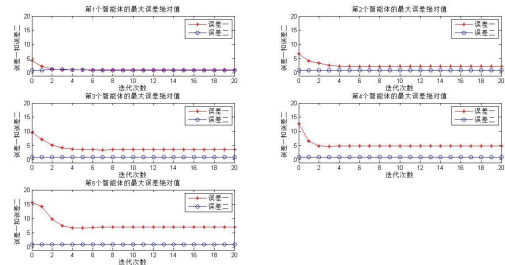


图2 存在初值误差时的输出最大误差绝对值

图2为存在初值误差时, 20次仿真对应的各智能体的输出值与期望值的最大误差. 由图2可以看出, 各跟随者在6次迭代后, 误差趋于一个非0的固定值, 不再变动. 这说明各跟随者的轨迹最终会收敛一条轨迹上, 然而该轨迹跟领导者的期望轨迹不能完美重合。

当初始误差为0时, 此时

$$x_1(0) = x_2(0) = x_3(0) = x_4(0) = x_5(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

由图3可以看出, 当初始误差为0时, 经过多次迭代, 相比于存在误差时, 各跟随者跟踪误差更大, 但仍不为0. 这是由于遗忘因子的存在, 使得挑选的初始输入对多次迭代后的输入信号仍有影响, 从而无法生

成正确的输入新号. 由图3可以看出, 随着迭代次数的增加, 跟随者1和4能较快地跟踪到期望轨迹, 而跟随者2需要更多的迭代次数才能实现完美跟踪. 各跟随者的收敛速度是受领导者的信息传递顺序所影响.

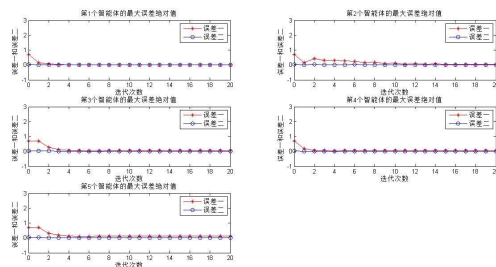


图3 无初值误差时的输出最大误差绝对值

6. 总结

本文针对多智能体系统的初值问题, 建立了含有领导者的多智能体系统一致性跟踪模型, 提出一种含有遗忘因子的迭代学习控制算法, 使得所有跟随智能体都能在有限时间内, 对领导智能体实现跟踪. 最后通过一个仿真实例, 验证了本文算法的有效性.

参考文献

- [1] Jadbabaie A, Lin J, Morse A S. Coordination of groups of mobile autonomous agents using nearest neighbor rules[J]. Automatic Control, IEEE Transactions on, 2003, 48(6): 988-1001.
- [2] Olfati-Saber R, Murray R M. Consensus problems in networks of agents with switching topology and time-delays[J]. Automatic Control, IEEE Transactions on, 2004, 49(9): 1520-1533.
- [3] Ren W, Beard R W. Consensus seeking in multiagent systems under dynamically changing interaction topologies[J]. IEEE Transactions on automatic control, 2005, 50(5): 655-661.
- [4] Cheng L, Hou Z G, Tan M, et al. Neural-network-based adaptive leader-following control for multi-agent systems with uncertainties[J]. Neural Networks, IEEE Transactions on, 2010, 21(8): 1351-1358.
- [5] Cao Y, Yu W, Ren W, et al. An overview of recent progress in the study of distributed multi-agent coordination[J]. Industrial Informatics, IEEE Transactions on, 2013, 9(1): 427-438.
- [6] Hong Y, Hu J, Gao L. Tracking control for multi-agent consensus with an active leader and variable topology[J]. Automatica, 2006, 42(7): 1177-1182.
- [7] Ren W. Multi-vehicle consensus with a time-varying reference state[J]. Systems & Control Letters, 2007, 56(7): 474-483.
- [8] Wang Y, Cheng L, Ren W, et al. Containment control of multi-agent systems with dynamic leaders based on a PI-type approach[J]. IEEE Transactions on Cybernetics, in press, DOI: 10.1109/TCYB.2015.2494738.
- [9] Ahn H S, Chen Y Q. Iterative learning control for multi-agent formation[C]//ICCAS-SICE, 2009. IEEE, 2009: 3111-3116.
- [10] Meng D, Jia Y. Iterative learning approaches to design finite-time consensus protocols for multi-agent systems[J]. Systems & Control Letters, 2012, 61(1): 187-194.
- [11] Moore K L, Chen Y Q, Ahn H S. Iterative learning control: A tutorial and big picture view[C]//Proceedings of the 45th IEEE Conference on Decision and Control. 2006: 2352-2357.
- [12] Xu J X. A survey on iterative learning control for nonlinear systems[J]. International Journal of Control, 2011, 84(7): 1275-1294.
- [13] Liu Y, Jia Y. An iterative learning approach to formation control of multi-agent systems[J]. Systems & Control Letters, 2012, 61(1): 148-154.
- [14] Yang S, Xu J X, Huang D, et al. Optimal iterative learning control design for multi-agent systems consensus tracking[J]. Systems & Control Letters, 2014, 69: 80-89.
- [15] Meng D, Jia Y, Du J, et al. Tracking control over a finite interval for multi-agent systems with a time-varying reference trajectory[J]. Systems & Control Letters, 2012, 61(7): 807-818.
- [16] Yang S, Xu J X, Huang D. Iterative learning control for multi-agent systems consensus tracking[C]//Decision and Control (CDC), 2012 IEEE 51st Annual Conference on. IEEE, 2012: 4672-4677.
- [17] 伍巧凤. 多智能体一致性迭代学习控制研究[D]. 浙江大学, 2014.
- [18] Yang S, Xu J X, Ren Q. Multi-agent consensus tracking with initial state error by iterative learning control[C]//Control & Automation (ICCA), 11th IEEE International Conference on. IEEE, 2014: 625-630.
- [19] Meng D, Jia Y, Du J. Robust Consensus Tracking Control for Multiagent Systems With Initial State Shifts, Disturbances, and Switching Topologies[J]. Neural Networks and Learning Systems, IEEE Transactions on, 2015, 26(4): 809-824.
- [20] Heinzinger G, Fenwick D, Paden B, et al. Robust learning control[C]//Decision and Control, 1989., Proceedings of the 28th IEEE Conference on. IEEE, 1989: 436-440. [2] Olfati-Saber R, Murray R M. Consensus problems in networks of agents with switching topology and time-delays[J]. Automatic Control, IEEE Transactions on, 2004, 49(9): 1520-1533.