DOI:10.13878/j.cnki.jnuist.2019.04.005

王瑞峰[1]　丰建文[1]

基于事件的非周期间歇牵制控制线性复杂网络的同步分析

摘要

本文将事件激发和非周期间歇牵制控制进行有机结合,分析了一般线性复杂网络的全局同步问题.通过设计事件激发条件和受控节点选择规则,使受牵制控制节点集在满足一定条件下实现更新,从而不仅极大提高网络实现同步的效率,而且避免现有研究中受控节点集随机选取导致保守性.构造简单的 Lyapunov 函数,应用稳定性理论和微分不等式,经过严格的数学理论分析得到了使该网络达到渐近指数同步的充分条件,同时也证明了受牵制节点集不可能在有限时间内进行无限次快速切换.

关键词

同步;复杂网络;事件激发控制;非周期间歇牵制控制;Lyapunov 函数

中图分类号 O429

文献标志码 A

收稿日期 2019-06-05

资助项目 国家自然科学基金(61873171,61603260)

作者简介

丰建文(通信作者),男,博士,教授,博士生导师,研究方向为动力系统在复杂网络中的应用.fengjw@szu.edu.cn

1 深圳大学 数学与统计学院,深圳,518000

0 引言

近年来由于复杂网络的同步应用到多个不同领域中,如保密通信、图像处理、模式识别等[1-3],引起了研究者们的极大关注.多种同步方式如完全同步[4-5]、簇同步[6]、有限时间同步[7]、滞后同步[8]等得到了深入研究.1996 年,Milanović等[4]介绍了复杂网络的同步问题以及同步在通信上的应用;2006 年,Li 等[5]研究了带有时滞的复杂网络的同步问题.

一般说来,网络不可能通过自身达到同步,对网络施加控制以使网络达到同步是必不可少的.目前为止各种各样的控制策略已经被成功地应用于复杂网络的同步研究中,按控制时间主要可分为两大类:连续控制策略和非连续控制策略.相比于连续控制策略,非连续控制策略更能有效地降低控制成本和减少信息传输的数量.脉冲控制、数据采样控制、事件激发控制、间歇控制等是常见的非连续控制策略.另一方面由于网络中节点的数量庞大,对所有的节点均施加控制显然是不现实的,研究者们提出了只对网络中的一小部分节点施加控制来使全局网络达到同步的控制方法,即牵制控制[9-10].如 Chen 等[9]通过设计一个单一牵制控制器,实现了复杂网络的全局同步.为了更好地节约能量提高控制效益,人们将牵制控制与间歇控制进行有机结合,被称为间歇牵制控制,而周期间歇牵制控制方法就是其中的一种[11-12].如文献[11]研究了周期间歇控制下簇同步问题,并得到了网络达到簇同步的充分条件.然而周期性限制了这种控制策略的范围,现实世界中大部分现象是非周期的.Liu[13]将周期间歇控制策略推广到了非周期间歇控制策略,将每一小段时间区间分为工作区间和休息区间,给出了任意小段工作区间的下确界以及任意小段整个时间区间的上确界,并只对工作区间上的一小部分节点施加控制,这种控制策略的优势更加明显.

然而,现有的大多数关于牵制控制的文献中,牵制节点集总是固定的,这同样有可能会带来资源的浪费.为了克服牵制固定节点这一缺陷,Lu 等[14]提出了一种新的脉冲牵制策略,其中牵制控制节点集由一个具有较大的同步误差范数的节点组成.据笔者所知,迄今为止,在文献[14]中提出的牵制控制方案仅限于脉冲控制研究,而对于非

周期间歇时间序列下这种牵制策略的工作尚未见报道.现有研究中非周期间歇时间序列是事先给定的,这在实际控制中会造成资源浪费,为此,受到事件激发控制思想启发,本文引入了一个事件激发函数,并根据每一次事件的发生来更新牵制节点集,从而实现了牵制节点集的实时更新.文献[15]研究了在时间尺度上基于事件的时间间隔牵制控制下复杂网络的同步问题,遗憾的是其时间间隔是周期性的.此外,将脉冲牵制控制扩展到非周期间歇牵制控制的主要挑战在于如何避免牵制节点集的无限次快速切换.

综上所述,本文将事件激发控制和非周期间歇牵制控制方法进行有机结合,讨论了一般线性复杂网络的全局同步问题.首先设计一个事件激发条件和受牵制节点集选取规则,对受牵制节点集在每一个激发时刻进行更新,不仅极大程度上提高了网络同步的效率,并且克服了现有研究中受控节点集随机选取导致资源浪费这一缺陷.此外,本文还证明了受牵制节点集不可能在有限时间内进行无限次快速切换.本文所得结果是对现有相关结论的一个重要推广.

1 网络数学模型介绍

考虑以下一般线性复杂网络模型:

$$\dot{\boldsymbol{x}}_i(t)=\boldsymbol{A}\boldsymbol{x}_i(t)+c\sum_{j=1}^{N}l_{ij}\boldsymbol{x}_j(t)+\boldsymbol{u}_i(t),\quad i=1,2,\cdots,N. \tag{1}$$

其中,$\boldsymbol{x}_i(t)=(x_{i1}(t),x_{i2}(t),\cdots,x_{in}(t))^{\mathrm{T}}\in\mathbf{R}^n$ 是第 i 个节点的状态向量,$\boldsymbol{A}$ 是一个 $n\times n$ 矩阵,c 代表网络的耦合强度,$\boldsymbol{L}=(l_{ij})_{N\times N}$ 表示网络的拓扑结构,其定义如下:若节点 i 和节点 j 之间存在连接($i\neq j$),则 $l_{ij}=l_{ji}=1$,否则,$l_{ij}=l_{ji}=0$,且对角元素 l_{ii} 的定义为 $l_{ii}=-\sum_{j=1,j\neq i}^{N}l_{ij}$,$\boldsymbol{u}_i(t)$ 表示反馈控制器.

网络(1)中的每个孤立节点满足:

$$\dot{\boldsymbol{s}}(t)=\boldsymbol{A}\boldsymbol{s}(t). \tag{2}$$

本文致力于设计一个非周期间歇牵制控制策略 $\boldsymbol{u}_i(t)$,使得网络(1)同步于目标状态(2),文中 $\boldsymbol{u}_i(t)$ 的设计为

$$\boldsymbol{u}_i(t)=\begin{cases}-\boldsymbol{K}\boldsymbol{e}_i(t), & t\in[t_n,s_n],\quad n=0,1,2,\cdots,\ i\in\mathcal{D}_t,\\ 0, & t\in[t_n,s_n],\quad n=0,1,2,\cdots,\ i\in\mathcal{D}_t,\\ 0, & t\in(s_n,t_{n+1}),n=0,1,2,\cdots,\ i\in\mathcal{I}_N.\end{cases} \tag{3}$$

其中:$\boldsymbol{K}\in\mathbf{R}^{n\times n}$ 表示反馈控制增益矩阵,$\boldsymbol{e}_i(t)=\boldsymbol{x}_i(t)-\boldsymbol{s}(t)$ 表示第 i 个节点的误差状态向量.对于时间区间 $[t_n,t_{n+1})$,$[t_n,s_n]$ 表示控制区间,(s_n,t_{n+1}) 表示休息区间.集合 $\mathcal{D}_t$ 表示受牵制节点集,若时刻 t_s 表示 $\mathcal{D}_t$ 的开始时间,则 $\mathcal{D}_{t_s}$ 的定义如下:如果在时刻 t_s 时受牵制节点数量为 $\mathcal{D}_{t_s}=\ell_{t_s}$,对误差向量 $\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_N$ 进行重新排序,有 $\|\boldsymbol{e}_{p_1}\|\geqslant\|\boldsymbol{e}_{p_2}\|\geqslant\cdots\geqslant\|\boldsymbol{e}_{p_{\ell_{t_s}}}\|\geqslant\|\boldsymbol{e}_{p_{\ell_{t_s}+1}}\|\geqslant\cdots\geqslant\|\boldsymbol{e}_{p_N}\|$,那么,$\mathcal{D}_{t_s}=\{p_1,p_2,\cdots,p_{\ell_{t_s}}\}$.显然,$\ell_{t_s}/N\in[P_1,P_2]$,其中 P_1 和 P_2 分别表示受牵制节点的最小和最大比例,$\mathcal{I}_N$ 表示全局网络节点集.

对于每一个受牵制节点集 $\mathcal{D}_{t_k}$,决定下一个牵制节点集合的事件激发时刻定义为

$$t_{k+1}=\min_{i,j}\inf\{t>t_k:f(\boldsymbol{e}_i,\boldsymbol{e}_j)\leqslant 0\}, \tag{4}$$

其中

$$f(\boldsymbol{e}_i,\boldsymbol{e}_j)=\|\boldsymbol{e}_i(t)\|^2-\|\boldsymbol{e}_j(t)\|^2+\frac{\varepsilon_0}{N}\sum_{k=1}^{N}\|\boldsymbol{e}_k(t)\|^2,$$

$\forall i\in\mathcal{D}_{t_k},\forall j\notin\mathcal{D}_{t_k}.\varepsilon_0>0$ 表示一个事先给出的控制精度.

注 1 现有相关文献中,受牵制节点集是固定的,受文献[14]和事件激发控制思想的启发,在此引入一个事件激发条件确定下一个控制区间起始时刻,并且在该时刻对误差向量的范数进行从大到小排序,选取误差范数较大的节点作为受牵制节点集,即受控节点集在每个激发时刻间实时更新.

2 受牵制节点实时更新下复杂网络的同步分析

首先给出一些与本文密切相关的定义、假设和引理.

假设 1 对于非周期间歇控制策略,存在两个正数 $0<\theta<\omega$,使

$$\begin{cases}\inf_n(s_n-t_n)=\theta>0,\\ \sup_n(t_{n+1}-t_n)=\omega<+\infty\end{cases}$$

对任意的 $n=0,1,2,\cdots$ 均成立.

定义 1[16] 复杂网络(1)达到完全同步,如果有:

$$\lim_{t\to+\infty}\|\boldsymbol{x}_i(t)-\boldsymbol{s}(t)\|=0,\quad i=1,2,\cdots,N,$$

其中目标状态 $\boldsymbol{s}(t)$ 由式(2)所定义,$\|\cdot\|$ 表示向量或矩阵欧几里得范数.

定义 2[17] 如果矩阵 $\boldsymbol{L}=(l_{ij})_{i,j=1}^{N}$ 满足以下条件:

1) $l_{ij} \geqslant 0, i \neq j, l_{ii} = -\sum_{j=1, j\neq i}^{N} l_{ij}, i = 1,2,\cdots,N$;

2) $\boldsymbol{L}$ 是不可约的.

则称矩阵 $\boldsymbol{L}$ 属于 A_1 类,记为 $\boldsymbol{L} \in A_1$.

定义 3[18] 对于非周期间歇控制策略,定义

$$\psi = \lim_{i\to\infty} \sup \frac{t_{i+1} - s_i}{t_{i+1} - t_i}.$$

ψ 表示在时间区间 $t_{i+1} - t_i$ 中休息长度 $t_{i+1} - s_i$ 的最大比例.显然,可以得到 $0 \leqslant \psi < 1$.当 $\psi = 0$ 时,非周期间歇控制就变成了连续控制.不失一般性,总是假设 $\psi \in (0,1)$.此外,容易得到 $\psi \leqslant 1 - \frac{\theta}{\omega}$.

引理 1[17] 若矩阵 $\boldsymbol{L} \in A_1$,则有:

1) 矩阵 $\boldsymbol{L}$ 的秩为 $\mathrm{rank}(\boldsymbol{L}) = N - 1$.零是矩阵 $\boldsymbol{L}$ 的一个单重特征根,且 $\boldsymbol{L}$ 的所有非零特征值都有负实部;

2) 矩阵 $\boldsymbol{L}$ 特征值为零相对应的特征向量为 $\boldsymbol{1} = (1,1,\cdots,1)^{\mathrm{T}}$,也就是说,$\boldsymbol{L}\cdot\boldsymbol{1} = 0$;

3) 定义 $\boldsymbol{\xi} = (\xi_1,\xi_2,\cdots,\xi_N)^{\mathrm{T}}$ 是矩阵 $\boldsymbol{L}$ 特征值零所对应的左特征向量,即 $\boldsymbol{\xi}^{\mathrm{T}}\boldsymbol{L} = 0$,那么,总是有 $\xi_i > 0, i = 1,2,\cdots,N$ 以及 $\sum_{i=1}^{N} \xi_i = 1$.

引理 2[18] 若存在常数 $\psi \in (0,1)$,连续函数 $y(\cdot):[t_0, +\infty) \to [0, +\infty)$ 满足:

$$\begin{cases} \dot{y}(t) < -a_1 y(t), & t_i \leqslant t \leqslant s_i, \\ \dot{y}(t) < a_2 y(t), & s_i \leqslant t \leqslant t_{i+1}, \end{cases}$$

其中,a_1, a_2 都是常数,以及 $i = 0,1,2,\cdots$,如果 $a_1 > 0, \rho = a_1(1-\psi) - a_2\psi > 0$,那么,就可以得到

$$V(t) \leqslant V(t_0)\exp\{-\rho t\}, \quad t \geqslant 0.$$

接下来将给出本文的主要结果.

定理 1 若假设 1 成立,并且存在一个正常数 $\psi \in (0,1)$,使下列条件满足:

$$a_1 > 0, \quad \rho = a_1(1-\psi) - a_2\psi > 0,$$

那么,网络(1) 就能够渐近指数同步于目标状态 $\boldsymbol{s}(t)$.其中,ψ 由定义 3 中给出,

$$-a_1 = \max_i \left\{ \lambda_{\max}(\boldsymbol{A}^s) + 2c\lambda_i - 2\lambda_{\min}(\boldsymbol{K}^s)\left(P_1 - \frac{\varepsilon_0}{4N}\right) \right\},$$

$$a_2 = \max_i \{ \lambda_{\max}(\boldsymbol{A}^s) + 2c\lambda_i \}.$$

证明 记 $\boldsymbol{e}(t) = (\boldsymbol{e}_1^{\mathrm{T}}(t), \boldsymbol{e}_2^{\mathrm{T}}(t), \cdots, \boldsymbol{e}_N^{\mathrm{T}}(t))^{\mathrm{T}} \in \mathbf{R}^{nN}$,误差系统可表示为

$$\begin{cases} \dot{\boldsymbol{e}}(t) = (\boldsymbol{I}_N \otimes \boldsymbol{A})\boldsymbol{e}(t) + c(\boldsymbol{L} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t) - \\ \qquad (\boldsymbol{\Xi}_t \otimes \boldsymbol{K})\boldsymbol{e}(t), \quad t \in [t_n, s_n], \\ \dot{\boldsymbol{e}}(t) = (\boldsymbol{I}_N \otimes \boldsymbol{A})\boldsymbol{e}(t) + c(\boldsymbol{L} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t), \\ \qquad t \in (s_n, t_{n+1}). \end{cases} \tag{5}$$

其中 $\boldsymbol{\Xi}_t = \mathrm{diag}\{\xi_1,\xi_2,\cdots,\xi_N\}$,如果 $k \in \mathcal{D}_{t_s}$,那么,$\xi_k = 1$,否则,$\xi_k = 0$.

因为 $\boldsymbol{L}$ 是对称矩阵,那么,存在一个正交矩阵 $\boldsymbol{U} \in \mathbf{R}^{N\times N}$,使得 $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{J}\boldsymbol{U}^{\mathrm{T}}$,其中 $\boldsymbol{J} = \mathrm{diag}\{\lambda_1,\lambda_2,\cdots,\lambda_N\}$,$\boldsymbol{U} = (\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_N)$,$\boldsymbol{u}_i \in \mathbf{R}^N, i \in \mathcal{I}_N$.

记 $\boldsymbol{\delta}_i(t) = (\boldsymbol{u}_i^{\mathrm{T}} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t)$ 和 $\boldsymbol{\delta}(t) = (\boldsymbol{\delta}_1^{\mathrm{T}}(t), \boldsymbol{\delta}_2^{\mathrm{T}}(t), \cdots, \boldsymbol{\delta}_N^{\mathrm{T}}(t))^{\mathrm{T}}$,则有 $\boldsymbol{\delta}(t) = (\boldsymbol{U}^{\mathrm{T}} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t)$ 和 $\boldsymbol{e}(t) = (\boldsymbol{U} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t)$.同时有

$$\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t) = [(\boldsymbol{u}_i^{\mathrm{T}} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t)]^{\mathrm{T}}(\boldsymbol{u}_i^{\mathrm{T}} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t) = \boldsymbol{e}(t)^{\mathrm{T}}(\boldsymbol{u}_i\boldsymbol{u}_i^{\mathrm{T}} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t) = \boldsymbol{e}(t)^{\mathrm{T}}(\boldsymbol{I}_{ii} \otimes \boldsymbol{I}_n)\boldsymbol{e}(t),$$

且

$$\boldsymbol{e}^{\mathrm{T}}(t)\boldsymbol{e}(t) = \boldsymbol{\delta}^{\mathrm{T}}(t)\boldsymbol{\delta}(t)\boldsymbol{e}_i^{\mathrm{T}}(t)\boldsymbol{e}_i(t) = \boldsymbol{\delta}_i^{\mathrm{T}}(t)^{\mathrm{T}}\boldsymbol{\delta}_i(t).$$

当 $t \in [t_n, s_n]$ 时,有

$$\dot{\boldsymbol{e}}(t) = (\boldsymbol{U} \otimes \boldsymbol{I}_n)\dot{\boldsymbol{\delta}}(t) =$$
$$(\boldsymbol{I}_N \otimes \boldsymbol{A})(\boldsymbol{U} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t) + c(\boldsymbol{L} \otimes \boldsymbol{I}_n)(\boldsymbol{U} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t) - (\boldsymbol{\Xi}_t \otimes \boldsymbol{K})(\boldsymbol{U} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t) =$$
$$(\boldsymbol{U} \otimes \boldsymbol{A})\boldsymbol{\delta}(t) + c(\boldsymbol{U}\boldsymbol{J} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t) - (\boldsymbol{\Xi}_t\boldsymbol{U} \otimes \boldsymbol{K})\boldsymbol{\delta}(t),$$

所以,

$$\dot{\boldsymbol{\delta}}(t) = (\boldsymbol{U}^{\mathrm{T}} \otimes \boldsymbol{I}_n)(\boldsymbol{U} \otimes \boldsymbol{A})\boldsymbol{\delta}(t) + c(\boldsymbol{U}^{\mathrm{T}} \otimes \boldsymbol{I}_n)(\boldsymbol{U}\boldsymbol{J} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t) - (\boldsymbol{U}^{\mathrm{T}} \otimes \boldsymbol{I}_n)(\boldsymbol{\Xi}_t\boldsymbol{U} \otimes \boldsymbol{K})\boldsymbol{\delta}(t) =$$
$$(\boldsymbol{I}_N \otimes \boldsymbol{A})\boldsymbol{\delta}(t) + c(\boldsymbol{J} \otimes \boldsymbol{I}_n)\boldsymbol{\delta}(t) - (\boldsymbol{\Xi}_t \otimes \boldsymbol{K})\boldsymbol{\delta}(t).$$

这意味着

$$\begin{cases} \dot{\boldsymbol{\delta}}_i(t) = (\boldsymbol{A} + c\lambda_i\boldsymbol{I}_n - \boldsymbol{K})\boldsymbol{\delta}_i(t), & i \in \mathcal{D}_t, \\ \dot{\boldsymbol{\delta}}_i(t) = (\boldsymbol{A} + c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t), & i \notin \mathcal{D}_t. \end{cases}$$

类似地,当 $t \in (s_n, t_{n+1})$ 时,有 $\dot{\boldsymbol{\delta}}_i(t) = (\boldsymbol{A} + c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t), i \in \mathcal{I}_N$.

构造如下的 Lyapunov 函数:

$$V(t) = \sum_{i=1}^{N} \boldsymbol{e}_i(t)^{\mathrm{T}}\boldsymbol{e}_i(t) = \sum_{i=1}^{N} \boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t).$$

当 $t \in [t_n, s_n]$ 时,记 $V_1(t) = \sum_{i\in\mathcal{D}_t} \boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)$,$V_2(t) = \sum_{i\notin D_t} \boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)$,于是 $V(t) = V_1(t) + V_2(t)$.沿着轨迹(5) 的方向对 $V(t)$ 求导,可得:

$$\dot{V}_1(t) = 2\sum_{i\in\mathcal{D}_t} \boldsymbol{\delta}_i^{\mathrm{T}}(t)\dot{\boldsymbol{\delta}}_i(t) =$$
$$2\sum_{i\in\mathcal{D}_t} \boldsymbol{\delta}_i^{\mathrm{T}}(t)(\boldsymbol{A} + c\lambda_i\boldsymbol{I}_n - \boldsymbol{K})\boldsymbol{\delta}_i(t),$$
$$\dot{V}_2(t) = 2\sum_{i\notin\mathcal{D}_t} \boldsymbol{\delta}_i^{\mathrm{T}}(t)\dot{\boldsymbol{\delta}}_i(t) =$$
$$2\sum_{i\in\mathcal{D}_t} \boldsymbol{\delta}_i^{\mathrm{T}}(t)(\boldsymbol{A} + c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t),$$

因此,

$$\dot{V}(t)=2\sum_{i=1}^{N}\boldsymbol{\delta}_i^{\mathrm{T}}(t)(\boldsymbol{A}+c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t)-2\sum_{i\in D_t}\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{K}\boldsymbol{\delta}_i(t)\leqslant\sum_{i=1}^{N}\boldsymbol{\delta}_i^{\mathrm{T}}(t)(\boldsymbol{A}+\boldsymbol{A}^{\mathrm{T}}+2c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t)-2\sum_{i\in D_t}\lambda_{\min}(\boldsymbol{K}^s)\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t).$$

注意到到激发函数(4),

$$\sum_{i\in\mathcal{D}_t}\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)=\frac{\ell_t}{N}\sum_{i\in\mathcal{D}_t}\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)+\frac{N-\ell_t}{N}\sum_{i\in\mathcal{D}_t}\boldsymbol{\delta}_i(t)^{\mathrm{T}}\boldsymbol{\delta}_i(t)\geqslant\frac{\ell_t}{N}\sum_{i\in\mathcal{D}_t}\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)+\frac{\ell_t}{N}\sum_{i\notin D_t}\left(\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)-\frac{\varepsilon_0V(t)}{N}\right)\geqslant\frac{\ell_t}{N}\sum_{i\in\mathcal{D}_t}\boldsymbol{\delta}_i^{\mathrm{T}}(t)\boldsymbol{\delta}_i(t)-\frac{\ell_t}{N}\cdot\frac{N-\ell_t}{N}\cdot\frac{\varepsilon_0}{N}V(t)\geqslant\left(P_1-\frac{\varepsilon_0}{4N}\right)V(t),$$

可得

$$\dot{V}(t)\leqslant\sum_{i=1}^{N}\boldsymbol{\delta}_i^{\mathrm{T}}(t)(\boldsymbol{A}+\boldsymbol{A}^{\mathrm{T}}+2c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t)-2\lambda_{\min}(\boldsymbol{K}^s)V(t)\leqslant\max_i\left\{\lambda_{\max}(\boldsymbol{A}^s)+2c\lambda_i-2\lambda_{\min}(\boldsymbol{K}^s)\left(P_1-\frac{\varepsilon_0}{4N}\right)\right\}V(t)=-a_1V(t).$$

类似地,当 $t\in(s_n,t_{n+1})$ 时,有

$$\dot{V}(t)=2\sum_{i=1}^{N}\boldsymbol{\delta}_i^{\mathrm{T}}(t)(\boldsymbol{A}+c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_i(t)\leqslant\max_i\{\lambda_{\max}(\boldsymbol{A}^s)+2c\lambda_i\}V(t)=a_2V(t).$$

根据引理 2 可得:

$$V(t)\leqslant V(0)\exp\{-\rho t\},\quad t\geqslant 0,$$

其中,$\rho=a_1(1-\psi)-a_2\psi>0$.综上所述,网络(1)指数渐近同步于 $s(t)$.

最后,将要证明在激发函数(4)下,避免了受牵制节点集在有限时间内无限次快速切换.

定义 t_1 是受牵制节点集 $\mathcal{D}_{t_1}$ 的开始时刻,t_3 是受牵制节点集 $\mathcal{D}_{t_1}$ 的结束时刻.显然,对于任意的 $t\in[t_1,t_3]$,有 $\dot{\boldsymbol{\delta}}_i(t)=(\boldsymbol{A}+c\lambda_i\boldsymbol{I}_n-\boldsymbol{K})\boldsymbol{\delta}_i(t)$,$i\in\mathcal{D}_{t_1}$,及 $\dot{\boldsymbol{\delta}}_j(t)=(\boldsymbol{A}+c\lambda_i\boldsymbol{I}_n)\boldsymbol{\delta}_j(t)$,$j\notin\mathcal{D}_{t_1}$.

定义 $F_{ij}(t)=\dfrac{\|\boldsymbol{\delta}_i(t)\|^2-\|\boldsymbol{\delta}_j(t)\|^2}{V(t)}$,根据 $\mathcal{D}_{t_1}$ 的定义,可得 $F_{ij}(t_1)\geqslant 0$.

根据 $\|\boldsymbol{\delta}_p(t)\|(p\in\mathcal{I}_N)$ 的连续性以及事件激发函数(4),随着牵制控制的进行,存在 $t_2\in(t_1,t_3)$,使 $F_{ij}(t_2)=0$ 成立.那么,当 $t\in[t_2,t_3]$ 时,有 $F_{ij}(t_1)\geqslant\dfrac{\|\boldsymbol{\delta}_i(t)\|^2-\|\boldsymbol{\delta}_j(t)\|^2}{\|\boldsymbol{\delta}_j(t)\|^2}\triangleq H_{ij}(t)$,计算得:

$$\dot{H}_{ij}(t)\geqslant\frac{\|\boldsymbol{\delta}_i(t)\|^2}{\|\boldsymbol{\delta}_j(t)\|^2}[\lambda_{\max}(\boldsymbol{M}_i)-\lambda_{\min}(\boldsymbol{N}_j)]\geqslant\lambda_{\max}(\boldsymbol{M}_i)-\lambda_{\min}(\boldsymbol{N}_j),$$

其中 $\boldsymbol{M}_i=(\boldsymbol{A}+c\lambda_i\boldsymbol{I}_n-\boldsymbol{K})^s$,$\boldsymbol{N}_j=(\boldsymbol{A}+c\lambda_j\boldsymbol{I}_n)^s$.

令 $G_{ij}(t)=[\lambda_{\max}(\boldsymbol{M}_i)-\lambda_{\min}(\boldsymbol{N}_j)](t-t_2)$,$t\geqslant t_2$,然后,由比较原理[19],对于任意的 $t\geqslant t_2$,均有 $G_{ij}(t)\leqslant H_{ij}(t)\leqslant F_{ij}(t)$ 成立.

情况 1.当 $\lambda_{\max}(\boldsymbol{M}_i)\geqslant\lambda_{\min}(\boldsymbol{N}_j)$ 时,显然,$\forall t\geqslant t_2$,有 $G_{ij}(t)\geqslant 0$,这意味着 $t_3\to+\infty$.

情况 2.当 $\lambda_{\max}(\boldsymbol{M}_i)\geqslant\lambda_{\min}(\boldsymbol{N}_j)$ 时,令 $G_{ij}(t)=-\dfrac{\varepsilon_0}{N}$,则 $t=t_2-\dfrac{\varepsilon_0}{N[\lambda_{\max}(\boldsymbol{M}_i)-\lambda_{\min}(\boldsymbol{N}_j)]}\triangleq t_2+T_{ij}>t_2$.

因此,要使 $F_{ij}(t)=-\dfrac{\varepsilon_0}{N}$,$t$ 必须大于 t_2+T_{ij},也就计算出 $\mathcal{D}_{t_1}$ 的维持时间一定不小于

$$\begin{cases}\min\limits_{i,j}\{T_{ij}\}, & \text{当 }\lambda_{\max}(\boldsymbol{M}_i)<\lambda_{\min}(\boldsymbol{N}_j)\text{ 时},\\ +\infty, & \text{当 }\lambda_{\max}(\boldsymbol{M}_i)\geqslant\lambda_{\min}(\boldsymbol{N}_j)\text{ 时}.\end{cases}$$

这说明避免了受牵制节点集在有限时间内的无限次快速切换,证毕.

注 2 与现有的相关工作不同,此处受牵制节点集在每一个激发时刻都进行更新,并且还排除了出现在有限时间内牵制节点变换无限多次.需要强调的是,对于需要牵制多少个节点以及如何选择受牵制节点集的问题,本文参考了文献[15]中的算法 1,该算法对于降低全局网络的同步误差起到了关键作用.

3 数值模拟

本节将给出一个数值例子来验证所得结果的正确性.

考虑如下一个带有 5 个节点的二维线性复杂网络:

$$\dot{\boldsymbol{x}}_i(t)=\boldsymbol{A}\boldsymbol{x}_i(t)+c\sum_{j=1}^{5}l_{ij}\boldsymbol{x}_j(t),\quad i=1,2,\cdots,5,\quad(6)$$

其中,矩阵 $\boldsymbol{A}=\begin{pmatrix}0.05 & 0\\ 0 & 0.05\end{pmatrix}$,耦合强度 $c=0.01$,网络的耦合矩阵为

$$\boldsymbol{L}=\begin{pmatrix}-3 & 1 & 1 & 0 & 1\\ 1 & -2 & 0 & 0 & 1\\ 1 & 0 & -2 & 1 & 0\\ 0 & 0 & 1 & -1 & 0\\ 1 & 1 & 0 & 0 & -2\end{pmatrix}.$$

计算可得矩阵 $\boldsymbol{L}$ 的特征值为 $\lambda_1=0,\lambda_2=-0.518\,8,\lambda_3=-2.311\,1,\lambda_4=-3$ 以及 $\lambda_5=-4.170\,1$.除此之外,受牵制节点数的最小和最大比例分别为 $P_1=0.2$ 和 $P_2=0.6$,取参数 $\varepsilon_0=0.01$.另外,考虑非周期间歇控制策略,通过定义每一个工作区间和休息区间,可得假设 1 中参数 $\theta=3,\omega=4$,由此,参数 $\psi=0.25$.

为了满足定理 1 中的条件,利用数学软件 Matlab 中的 LMI 工具包,模拟可得反馈控制增益矩阵 $\boldsymbol{K}$ 需要满足条件 $\lambda_{\min}(\boldsymbol{K}^s)=0.167\,1$.选取反馈控制增益矩阵 $\boldsymbol{K}=\begin{pmatrix}0.18 & 0\\ 0 & 0.18\end{pmatrix}$,计算可得定理1中的参数 $a_1=0.021\,8,a_2=0.05$ 以及 $\rho=0.003\,9$.除此之外,参考文献[15]中算法 1 对受牵制节点集的选取规则,计算可得受控节点集在满足每一个事件激发时刻条件下的更新变化为 $\{5,4\}\rightarrow\{5\}\rightarrow\{3\}\rightarrow\{4\}\rightarrow\{5\}\rightarrow\{4\}\rightarrow\{3\}\rightarrow\{4\}\rightarrow\{5\}\rightarrow\cdots$.

图 1 描述了事件激发函数 $f(\cdot,\cdot)$ 随时间变化曲线,可以很直观地看到每一个事件激发确定时刻.图 2 描述了同步误差 $\boldsymbol{e}(t)$ 的运动轨迹,很明显地看到随着时间的进行,系统(6) 最终同步于目标状态 $\boldsymbol{s}(t)$.

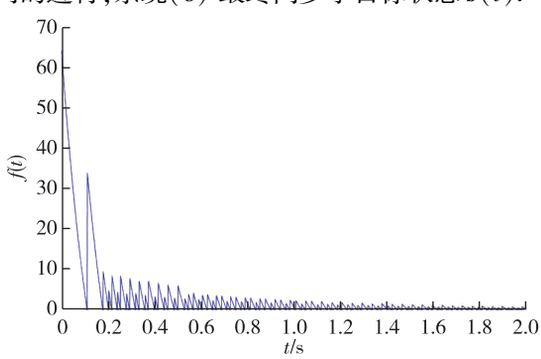


图 1 事件激发函数 $f(\cdot,\cdot)$ 随时间变化情况

Fig. 1 Time response of the event-triggered function $f(\cdot,\cdot)$

4 总结

本文将事件激发控制和非周期间歇牵制控制方法进行有机结合,研究了基于该策略下一般线性复杂网络的同步问题.需要指出的是,本文引入了一个事件激发函数和受牵制控制节点集选取法则,在每个激发时刻更新受牵制的节点集,这一改进显然更接近于现实情况,使网络更加经济有效地实现同步,极大地提高了网络同步效率,并且克服了现有文献中受牵制控制节点集随机选取导致的保守性.相比于现有的文献,又进一步将周期间歇控制策略推广到了非周期间歇控制策略.利用 Lyapunov 稳定性理论和微分不等式,分析得到了使网络达到渐近指数同步的充分条件,并通过数值模拟得到了使这些条件成立的参数.与此同时,通过严格的数学计算,证明了受牵制控制节点集不可能在有限时间内进行无限次快速切换.

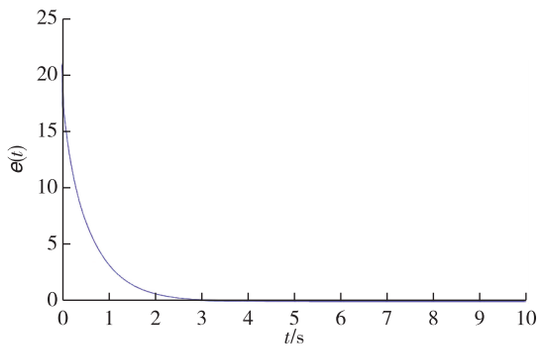


图 2 系统(6)和目标状态 $\boldsymbol{s}(t)$ 之间的误差 $\|\boldsymbol{e}(t)\|$ 随时间演变情况

Fig. 2 Time response of the error variables' modulus $\|\boldsymbol{e}(t)\|$ between system (6) and corresponding target states $\boldsymbol{s}(t)$

参考文献

References

[1] Pareek N K,Patidar V,Sud K K.Image encryption using chaotic logistic map[J].Image and Vision Computing,2006,24(9):926-934

[2] Kuhnert L,Agladze K I,Krinsky V I.Image processing using light-sensitive chemical waves[J].Nature,1989,337(6204):244-247

[3] Shawe-Taylor J,Cristianini N.Kernel methods for pattern analysis[M].Cambridge:Cambridge University Press,2003:3-24.DOI:10.1017/cbo9780511809682.002

[4] Milanović V,Zaghloul M E.Synchronization of chaotic neural networks and applications to communications[J].International Journal of Bifurcation and Chaos,1996,6(12b):2571-2585

[5] Li C P,Sun W G,Kueths J.Synchronization of complex dynamical networks with time delays[J].Physica A:Statistical Mechanics & Its Applications,2006,361:24-34

[6] Wang J Y,Feng J W,Xu C,et al.Cluster synchronization of nonlinearly-coupled complex networks with nonidentical nodes and asymmetrical coupling matrix[J].Nonlinear Dynamics,2012,67(2):1635-1646

[7] Feng J W,Li N,Zhao Y,et al.Finite-time synchronization

analysis for general complex dynamical networks with hybrid couplings and time-varying delays[J].Nonlinear Dynamics,2017,88(4):2723-2733
[8] Feng J W,Dai A D,Xu C,et al.Designing lag synchronization schemes for unified chaotic systems[J].Computers & Mathematics with Applications,2011,61(8):2123-2128
[9] Chen T P,Liu X W,Lu W L.Pinning complex networks by a single controller[J].IEEE Transactions on Circuits and Systems I(Regular Papers),2007,54(6):1317-1326
[10] Yu W W,Chen G R,Lü J.On pinning synchronization of complex dynamical networks[J].Automatica,2009,45(2):429-435
[11] Liu X W,Chen T P.Cluster synchronization in directed networks via intermittent pinning control[J].IEEE Transactions on Neural Networks,2011,22(7):1009-1020
[12] Mei J,Jiang M H,Wang X H,et al.Finite-time synchronization of drive-response systems via periodically intermittent adaptive control[J].Journal of the Franklin Institute,2014,351(5):2691-2710
[13] Liu X W.Synchronization of delayed complex-valued networks via aperiodically intermittent pinning control[C]// 2015 IEEE International Conference on Information and Automation,Lijiang,China,2015:1246-1251
[14] Lu J Q,Kurths J,Cao J D,et al.Synchronization control for nonlinear stochastic dynamical networks:pinning impulsive strategy[J].IEEE Transactions on Neural Networks and Learning Systems,2012,23(2):285-292
[15] Xiao Q,Lewis F L,Zeng Z G.Event-based time-interval pinning control for complex networks on time scales and applications[J].IEEE Transactions on Industrial Electronics,2018,65(11):8797-8808
[16] Li L L,Ho D W C,Cao J D,et al.Pinning cluster synchronization in an array of coupled neural networks under event-based mechanism[J].Neural Networks,2016,76:1-12
[17] Liu X W,Chen T P.Synchronization analysis for nonlinearly-coupled complex networks with an asymmetrical coupling matrix[J].Physica A:Statistical Mechanics andits Applications,2008,387(16/17):4429-4439
[18] Liu X W,Chen T P.Synchronization of linearly coupled networks with delays via aperiodically intermittent pinning control[J].IEEE Transactions on Neural Networks and Learning Systems,2015,26(10):2396-2407
[19] Bohner M,Peterson A.Dynamic equations on time scales:an introduction with applications[M].Boston,MA,USA:Birkh Wauser,2001

Synchronization analysis for general linear complex networks via event-based aperiodically intermittent pinning control

WANG Ruifeng[1] FENG Jianwen[1]
1 College of Mathematics and Statisticg,Shenzhen University,Shenzhen 518000

Abstract In this paper,the global synchronization problem of general linear complex networks is analyzed by combining the event-triggered control strategy with the aperiodically intermittent pinning control scheme.By designing an event-triggered condition and the rules for selecting the controlled nodes,the pinning node set can be updated under certain conditions,which not only greatly improves the efficiency of networks synchronization,but also avoids the deficiencies caused by random selection of controlled node set in existing research.Furthermore,a simple Lyapunov function is constructed,and the stability theory and differential inequality are applied,some sufficient conditions for the asymptotically exponential synchronization of the networks are obtained through rigorous mathematical analysis.At the same time,the infinitely fast switching of the pinning node set is also avoided.

Key words synchronization;complex networks;event-triggered control;aperiodically intermittent pinning control;Lyapunov function