



多智能体系统分布式事件触发控制综述

摘要

随着人工智能以及通信技术的发展,减少通信资源及计算资源的浪费,延长宽带和硬件的使用寿命成为了一个关键的问题.事件触发控制,即当某一变量超过给定阈值时,系统才进行采样,控制器才进行更新,有效地节约了线上通信及计算资源.本文首先分析了连续时间和离散时间下的事件触发控制,讨论了自触发控制、基于边的事件触发控制、动态事件触发控制以及异步事件触发控制的主要思想.其次,分别从一致性、优化以及博弈方面阐述了事件触发的广泛应用.同时,给出了两种常见的稳定性分析方法 and 两种排除系统中可能存在的 Zeno 行为的论证方法.最后对本文进行了总结,指出了当前事件触发工作的不足之处并对未来的研究做出展望.

关键词

多智能体系统;分布式控制;事件触发控制;Zeno 行为

中图分类号 TP18;TP13

文献标志码 A

收稿日期 2020-07-08

资助项目 国家自然科学基金(61903083);江苏省自然科学基金(BK20190333)

作者简介

张凯杰,女,硕士生,主要研究方向为博弈论.kjzhangmath@163.com

吕跃祖(通信作者),男,博士,讲师,主要研究方向为多智能体系统分布控制与估计.yzlv@seu.edu.cn

0 引言

随着计算机科学、人工智能和互联网的发展,多智能体系统(Multi-Agent Systems, MASs)受到众多研究者的关注,成为了控制学科和人工智能领域的研究热点^[1-2].MASs 是由多个具有感知、通信、计算和执行能力的个体(智能体)组成的系统,智能体之间通过网络拓扑与其他智能体进行信息交互,从而协调完成较为复杂的任务.在现实应用中,智能体的通信资源、计算能力等通常是有限的,且智能体之间往往不能进行实时通信,每个智能体的控制器也无法进行实时更新.为了解决上述问题,提高线上资源的利用率,传统的处理方式是采取时间触发,即触发时刻是事先确定好的,而不是根据系统的状态或时间进行动态确定的,甚至是周期控制^[3].这种方式虽然在一定程度上降低了通信及计算成本,但具有一定的局限性:一方面是当智能体的行为临近收敛状态时刻或者是在干扰较小的理想状态,如果继续使用时间触发采样控制会使得网络代价成本较高,造成不必要的资源浪费;另一方面是当时间触发中的采样周期较小,即采样频率较高时,网络中会存在大量的冗余信息,从而会导致网络堵塞,增加网络传感器的负担,甚至导致网络系统的崩塌.

事件触发是基于预先给定触发条件的一种控制方式,若控制任务满足触发条件,例如系统的状态误差超过某一设定阈值(通常与系统状态或时间有关),则事件发生,此时系统执行触发任务,进行邻居之间信息的传递或者控制器的更新.与时间触发控制相比,由于事件触发控制的采样时间取决于触发条件是否满足,因此它可以有效减少控制任务的执行次数,节约通信、计算资源,一定程度上解决了网络拥堵的问题.在分布式控制系统中,文献[4]更加清楚地描述了时间触发和事件触发的区别.在 MASs 分布式控制中,智能体之间进行信息交互会产生一定程度的消耗,所以采用连续时间控制会很大程度上造成通信资源的浪费.由于事件触发控制在节约通信、计算等资源方面的优势,其在多智能体系统分布式控制中已得到广大研究者的关注,并已经被广泛地应用到各个领域^[5-6].

1962年,事件触发机制率先由 Dorf 等^[7]提出,它是一种通过测量系统参数来改变采样频率的自适应系统.随后,事件触发的思想应用于发动机控制中,用来解决传感器问题^[8].早在 1999 年的 IFAC(国际自动控制联合会)会议上,有关事件触发的论文^[9-10]一经发表就引起

1 辽宁工程技术大学 理学院,阜新,123000

2 东南大学 数学学院,南京,211189

了科研工作者的广泛关注,文献[9-10]说明了与时间触发控制相比,事件触发控制具有的一些优势.2002年,CDC(IEEE决策和控制会议)会议上Aström等^[11]通过对比事件触发采样和周期采样的区别,得到了事件触发采样的效果更好的结论.随着越来越多研究者对事件触发控制研究的关注,而后的CDC会议上针对事件触发还设置了专题分会,这在一定程度上反映了事件触发控制在多智能体系统中的重要性.与此同时,在人工智能飞速发展的浪潮下,智能体之间的通信成为了关键的传递信息的方式,它是维持多智能体系统保持协同一致以及追踪的重要基础,在这样的背景之下,对于通信资源的整合和利用变得愈加重要.在保证系统稳定性的基础上,事件触发控制能够更好地节约通信资源,带来更大的经济效益,有着广阔的应用前景.

本文基于事件触发通信,介绍了不同状态下系统中智能体的通信情况,从时间方面分别分析了连续时间和离散时间下的触发机制;给出了基于边的事件触发、自触发控制、动态事件触发以及异步事件触发的主要内容;阐述了事件触发在MASs实现一致性、分布式优化以及博弈上的一些应用.

1 事件触发通信机制

本节主要从时间的角度介绍事件触发的不同通信方式,包括连续时间下的事件触发控制和离散时间下的事件触发控制.在多智能体系统中,当智能体状态满足事件触发条件时,它需要向其他智能体传递自身的状态使得控制策略更新,进而执行相对应的任务,但是如何与其他智能体进行通信,这是一个关键性的问题.

1.1 事件触发控制

1.1.1 连续时间通信

连续事件触发需要硬件连续对智能体的状态进行检测判断触发条件是否满足,连续时间状态下事件触发的运行机制如图1所示.在连续时间触发控制中,检测器会对智能体的状态进行实时的检测,当触发条件满足时才进行信息的传输.

图1说明对于一阶系统^[12]

$$\dot{x}_i = u_i(t), i = 1, 2, \dots, N, \quad (1)$$

一般定义状态误差 $e_i(t) = \hat{x}_i(t) - x_i(t)$, 其中 $\hat{x}_i(t) = x_i(t_k^i)$, $t \in [t_k^i, t_{k+1}^i)$, $x_i(t)$ 表示智能体 i 在 t 时刻的状态, $\hat{x}_i(t)$ 表示智能体 i 最近一次触发传递的信息, 如果两者之间的差值超过设定的阈值时, 即 $\|e\| \leq$

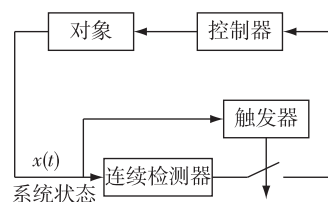


图1 连续事件触发控制

Fig. 1 Continuous event-triggered control

$\sigma \frac{\|Lx\|}{\|L\|}$, $\sigma \in (0, 1)$ 不成立时进行事件触发, 其中 $e(t) = [e_1(t), \dots, e_N(t)]^T$, L 是通信图的拉普拉斯矩阵. 它的控制协议如下:

$$u_i(t) = - \sum_{j \in N_i} [x_i(t_k^i) - x_j(t_{k'(t)}^j)], \quad (2)$$

其中 $k'(t) = \arg \min_{l \in N_i: t \geq t_l^i} \{t - t_l^i\}$. 对于二阶积分系统

$$\begin{cases} \dot{x}_i(t) = v_i(t), \\ \dot{v}_i(t) = u_i(t), \end{cases} i = 1, 2, \dots, N, \quad (3)$$

$x_i(t), v_i(t), u_i(t)$ 分别代表智能体 i 的位置、速度和控制输入. 为了便于分析, 文献[13]中设置了两种状态误差:

$$\begin{cases} e_{x,i}(t) = x_i(t_k^i) - x_i(t), \\ e_{v,i}(t) = v_i(t_k^i) - v_i(t), \end{cases} i = 1, 2, \dots, N, \quad (4)$$

它所对应的事件触发控制为

$$u_i(t) = k^2 \sum_{j \in N_i} a_{ij} [x_i(t_k^i) - x_j(t_{k'(t)}^j)] + k^3 \sum_{j \in N_i} a_{ij} [v_i(t_k^i) - v_j(t_{k'(t)}^j)]. \quad (5)$$

然后利用稳定性定理进行分析证明, 得到系统的状态是收敛的. 对于一般线性系统

$$\dot{x}_i(t) = Ax_i(t) + Bu_i(t), \quad (6)$$

文献[14]为了实现领导者和追随者的状态一致, 基于事件触发机制设计控制协议如下:

$$u_i(t) = K \sum_{j=1}^N a_{ij} [x_j(t_k^j) - x_i(t_k^i)] + K \hat{d}_i (s(t_k^i) - x_i(t_k^i)), \quad (7)$$

其中 K 和 \hat{d}_i 都是智能体 i 的增益, $s(t_k^i)$ 是领导者在触发时刻的状态. 非线性系统中事件触发的研究可在文献[15]看到, 它通过测量系统状态与最优预测之间的误差, 设计了事件触发模型预测算法, 通过该算法可得到智能体的状态收敛于一个不变集.

1.1.2 离散时间通信

与连续通信相对应的就是离散通信. 不同于连续时间下的事件触发控制, 离散时间下的事件触发是以周期进行采样的, 也可称为周期事件触发. 周期

事件触发最早是由 Heemels 等^[16]提出的,其触发原理流程如图 2 所示.

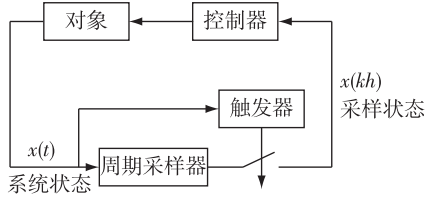


图 2 周期事件触发控制

Fig. 2 Periodic event-triggered control

与连续时间触发不同,图 2 周期事件触发控制是基于周期采样器,周期性地对事件触发条件进行检测,当系统触发时智能体将采样状态进行传输.周期事件触发是一种结合了周期采样和事件触发的控制机制,由于它是周期性地对事件触发条件进行检测,任意两次触发的时间之间一定存在一个正的下界,因此这种触发是不存在 Zeno 行为的.周期事件触发实质上是连续事件触发的离散化,在实际应用中连续的检测需要耗费一定的资源,因此周期事件触发在系统控制中的应用更加广泛.周期事件触发以时间间隔 h 进行采样,可由下面算法来判断是否进行传输:

$$[\mathbf{x}((k+j)h) - \mathbf{x}(kh)]^T \Phi [\mathbf{x}((k+j)h) - \mathbf{x}(kh)] \leq \eta \mathbf{x}^T((k+j)h) \Phi \mathbf{x}((k+j)h), \quad (8)$$

其中 Φ 是一个对称正定的矩阵, $\eta \in [0, 1)$, $j = 1, 2, \dots$.

需要注意的是,若采样状态 $\mathbf{x}(k+j)h$ 满足上述不等式(8)将不会被传输,只有超过上述不等式的阈值时,采样状态才会被发送到控制器.由于周期事件触发的一些优良特质,它被应用于很多方面.在一类由高维主系统和低维从系统构成的主从耦合系统中,文献[17]设计了一种事件触发的采样数据传输策略.在线性系统中,文献[18]研究了基于观测器的控制,并提出了一种新的事件触发机制,这种机制可以减少传感器到控制器通道和控制器到执行器通道的通信.对于线性定常系统,文献[19]提出了基于周期事件触发的滑模控制设计的方法,这种触发机制在智能体状态可测的那些瞬间进行周期性评估,不再需要连续判断事件触发策略,实现起来更具有经济价值.

1.2 基于边的通信

对于事件触发控制,可以根据事件触发条件设计的对象将其分为基于边的事件触发机制和基于点

的事件触发机制.基于边的事件触发建立在智能体之间的连边上,当事件触发条件满足时,连边上的智能体之间进行信息交互.而基于点的事件触发针对的是系统中的每个智能体,触发条件满足时智能体会与它所有的邻居进行信息交互,上述相关设计都是基于点的.

在无向连通图 $G = (V, E)$ 中, $\{i, j\} \in E$ 代表智能体 i 和智能体 j 之间的通信连边,用 $\tilde{x}_{ij}(k)$ 表示智能体 i 在最新一次触发发送到智能体 j 的状态,其表达式如下:

$$\tilde{x}_{ij}(k) = \begin{cases} x_i(k), \\ \tilde{x}_{ij}(k-1), \end{cases} \quad (9)$$

当 k 属于触发时间集合时,智能体 i 通过连边向智能体 j 传递它此时的状态,否则智能体 j 保持上一时刻智能体 i 对其传输的状态,事件触发条件如下:

$$\|x_i(k) - \tilde{x}_{ij}(k)\| \leq e_{ij}(k), \quad (10)$$

其中 $e_{ij}(k)$ 是可调节的阈值.阈值和事件触发的次数相关,阈值越小,触发次数越多,意味着智能体之间的通信越频繁.文献[20]研究基于边的事件触发机制下的网络系统的协调问题.对于一般线性 MASs 的编队控制问题,文献[21]提出了四种基于边的事件触发控制协议,且每种协议都不存在 Zeno 行为.

1.3 自触发控制

由于事件触发需要专门的硬件设施去实时监测系统状态,当系统的状态误差满足触发条件时,事件发生.正常的操作系统中,事件触发机制是可以满足的,但是在一些大规模或者比较极端的情况下,使用事件触发机制可能会造成一些资源的浪费.与事件触发相比,自触发是主动的一种通信方式,它可以提前根据上一触发时刻的数据计算得到下一个触发时刻,其原理如图 3 所示.

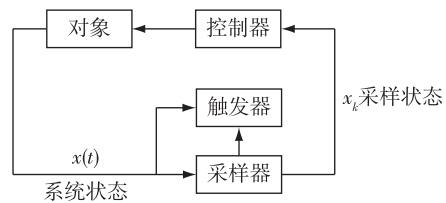


图 3 自触发控制

Fig. 3 Self-triggered control

图 3 自触发控制通过当前的采样值和采样时刻来决定下一时刻的触发,通过这种机制直接进行采样状态的传输.基于自触发机制,下一个触发时刻通过下述公式确定:

$$t_{k+1} = t_k + \tau_k, \quad (11)$$

其中 $\tau_k = \Gamma(x(t_k))$, Γ 是 $x(t)$ 在 t_k 时刻的函数. 自触发控制是 Velasco 等在实时系统中提出的一个概念^[22], 它和事件触发控制有着一定的区别, 主要体现在通信方式上, 具体详细阐述可参考文献[23-24]. 由于自触发控制只需要当前智能体的信息状态就能够确定下一个触发时刻, 因此它在分布式控制中可以得到很好的应用. 文献[25]分析了自触发控制在具有一致目标的 MASs 中的应用. 文献[26]抛弃了传统反馈控制律上的周期性的假设, 通过自触发机制利用被测对象的当前状态来决定下一个被测状态、计算控制律和更新执行器的时刻. 文献[27]基于自触发提出了一种协调算法, 分析了在同步和异步情况下算法的正确性.

1.4 动态事件触发

静态事件触发的触发条件中只有智能体的状态项和误差项, 为了更好地对 Zeno 行为进行分析, 同时减少触发次数, 通过引入动态变量 η , 文献[28]提出了如下动态事件触发机制:

$$t_{k+1}^i = \max_{r \geq t_k^i} \left\{ r: \theta_i \left(L_{ii} e_i^2(t) - \frac{\sigma_i}{2} q_i(t) \right) \leq \eta_i(t), \forall t \in [t_k^i, r] \right\}, \quad (12)$$

其中 $q_i(t) = -\frac{1}{2} \sum_{j=1}^n L_{ij} (x_j(t) - x_i(t))^2 \geq 0, \sigma_i \in [0, 1]$, $\eta_i(t)$ 是内部动态变量表达式如下:

$$\dot{\eta}_i(t) = -\beta_i \eta_i(t) + \xi_i \left(\frac{\sigma_i}{2} q_i(t) - L_{ii} e_i^2(t) \right), \quad (13)$$

其中 $\beta_i > 0, \xi_i \in [0, 1]$, 内部动态变量的初值 $\eta_i(0) > 0$. 需要注意的是, 触发条件中的 θ_i 是后期待定的参数, 当 θ_i 趋于无穷时, 式(12)就对应于静态事件触发条件, 静态事件触发实质上是动态事件触发的特例. 文献[29]通过引入动态变量证得闭环系统的稳定性. 文献[30]的动态事件触发控制策略同时保证了干扰到输出的有限 L_p 增益和事件触发时间的严格正的下界.

1.5 异步事件触发

事件触发根据每个智能体是否同时触发还可以分为同步事件触发和异步事件触发. 在控制系统中, 如果事件触发条件是针对所有的智能体, 当满足触发条件时, 所有的智能体都进行信息的传输和策略的更新, 这种触发控制称为同步事件触发, 对于每个智能体而言该触发通信是同步的. 但是在一些实际

情况中, 同步更新并不一定能够实现, 因为它需要用到系统中所有智能体的状态信息来判断触发条件是否满足, 而且可能造成不必要的资源浪费, 这时需要对智能体单独设计事件触发条件, 使得智能体的更新并不是同步的, 即每个智能体都有自己的事件触发时刻序列, 此为异步事件触发. 在具有完全状态反馈的非线性系统中, 文献[31]提出了一种分布式的异步事件触发方法, 利用局部信息来确定从传感器到中央控制器的传输时间, 所提出的方法不仅保证了系统的稳定性, 也使得采样的时间具有正下界, 排除了 Zeno 现象. 文献[32]针对一组不在同一位置的传感器, 设计了分布式事件触发控制器, 其中传感器不需要同步测量的采样方式. 在输出测量和控制输入受到噪声干扰的系统中, 文献[33]根据控制器和被控系统的动态输出分别设计了事件触发条件.

2 事件触发的应用

事件触发控制在保证了底层需要的同时, 减少了智能体之间的通信频率. 一些研究人员已经分析了它在各个领域的作用, 比如在解决滤波问题^[34]、网络电力系统^[35]以及在时延控制系统^[36]中. 下面简要分析事件触发在 MASs 一致性、分布式优化以及博弈方面的应用.

2.1 MASs 一致性问题

在 MASs 控制中, 一致性是实现协同控制的关键问题. 主要是设计一致性协议, 使智能体通过邻居之间的信息交互使其状态达到一致. 对于智能体的动态方程 $\dot{x}_i = u_i(t)$, 通常设计一致性算法 $u_i(t) = \sum_{j \in N_i} a_{ij} (x_j(t) - x_i(t))$. 对于二阶动态方程而言, 较为成熟的一致性控制输入算法为 $u_i(t) = -\sum_{j=1}^n a_{ij} [(x_i(t) - x_j(t)) + \gamma(v_i(t) - v_j(t))]$, 在上述连续事件触发通信介绍部分已经给出. 而在实际情况中, 在保证系统稳定性前提下, 应当尽量降低系统的通信成本. 由于事件驱动控制在节约线上资源方面的优势, 其已被广大学者应用到 MASs 一致性研究中. 文献[37]提出了一种新的分布式事件触发算法来解决多智能体平均一致性问题. 在一致性协议上引入事件触发通信不仅能够使智能体达到目标状态, 而且一定程度上减少了通信. 对于一类由 N 个智能体所组成的线性系统, 通过设置事件触发机制:

$$f_i(e_i(t)) \leq \Delta_i(z_i(t)), \quad (14)$$

其中 $e_i(t) = x_i(t_k^i) - x_i(t)$, $z_i(t) = \sum_{j \in N_i} w_{ij}(x_i(t) - x_j(t))$, Δ_i 是阈值函数, 若阈值和智能体的状态有关, 称为状态依赖阈值, 否则称为状态独立阈值.

当智能体的状态不满足事件触发条件时, 智能体与其邻居之间进行通信, 引发控制一致性协议更新, 最终在事件触发一致性控制下达到协同状态. 在一般线性动力学的多智能体系统的一致性问题上, 文献[38]利用事件触发控制和自适应控制技术, 提出了一种基于观测器输出反馈的分布式事件触发控制策略. 对于具有时变通信延迟的线性多智能体系统的事件触发一致性问题, 文献[39]提出了一种新的控制器去解决. 上述研究是针对线性系统下的分布式事件触发一致性控制, 其中单积分器和双积分器可以当作它的特殊情况. 文献[40]对多智能体系统的事件触发一致性控制进行了详细的综述. 上述多数研究内容中, 在确定系统相关参数时往往需要知道系统的全局信息, 如系统通信拓扑结构的拉普拉斯矩阵的特征值或其范数等, 从某种意义上来说并不是完全分布式的. 为了解决上述问题, 研究者就结合自适应控制, 提出了一种完全分布式的自适应事件触发控制协议, 如文献[41]中的基于事件触发的自适应协议是完全分布的和可扩展的, 它不依赖于网络图的任何全局信息, 解决了无领导和领导-追随者一致性问题. 文献[42]提出了一种完全分布式事件触发控制策略, 对于任意连通的无向通信图, 在完全分布的情况下都能解决一致性问题. 针对具有通信时延和无通信时延的单积分器和双积分器网络情况, 文献[43]提出的事件触发控制策略能保证智能体的行为平均一致收敛或者一致收敛到状态平均值的邻域范围内. 文献[44]在无向图上提出了基于事件触发控制策略的一致性协议, 该协议中的每个智能体都实现了与邻居的解耦.

2.2 分布式优化问题

优化与事件触发结合主要体现在求解目标函数的最优值, 研究较多的是凸优化问题. 文献[45]针对二次凸优化问题:

$$\begin{aligned} \min \quad & \sum_{i=1}^n f_i(x_i(t)) \\ \text{s.t.} \quad & \sum_{i=1}^n x_i(t) = X_D, \end{aligned} \quad (15)$$

其中 $f_i(x_i(t)) = a_i x_i^2(t) + b_i x_i(t) + c_i$, a_i, b_i, c_i 是二次函数 $f_i(x_i(t))$ 的系数, X_D 是一个常数用来约束加和

项 $\sum_{i=1}^n x_i(t)$.

为了解决式(15)的凸优化问题, 基于事件触发机制提出了分布式优化算法:

$$\begin{aligned} \dot{\xi}_i(t) &= -k_i \sum_{j \in N_i} a_{ij} (\xi_i(t_k^i) - \xi_j(t_k^j)), \\ x_i(t) &= \frac{\xi_i(t) - b_i}{2a_i}, \end{aligned} \quad (16)$$

k_i 是一个常数, 通过上述策略进行智能体状态的更新, 最终得到优化的目的. 该算法是在无向连通图的通信拓扑下进行更新的, 它也可以推广到有向图和切换拓扑中. 对于类似的分布式凸优化问题, 文献[46]提出了一种事件触发零梯度和的算法, 该算法的通信过程是由节点监控的触发条件所驱动的, 在连续时间下基于采样进行监控, 离散时间下则使用所提出的算法, 并且证明了此算法是指数收敛的. 文献[47]基于事件触发通信, 给出了一种步长恒定的分布式优化算法, 并且利用小增益定理证明了收敛性. 文献[48]通过异步事件触发方案, 解决了多个智能体最优化自身的目标函数的问题, 并且在忽略通信延迟的情况下可以保证系统的收敛性. 对于带有约束的优化问题, 文献[49]提出了一种基于事件触发通信的分布式次梯度方法, 在该方法中, 所有代理的状态在步长递减和可加性条件以及触发条件的阈值下渐近收敛于一个最优解. 在连续时间动力学系统中, 文献[50]通过对系统设置事件触发条件使得智能体的状态渐近收敛到全局的最优解. 在二阶多智能体的凸优化问题中, 文献[51]将事件触发和时间触发算法相结合, 使智能体协同收敛于优化问题的最优解. 同样在二阶连续时间系统中, 文献[52]通过一种基于事件触发的完全分布式优化算法, 证明了该系统的指数收敛性.

2.3 博弈问题

在一场博弈中, 每个智能体都趋于使得自己的利益最大化或者成本最小化, 将事件触发机制引入到博弈中也是一种很好的应用. 在 N 个智能体下的聚合博弈中^[53], 每个博弈者 i 的目标是解决如下的优化问题:

$$\begin{aligned} \min_{x_i} \quad & C_i(x_i, x_{-i}) \\ \text{s.t.} \quad & x_i \in X_i, \end{aligned} \quad (17)$$

其中 $C_i(x_i, x_{-i}) : X \rightarrow \mathbf{R}$ 表示智能体 i 的代价函数, 它不仅依赖自己的策略, 而且依赖于其他博弈者的策略, 且有 $C_i(x_i, x_{-i}) = J_i(x_i, \bar{x})$, 其中 $\bar{x} = \sum_{i=1}^N x_i$ 表示所

有博弈者决策的聚合。 χ_i 是智能体 i 的局部策略集合。针对上述优化问题, 设置了事件触发条件 $\| \hat{v}_j^{k-1} - v_i^k \| > E_{ij}^k$, 当满足触发条件时智能体 i 向智能体 j 发送 v_i^k 。对于非线性连续时间系统下的非零和博弈, 文献[54] 提出了一种事件触发自适应控制方案。

3 稳定性分析

稳定性一直是系统研究的重要问题。在控制系统中稳定性能够保证智能体的状态是收敛的, 这在实际生活中有着重要的意义。因此在事件触发控制中, 触发条件的设计必须保证系统的稳定性。

3.1 李雅普诺夫稳定性

在现代控制理论中, 一般采用李雅普诺夫稳定性 (Lyapunov stability) 去判断该系统的特性。在事件触发通信过程中, 依旧可以通过构造李雅普诺夫候选函数, 根据稳定性定理证得智能体的状态是收敛的。对于带有时滞的控制系统^[55], 需要构造 Lyapunov 泛函去证明系统的稳定性。

3.2 输入状态稳定性

可以利用输入状态稳定性 (input-to-state-stability) 去分析系统的性能。文献[56] 研究了具有有界扰动的事件触发控制系统, 给出了具有观测器输出反馈的输入状态稳定性的结果。

4 Zeno 行为分析

Zeno 行为是指在事件触发控制中, 在有限时间内发生无限次触发的现象, 这种情况在物理上是不可实现的, 在现实中也是不合理的。Zeno 行为存在与否与事件触发条件的设置有关, 通常事件触发条件的设置会保证两次触发的间隔有一个正的下界。对于周期事件触发而言, 由于其采样的数据是周期性的, 这种采样是离散的, 任意两次采样的时间间隔一定会大于一个正值, 所以已经避免了 Zeno 行为, 但是对于其他方式的触发, 需要分析 Zeno 行为是否存在。目前 Zeno 行为的分析没有特定的方法, 比较常见的方法如下。

4.1 正下界法

如果任意两次触发间隔都大于一个正的下界, 那么这样就可以排除 Zeno 行为。在多智能体系统的分布式事件触发控制中, 文献[57] 的引理 4 通过证明得到智能体 i 的触发间隔 $\tau_k^i = \beta_i \| q_i(t_k^i) \| / 2\zeta_i(1 + \beta_i) \sqrt{V(t_k^i)}$ 为正值, 从而说明不存在 Zeno 行

为。在反馈控制系统中, 对于文献[58] 所提的触发条件 $t_{k+1}^i = \max \{ \tau \geq t_k^i : |x_i(t_k^i) - x_i(t)| \leq \delta_i(t), \forall t \in [t_k^i, \tau] \}$, 通过利用智能体状态的有界性等条件得到正的下界 $1 / ((M/\varphi_i) e^{\alpha_i T}) + \alpha_i$ 。

4.2 反证法

反证法首先假设存在 Zeno 行为, 根据 Zeno 行为的实际意义可知必存在一个具有聚点的时间序列 $t_0^i, t_1^i, t_2^i, \dots, t_k^i, \dots$, 当触发次数趋于无穷时, 此时的时间点为一个固定的时间值。然后根据稳定性分析以及智能体的状态等条件得到矛盾, 从而说明不存在 Zeno 行为。文献[24] 中对于静态和动态事件触发 Zeno 行为的分析使用的都是反证法。

5 总结与展望

事件触发控制策略, 由于它自身具有很多优良的特点, 可以尽可能地实现资源的优化, 节省成本, 在现实生活中发挥着重要的作用。越来越多的科研工作者对事件触发控制进行研究。本文首先从时间方面分析了连续时间和离散时间系统下的事件触发机制, 又介绍了几种常见的事件触发控制, 例如自触发控制、基于边的事件触发控制、动态事件触发控制以及异步事件触发控制。然后分析了其在 MASs 一致性问题、分布式优化以及博弈问题中的相关研究进展。

虽然, 关于事件触发控制方面的研究已经取得了不少不错的成果, 但依然存在很多问题需要去解决。本文对于事件触发控制的进一步研究做出了如下展望:

1) 目前分析 Zeno 行为的方法较少, 没有一套成熟的、可普遍适用的理论框架。一般情况下还是使用任意两次触发之间间隔的正下界性和反证法, 对于其他的证明方法有待进一步探究。

2) 对于误差状态小于阈值的触发条件来说, 阈值的大小决定了事件触发的频率, 阈值越大, 触发次数越少, 可以更大程度地减少通信频率, 但系统的收敛速率就会相对较慢。所以如何更好地设计误差触发条件, 在保证收敛性能的同时使得通信频率也较少, 这一问题值得更深层次的研究。

3) 现阶段的事件触发分析是基于控制器已经设定好的情况下, 如何设计事件触发条件和控制器结合起来的控制机制需要更多的科研工作者进行研究。

4) 在离散时间下, 设计事件触发条件时如何使

用内部动态变量,这个问题有待解决.

参考文献

References

- [1] Ren W, Beard R W, Atkins E M. Information consensus in multivehicle cooperative control [J]. IEEE Control Systems Magazine, 2007, 27(2) : 71-82
- [2] Yu W W, Wen G H, Chen G R, et al. Distributed cooperative control of multi-agent systems [M]. Singapore: John Wiley & Sons Singapore Pte. Ltd, 2016
- [3] Yin X X, Yue D, Hu S L. Adaptive periodic event-triggered consensus for multi-agent systems subject to input saturation [J]. International Journal of Control, 2015, 89(4) : 1-21
- [4] Albert A. Comparison of event-triggered and time-triggered concepts with regard to distributed control systems [J]. Embedded World, 2004: 235-252
- [5] Yu D D, Xia Y Q, Li L, et al. Event-triggered distributed state estimation over wireless sensor networks [J]. Automatica, 2020, 118: 109039. DOI: 10.1016/j. automatica. 2020.109039
- [6] Yang D P, Ren W, Liu X D, et al. Decentralized event-triggered consensus for linear multi-agent systems under general directed graphs [J]. Automatica, 2016, 69: 242-249
- [7] Dorf R, Farren M, Phillips C. Adaptive sampling frequency for sampled-data control systems [J]. IRE Transactions on Automatic Control, 1962, 7(1) : 38-47
- [8] Hendricks E, Jensen M, Chevalier A, et al. Problems in event based engine control [C] // Proceedings of 1994 American Control Conference, 1994: 1585-1587
- [9] Arzen K. A simple event-based PID controller [J]. IFAC Proceedings Volumes, 1999, 32(2) : 8687-8692
- [10] Åström K J, Bernhardsson B M. Comparison of periodic and event based sampling for first-order stochastic systems [J]. IFAC Proceedings Volumes, 1999, 32(2) : 5006-5011
- [11] Åström K J, Bernhardsson B M. Comparison of Riemann and Lebesgue sampling for first order stochastic systems [C] // Proceedings of the 41st IEEE Conference on Decision and Control, 2002: 2011-2016
- [12] Dimarogonas D V, Frazzoli E, Johansson K H. Distributed event-triggered control for multi-agent systems [J]. IEEE Transactions on Automatic Control, 2012, 57(5) : 1291-1297
- [13] Yan H C, Shen Y C, Zhang H, et al. Decentralized event-triggered consensus control for second-order multi-agent systems [J]. Neurocomputing, 2014, 133: 18-24
- [14] Zhou B, Liao X F, Huang T W, et al. Leader-following exponential consensus of general linear multi-agent systems via event-triggered control with combinational measurements [J]. Applied Mathematics Letters, 2015, 40: 35-39
- [15] Li H P, Shi Y. Event-triggered robust model predictive control of continuous-time nonlinear systems [J]. Automatica, 2014, 50(5) : 1507-1513
- [16] Heemels W P M H, Sandee J H, van D B P P J. Analysis of event-driven controllers for linear systems [J]. International Journal of Control, 2008, 81(4) : 571-590
- [17] Wen G H, Chen M Z Q, Yu X H. Event-triggered master: slave synchronization with sampled-data communication [J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2016, 63(3) : 304-308
- [18] Heemels W P M H, Donkers M C F. Model-based periodic event-triggered control for linear systems [J]. Automatica, 2013, 49(3) : 698-711
- [19] Behera A K, Bandyopadhyay B, Yu X H. Periodic event-triggered sliding mode control [J]. Automatica, 2018, 96: 61-72
- [20] De P C, Frasca P. Robust self-triggered coordination with ternary controllers [J]. IEEE Transactions on Automatic Control, 2013, 58(12) : 3024-3038
- [21] Cheng B, Wu Z Z, Li Z K. Distributed edge-based event-triggered formation control [J]. IEEE Transactions on Cybernetics, 2019: 1-12
- [22] Velasco M, Fueiues J, Marti P. The self triggered task model for real-time control systems [C] // The 24th IEEE Conference on Real Time Systems Symposium, 2003
- [23] Heemels W P M H, Johansson K H, Tabuada P. An introduction to event-triggered and self-triggered control [C] // 2012 IEEE 51st Annual Conference on Decision and Control (CDC), 2012: 3270-3285
- [24] Yi X L, Liu K, Dimarogonas D V, et al. Dynamic event-triggered and self-triggered control for multi-agent systems [J]. IEEE Transactions on Automatic Control, 2019, 64(8) : 3300-3307
- [25] Dimarogonas D V, Frazzoli E, Johansson K H. Distributed self-triggered control for multi-agent systems [C] // 49th IEEE Conference on Decision and Control (CDC), 2010: 6716-6721
- [26] Anta A, Tabuada P. To sample or not to sample: self-triggered control for nonlinear systems [J]. IEEE Transactions on Automatic Control, 2010, 55(9) : 2030-2042
- [27] Nowzari C, Cortés J. Self-triggered coordination of robotic networks for optimal deployment [J]. Automatica, 2012, 48(6) : 1077-1087
- [28] Yi X L, Liu K, Dimarogonas D V, et al. Distributed dynamic event-triggered control for multi-agent systems [C] // 2017 IEEE 56th Annual Conference on Decision and Control (CDC), 2017: 6683-6698
- [29] Girard A. Dynamic triggering mechanisms for event-triggered control [J]. IEEE Transactions on Automatic Control, 2015, 60(7) : 1992-1997
- [30] Dolk V S, Borgers D P, Heemels W P M H. Dynamic event-triggered control: tradeoffs between transmission intervals and performance [C] // 53rd IEEE Conference on Decision and Control, 2014: 2764-2769
- [31] Tallapragada P, Chopra N. Decentralized event-triggering for control of nonlinear systems [J]. IEEE Transactions on Automatic Control, 2014, 59(12) : 3312-3324
- [32] Mazo M Jr, Cao M. Asynchronous decentralized event-triggered control [J]. Automatica, 2014, 50(12) : 3197-3203
- [33] Abdelrahim M, Postoyan R, Daafouz J, et al. Robust event-triggered output feedback controllers for nonlinear

- systems[J].Automatica,2017,75:96-108
- [34] Suh Y S,Nguyen V H,Ro Y S.Modified Kalman filter for networked monitoring systems employing a send-on-delta method[J].Automatica,2007,43(2):332-338
- [35] Dong L,Tang Y F,He H B,et al.An event-triggered approach for load frequency control with supplementary ADP[J].IEEE Transactions on Power Systems,2017,32(1):581-589
- [36] Yue D,Tian E G,Han Q L.A delay system method to design of event-triggered control of networked control systems[C]//2011 50th IEEE Conference on Decision and Control and European Control Conference,2011:1668-1673
- [37] Nowzari C, Cortés J. Distributed event-triggered coordination for average consensus on weight-balanced digraphs[J].Automatica,2016,68:237-244
- [38] Qian Y Y,Liu L,Feng G.Consensus of linear multi-agent systems via fully distributed event-triggered output-feedback control [C]//2017 36th Chinese Control Conference (CCC),2017:8171-8176
- [39] Deng C, Er M J, Yang G H, et al. Event-triggered consensus of linear multiagent systems with time-varying communication delays[J].IEEE Transactions on Cybernetics,2020,50(7):2916-2925
- [40] Ding L,Han Q L,Ge X H,et al.An overview of recent advances in event-triggered consensus of multiagent systems[J].IEEE Transactions on Cybernetics,2018,48(4):1110-1123
- [41] Cheng B, Li Z K. Fully distributed event-triggered protocols for linear multiagent networks[J].IEEE Transactions on Automatic Control,2019,64(4):1655-1662
- [42] Cheng B,Li Z K.Consensus of linear multi-agent systems via fully distributed event-triggered protocols[C]//2017 36th Chinese Control Conference (CCC),2017:8607-8612
- [43] Seyboth G S,Dimarogonas D V,Johansson K H.Event-based broadcasting for multi-agent average consensus [J].Automatica,2013,49(1):245-252
- [44] Garcia E,Cao Y C,Casbeer D W.Decentralized event-triggered consensus with general linear dynamics[J].Automatica,2014,50(10):2633-2640
- [45] Zhao Z Y,Chen G,Dai M X.Distributed event-triggered scheme for a convex optimization problem in multi-agent systems[J].Neurocomputing,2018,284:90-98
- [46] Chen W S,Ren W.Event-triggered zero-gradient-sum distributed consensus optimization over directed networks [J].Automatica,2016,65:90-97
- [47] Li H Q, Liu S, Soh Y C, et al. Achieving linear convergence for distributed optimization with zeno-like-free event-triggered communication scheme [C]//2017 29th Chinese Control and Decision Conference (CCDC),2017:6224-6229
- [48] Zhong M Y,Cassandras C G.Asynchronous distributed optimization with event-driven communication [J].IEEE Transactions on Automatic Control,2010,55(12):2735-2750
- [49] Kajiyama Y,Hayashi N,Takai S.Distributed subgradient method with edge-based event-triggered communication [J].IEEE Transactions on Automatic Control,2018,63(7):2248-2255
- [50] Liu S,Xie L H,Quevedo D E.Event-triggered quantized communication-based distributed convex optimization [J].IEEE Transactions on Control of Network Systems,2018,5(1):167-178
- [51] Tran N T,Wang Y W,Liu X K,et al.Distributed optimization problem for second-order multi-agent systems with event-triggered and time-triggered communication [J].Journal of the Franklin Institute,2019,356(17):10196-10215
- [52] Yi X L,Yao L S,Yang T,et al.Distributed optimization for second-order multi-agent systems with dynamic event-triggered communication[C]//2018 IEEE Conference on Decision and Control (CDC),2018:3397-3402
- [53] Shi C X,Yang G H.Distributed Nash equilibrium computation in aggregative games:an event-triggered algorithm [J].Information Sciences,2019,489:289-302
- [54] Su H G,Zhang H G,Liang Y L,et al.Online event-triggered adaptive critic design for non-zero-sum games of partially unknown networked systems [J].Neurocomputing,2019,368:84-98
- [55] Yue D,Tian E G,Han Q L.A delay system method for designing event-triggered controllers of networked control systems[J].IEEE Transactions on Automatic Control,2013,58(2):475-481
- [56] Yu H,Hao F.Input-to-state stability of integral-based event-triggered control for linear plants[J].Automatica,2017,85:248-255
- [57] Fan Y,Feng G,Wang Y,et al.Distributed event-triggered control of multi-agent systems with combinational measurements[J].Automatica,2013,49(2):671-675
- [58] Yi X L,Lu W L,Chen T P.Distributed event-triggered consensus for multi-agent systems with directed topologies[C]//2016 Chinese Control and Decision Conference (CCDC),2016:807-813

An overview of distributed event-triggered control for multi-agent systems

ZHANG Kaijie¹ WANG Dandan² LÜ Yuezuo²

¹ College of Science,Liaoning Technical University,Fuxin 123000

² School of Mathematics,Southeast University,Nanjing 211189

Abstract With the development of artificial intelligence and communication technology,it has become a key issue

to release the communication resources and extend the service life of broadband and hardware. Different from traditional time-based control, the event-triggered control takes samples only when the event trigger condition is satisfied to update the control policy. This paper first analyzes the event-triggered control in both continuous-time and discrete-time systems from aspect of event triggering communication, and discusses the main ideas of self-triggered control, edge-based event-triggered control, dynamic event-triggered control, and asynchronous event-triggered control. The application of event-triggered control is discussed in terms of consistency, optimization, and game. Two common theoretical analysis methods are given for stability demonstration. Besides, two suitable methods are also presented to exclude the Zeno behavior. At last, the paper summarizes the shortcomings of the current event-triggered work and puts forward some directions on future development.

Key words multi-agent systems; event-triggered control; distributed control; Zeno-behavior