

Distributed Model Based Event-Triggered Control for Synchronization of Multi-Agent Systems

Davide Liuzza* Dimos V. Dimarogonas**
Mario di Bernardo* Karl H. Johansson**

* *Department of Electrical Engineering and Information Technology,
University of Naples Federico II, Italy,
{davide.liuzza,mario.dibernardo}@unina.it.*

** *School of Electrical Engineering, Royal Institute of Technology
(KTH), Stockholm, Sweden, {dimos,kallej}@kth.se.*

Abstract: This paper investigates the problem of event-based control for the synchronization of networks of nonlinear dynamical agents. A distributed model based approach able to guarantee that all the agents converge to an adjustable synchronization region is derived. In such control scheme all the agents use a model of their neighborhood in order to generate triggering instants in which the local controller is updated and, if needed, local information is broadcasted to neighboring agents. The existence of a minimum lower bound between inter-event times is proven both for broadcasted information as well as for control signal updating, thus allowing implementation of the proposed strategy in applications where both the communication bandwidth and the maximum updating frequency of actuators are critical.

1. INTRODUCTION

The problem of controlling a large scale multi-agent system in order to reach a cooperative behaviour has been widely exploited in literature. In general, the problem is to design a distributed control law able to coordinate an ensemble of N interacting dynamical systems which communicate over a network of interconnections, see for example Boccaletti et al. [2006]. In particular, referring to this general problem, synchronization of dynamical systems has been investigated in Boccaletti et al. [2006], DeLellis et al. [2010], Zhao et al. [2011] as a paradigm for more specific behaviours like consensus algorithms see Olfati-Saber and Murray [2004] and platooning and formation control, see Arcaç [2007], Dimarogonas and Kyriakopoulos [2008].

Distributed control algorithms able to solve the cited problems have been, in general, realized in a continuous time fashion. However, continuous time control laws for such kind of networks are typically not easy or even not possible to be implemented in real applications. Indeed, in a future scenario a large number of dynamical systems is supposed to communicate over a wireless communication media, which represents a shared resource with limited throughput capacity. Classical time-driven control in Åström and Wittenmark [1997] require sampling the systems at a pre-specified time interval. This creates both the problem of synchronization of sampling instants among the interconnected systems and the simultaneous transmission of all the information over the network. Conversely, an event-triggered approach (see Tabuada [2007]) seems to be a better solution in decentralized control of multi-agent systems. Indeed, the control is updated only when an event criterion is satisfied, generally related to the state of the plant, thus saving computational and hardware

resources. Event-triggered has been successfully developed for consensus algorithms and for control and synchronization of linear dynamical systems. In particular, the case of consensus among single integrator agents is studied in Dimarogonas and Johansson [2009], Dimarogonas et al. [2012] both for a centralized and a decentralized solution. In Dimarogonas and Johansson [2009], the value of the control input for each agent is updated when a triggering condition defined as a function of its and neighbours' state is fulfilled. Consensus is then proved while the network is guaranteed not to reach an undesired accumulation point, so *Zeno* behaviour (Johansson et al. [1999]) are excluded. The case of controlling a network of general linear systems is addressed in Guinaldo et al. [2011]. Also in this paper the triggering events are defined for each node thanks to a function of the error between the current state and the previous broadcasted value. In the paper also a model-based solution is proposed, where the control input is now a continuous time function evaluated at the local level of each node using its own dynamical model and the (uncoupled) dynamical models of the neighbours. In this case for each node an event is generated any time the error between the real state and the predicted state exceeds a certain threshold. A similar idea to the proposed model-based solution can be found in Demir and Lunze [2012] where synchronization of linear dynamical agents is studied. Also in this case the control signals are continuous in time while the communication signals are piecewise constant.

In the current paper we will instead study a novel distributed event-triggered control scheme able to guarantee synchronization of nonlinear multi-agent systems by looking at distributed information related to each pair of connected agents. In particular, we will consider the sce-

nario where each agent is equipped with its own embedded processor and that can gather only information from a subset of the agents it directly communicates. Between each pair of connected agents, relative information of their state mismatch is considered in order to generate local events and update the control law. So, differently from the recent related literature, the event conditions will be defined on the relative state errors between coupled pairs of agents, rather than on their own states. The proposed idea follows a model based approach since each agent knows the dynamical model of its neighbours and predicts their state evolutions between any two consecutive triggering events. We will show how the dynamical agents achieve synchronization with appropriate event-triggered policies. Both the control and communication signals will be piecewise constant and, furthermore, we will also ensure that the overall switched system does not exhibit Zeno behaviour. In addition to this, we will also guarantee that with an appropriate triggering policy that the control law for each agent will be updated with a lower bound for the inter-event times.

2. BACKGROUND AND PROBLEM STATEMENT

2.1 Algebraic Graph Theory and Mathematical Background

Given a set of N nodes (also called vertices) $\mathcal{N} = \{1, 2, \dots, N\}$ and a set of edges $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$, a graph $G(\mathcal{N}, \mathcal{E})$ can be represented by the *adjacency matrix* $A = A(G) = [a_{ij}] \in \mathbb{R}^{N \times N}$, where $a_{ii} = 0$ for all i , and $a_{ij} = 1$, with $i \neq j$, if there is an edge between nodes i and j , i.e. $(i, j) \in \mathcal{E}$, while it is $a_{ij} = 0$ otherwise. In the case of undirected graphs, the matrix A is symmetric and so $a_{ij} = a_{ji}$. When $a_{ij} \neq 0$, for $i \neq j$, the two nodes i and j are said to be *adjacent* or *neighbours*. Using the adjacency matrix, we can easily describe the neighbourhood of a generic node $i \in \{1, \dots, N\}$ as $\mathcal{N}_i \subseteq \mathcal{N} = \{j : a_{ij} \neq 0\}$, while we indicate with $N_i = |\mathcal{N}_i|$ the number of neighbours of node i , also called *degree* of node i .

A *path* from node i to node j is a sequence of nodes in the graph starting from i and ending with j such that consecutive nodes are adjacent.

An undirected graph is said to be *connected* if for any pair of nodes i and j there exists a path between them.

The *Laplacian matrix* $L = L(G) \in \mathbb{R}^{N \times N}$ is defined as $L = \Delta - A$, where Δ is the diagonal matrix of all nodes' degrees, i.e. $\Delta = \text{diag}\{N_1, N_2, \dots, N_N\}$. A Laplacian matrix has at least one zero eigenvalue and in case of undirected graph all the eigenvalues are real and can be ordered as $0 = \lambda_1(L) \leq \lambda_2(L) \leq \dots \leq \lambda_N(L)$. If the graph is connected, the zero eigenvalue is unique and $\lambda_2(L) > 0$.

Here we now give the definition of Lipschitz function and one-sided Lipschitz function, see for instance Agarwal and Lakshmikantham [1993].

Definition 1. A function $f(x) = \mathbb{R}^n \mapsto \mathbb{R}^n$ is said to be *globally Lipschitz* if there exists a constant $L_f > 0$ such that $\|f(x) - f(y)\|_2 \leq L_f \|x - y\|_2$, for all $x, y \in \mathbb{R}^n$.

Definition 2. A function $f(x) = \mathbb{R}^n \mapsto \mathbb{R}^n$ is said to be *one-sided Lipschitz* if there exists a constant $L'_f > 0$ such that $[f(x) - f(y)]^T (x - y) \leq L'_f \|x - y\|_2^2$, for all $x, y \in \mathbb{R}^n$.

Notice that it is immediate to prove that a Lipschitz function with Lipschitz constant L_f is also one-sided Lipschitz with the same constant.

2.2 Model-based Event-triggered

In this paper we will consider N identical dynamical agents of the form:

$$\dot{x}_i = f(t, x_i) + u_i \quad x_i, u_i \in \mathbb{R}^n, t \geq 0, \quad \forall i = 1, \dots, N, \quad (1)$$

We aim at guaranteeing a coordinated motion (synchronization) for the systems in (1) by considering a distributed event-triggered control law. More precisely, as done in Sun et al. [2009], Zhao et al. [2011], we define the average trajectory as $\bar{x}(t) = \frac{1}{N} \sum_{j=1}^N x_j(t)$, and the synchronization errors as $e_i(t) = x_i(t) - \bar{x}(t)$, which in stack form is $e(t) = (e_1^T, \dots, e_N^T)^T \in \mathbb{R}^{nN}$. We want to prove the goal of a *bounded* (or *practical*) synchronization, and so that there exists an $\epsilon > 0$ sufficiently small such that¹

$$\lim_{t \rightarrow \infty} \|e(t)\|_2 \leq \epsilon. \quad (2)$$

In order to ensure synchronization between systems in (1), we imagine the scenario where each agent is able to exchange information between a subset of the other agents. The resulting communication network, that here we suppose bidirectional, can be described by an undirected adjacency matrix A . In other words, if $a_{ij} \neq 0$, there exists a communication channel between nodes i and j . Furthermore, we also consider that each agent is equipped with its own embedded processor able to execute a local control law based on the prediction of the evolution of its neighbours. Thanks to this local information, each node will execute an event-triggered update of its controller. In particular, in the event-triggered scheme we propose, at each node i we associate:

- (1) $\{t_k^{ij}\}_{k^{ij}=0}^\infty : \mathbb{N} \mapsto [0, +\infty)$ a time sequence of events on node i referring to information from node j , where $a_{ij} \neq 0$ and where k^{ij} is the index of the sequence related to the pair (i, j) ;
- (2) $\{t_k^i\}_{k^i=0}^\infty : \mathbb{N} \mapsto [0, +\infty)$ a time sequence of the instants when node i updates its control input $u_i(t)$, where k^i is the index of the sequence related to the updating of $u_i(t)$.

In both the cases, for any index $k^{ij} \in \mathbb{N}$ (or $k^i \in \mathbb{N}$) we have that $t_k^{ij} \leq t_{k+1}^{ij}$ (or $t_k^i \leq t_{k+1}^i$).

For each sequence $\{t_k^{ij}\}_{k^{ij}=0}^\infty$ we introduce the *last function* $l^{ij}(t) : [0, +\infty) \mapsto \mathbb{N}$ defined as

$$l^{ij}(t) = \arg \min_{k^{ij} \in \mathbb{N}: t \geq t_k^{ij}} \{t - t_k^{ij}\}.$$

So, for each time instant t , $t_{l^{ij}(t)}^{ij}$ is the most recent event occurred to i with respect to j , while with $t_{l^{ij}(t)+1}^{ij}$ we indicate the next event.

¹ With the symbol (2) we mean that for all $\nu > 0$ there exists a $t_\nu > 0$ such that for all $t > t_\nu$ we have that $\|e(t)\|_2 \leq \epsilon + \nu$. So, this does not imply that the limit in (2) exists, but we use the same mathematical notation as in the case of an existing limit due to the analogy with the eventual boundedness of signal $e(t)$.

Analogously, we define the function $l^i(t)$ for the sequence $\{t_k^i\}_{k^i=0}^\infty$.

As will be clear in what follows, the last indices $l^{ij}(t)$ and $l^i(t)$ will be used to generate implicitly the sequences $\{t_k^{ij}\}_{k^{ij}=0}^\infty$ and $\{t_k^i\}_{k^i=0}^\infty$. In particular, borrowing a notation used for hybrid systems Goebel and Teel [2006], after an event the counter $l^{ij}(t)$ will be updated to $l^{ij+} = l^{ij} + 1$, where by l^{ij+} we mean the value of $l^{ij}(t)$ immediately after a new event. Analogously for the counter $l^i(t)$ that will be updated to $l^{i+} = l^i + 1$. It is worth mentioning that, although the communication graph is undirected, events related to coupled pairs (i, j) are, in general, not synchronous and so $t_{l^i}^{ij} \neq t_{l^i}^{ji}$. For this reason sequences $\{t_k^{ij}\}_{k^{ij}=0}^\infty$ and $\{t_k^{ji}\}_{k^{ji}=0}^\infty$ are, in the general case, different.

We are now ready to write the control input $u_i(t)$ for each i -th system in (1) as the piecewise constant signal

$$u_i(t) = c \sum_{i=1}^N a_{ij} \Gamma e_{ij}(t_l^{ij}) \quad t \in [t_l^i, t_{l^i+1}^i), \quad (3)$$

where $c > 0$ is a *coupling gain*, $\Gamma = \Gamma^T > 0$ is the *inner coupling matrix* and where we define $e_{ij}(t) = x_j(t) - x_i(t)$. In this case $e_{ij}(t)$ is evaluated at the time instant t_l^{ij} .

The control input (3) leads to a diffusively coupled event-triggered dynamical network of nodes' dynamics

$$\dot{x}_i(t) = f(t, x_i(t)) + c \sum_{i=1}^N a_{ij} \Gamma e_{ij}(t_l^{ij}), \quad t \in [t_l^i, t_{l^i+1}^i) \quad (4)$$

where here and in what follows we omit the explicit dependence of l^{ij} and l^i on time. The aim of the paper is to study under which conditions and under which sequences of triggering events $\{t_k^{ij}\}_{k^{ij}=0}^\infty$ and $\{t_k^i\}_{k^i=0}^\infty$, for all $(i, j) \in \mathcal{E}$, the network (4) guarantees the bounded synchronization defined in (2).

3. BOUNDED EVENT-TRIGGERED SYNCHRONIZATION

In this paper we assume exact knowledge of the dynamic model (1) describing the dynamics of the nodes. Furthermore, each node is supposed to know the initial conditions of its neighbours (or the value of the state at a specific time instant, for example at the first trigger). Therefore, each node i can compute from any event at time t_k^{ij} the evolution

$$\varphi_f(t - t_k^{ij}, t_k^{ij}, x_j(t_k^{ij})), \quad \forall j \in \mathcal{N}_i. \quad (5)$$

The case where exact knowledge of the dynamical model is not available is more cumbersome and cannot be studied here due to space limitation. Such case will be presented later in a journal publication.

Obviously, in order to evaluate (5), node i must also have information on the current control input $u_j(t)$ acting on each of its neighbours. Later we will present an algorithm able to guarantee that this information is shared among nodes. However, we firstly focus on the triggering events occurring at a generic node i .

For all $(i, j) \in \mathcal{E}$ we define the *trigger error*

$$\tilde{e}_{ij}(t) := e_{ij}(t_l^{ij}) - e_{ij}(t), \quad t \in [t_l^i, t_{l^i+1}^i). \quad (6)$$

Notice that, as we said before, events referred to node i with respect to j are, in general, not synchronous with the events referred to j with respect to i . For this reason, the pair (i, j) is treated here as an oriented link and in general $\tilde{e}_{ij}(t) \neq \tilde{e}_{ji}(t)$.

For all pairs (i, j) , we define the *trigger function*

$$p_{ij}(t, \tilde{e}_{ij}(t)) = \|\tilde{e}_{ij}(t)\|_2 - \varsigma_{ij}(t), \quad (7)$$

where $\varsigma_{ij}(t)$ is a continuous-time nonincreasing *threshold function*. Then, an event occurs when the following condition is violated

$$p_{ij}(t, \tilde{e}_{ij}(t)) < 0. \quad (8)$$

Using the trigger function, we generate the sequence of events for each node $i \in \mathcal{N}$ according to the following distributed algorithm:

Algorithm 1.

1. Node i broadcasts to its neighbours the triplet $(t_0^i, x_i(t_0^i), u_i(t_0^i))$. This information represents the first triggering event and allows the nodes $j \in \mathcal{N}_i$ to initialize the algorithm;
2. Node i broadcasts the value of its current control input $u_i(t_l^i)$;
3. Node i computes the flows in (5) for all its neighbouring nodes $j \in \mathcal{N}_i$ using the dynamical model $\dot{x}_j = f(t, x_j(t_l^j)) + u_j(t_l^j)$. Thanks to the evaluation of the neighbours' flows, node i can compute the trigger error (6) and monitor condition (8) for all of its neighbours;
4. Once condition (8) is violated, say for a certain node $h \in \mathcal{N}_i$, node i updates the value $e_{ih}(t_l^{ih})$ to $e_{ih}(t_{l^i+1}^{ih})$. Then, after the event the counter $l^{ih}(t)$ will be updated to $l^{ih+} = l^{ih} + 1$. At the same time the current value $\tilde{e}_{ih}(t = t_l^{ih})$ will be reset by (6) evaluated at the new event $l^{ih} + 1$;
5. Node i computes the new control law u_i and updates $l^{i+} = l^i + 1$. Hence, the last updating event of the controller happens once the new value e_{ih} is considered and, therefore, $t_l^i = t_{l^i}^{ih}$. The control input takes the value

$$u_i(t) = \sum_{j=1}^N a_{ij} e_{ij}(t_l^{ij}), \quad t \in [t_l^i, t_{l^i+1}^i), \quad (9)$$

Thus, the new value of the input is held until the next output trigger of node i ;

6. Repeat from step 2.

Note that, as every node that triggers changes its control input (step 5) and broadcast it to its neighbours (step 6), then all the nodes $j \in \mathcal{N}_i$ can update their dynamic model of i taking into account the new input $u_i(t_l^i)$ and the current state $x_i(t_l^i)$ (step 3). As we have already said in Section 2, the time event t_l^j , with j generic neighbour of node i , and the last update event of the control input t_l^i implicitly define the sequences $\{t_k^{ij}\}_{k^{ij}=0}^\infty$ and $\{t_k^i\}_{k^i=0}^\infty$.

Remark 1. It is worth to notice that due to the updating criterion expressed in the step 5, in control input (9) each e_{ij} holds the value from its past event at t_l^{ij} . So, in general, $t_l^{ij} \neq t_l^{ih}$, with $h, j \in \mathcal{N}_i$ being two different neighbours of node i . On the other hand, due to the symmetry of the trigger condition expressed by (6)-(7)-(8), if we choose $\varsigma_{ij}(t) = \varsigma_{ji}(t)$ then when node i triggers and updates its

control because of the (8) with respect to node j does not hold anymore, then also node j triggers for the same reason and we have that $t_l^{ij} = t_l^{ji}$. This implies the symmetry of the coupling strengths between any connected pairs (i, j) .

Note that step 5 can be substituted with the following step:

5'. Node i updates $l^{i+} = l^i + 1$, so $t_l^i = t_l^{ih}$ and computes the control law u_i using the expression

$$u_i(t) = \sum_{j=1}^N a_{ij} e_{ij}(t_l^i), \quad t \in [t_l^i, t_{l^i+1}^i), \quad (10)$$

Then, all trigger errors (6) are reset, for all $j \in \mathcal{N}_i$.

Basically, once the first trigger occurs, say for $\tilde{e}_{ih}(t)$, then not only the current value e_{ih} will be updated and the corresponding trigger error (6) reset, but also all other values e_{ij} with $j \in \mathcal{N}_i$.

Remark 2. When using step 5', all triggers related to pairs (i, j) , with $j \in \mathcal{N}_i$, are forced to be synchronous and, moreover, $t_l^{ij} = t_l^{ih}$ for all $j, h \in \mathcal{N}_i$. Conversely, when step 5' is used at a generic time instant t , we have $e_{ij}(t) \neq e_{ji}(t)$. So, considering (10) instead of (9) all e_{ij} are updated at the same time but symmetry of the control actions between coupled pairs (i, j) is lost.

We denote the algorithm obtained by using step 5' as Algorithm 1'. Notice that, both control schemes lead to piecewise constant communication and control signals. We now give a bounded synchronization result.

Theorem 3. Let us consider the event-triggered connected network (4), where the function $f(t, x)$ is Lipschitz continuous with respect to x with Lipschitz constant L_f and let us considering a coupling gain c such that

$$L_f - c\lambda_2(L \otimes \Gamma) < 0. \quad (11)$$

Let us also consider thresholds functions $\varsigma_{ij}(t)$ such that $\lim_{t \rightarrow \infty} \varsigma_{ij}(t) = \bar{\varsigma}_{ij}$, with $\bar{\varsigma}_{ij} > 0$ for all i, j such that $a_{ij} \neq 0$. Then both Algorithm 1 and Algorithm 1' guarantee bounded synchronization of the network. Furthermore, no Zeno behaviours will occur.

Proof. We will split the proof in two steps. Firstly we will prove bounded synchronization and then that no Zeno behaviours occur. Firstly, we rewrite equation (4) as

$$\begin{aligned} \dot{x}_i(t) &= f(t, x_i) + c \sum_{j=1}^N a_{ij} \Gamma e_{ij}(t) + \\ & c \sum_{j=1}^N a_{ij} \Gamma \tilde{e}_{ij}(t) \quad \forall i = 1, \dots, N. \end{aligned} \quad (12)$$

Step 1. Let us consider the candidate Lyapunov function $V(e(t)) = \frac{1}{2} e^T e$ defined in the error space and let us consider its derivative with respect to time. We obtain

$$\begin{aligned} \dot{V}(e(t)) &= \sum_{i=1}^N e_i^T \dot{e}_i = \sum_{i=1}^N e_i^T f(t, x_i) - \sum_{i=1}^N e_i^T \dot{\bar{x}} - \\ & ce^T (L \otimes \Gamma) e + \sum_{i=1}^N e_i^T \sum_{j=1}^N a_{ij} \Gamma \tilde{e}_{ij}. \end{aligned}$$

Now, taking into account that $\sum_{i=1}^N e_i^T \dot{\bar{x}} = 0$, adding and subtracting $\sum_{i=1}^N e_i^T f(t, \bar{x})$ and using the one-sided Lipschitz property in Definition 2, we can write the following inequality

$$\dot{V} \leq L_f e^T e - ce^T (L \otimes \Gamma) e + \|e\|_2 \sqrt{N} N_{\max} \|\Gamma\|_2 \varsigma(t),$$

where L_f is the Lipschitz constant of the function f , $N_{\max} \leq N - 1$ is the maximum degree of the graph A and $\varsigma(t) = \max_{i,j} \varsigma_{ij}(t)$. Writing $e = a\hat{e}$, where $a = \|e\|_2$ is the module of the error and $\hat{e} = \frac{1}{\|e\|_2} e$ is the unitary vector associated to e , the above inequality can be rewritten as

$$\dot{V}(e) \leq (L_f - c\lambda_2(L \otimes \Gamma)) a^2 + c\sqrt{N} N_{\max} \|\Gamma\|_2 \varsigma(t) a. \quad (13)$$

Now, since c is chosen in order to fulfill inequality (11), then the error trajectory $e(t)$ converges to the invariant region $\|e(t)\|_2 \leq \epsilon$, where

$$\epsilon = \frac{c\sqrt{N} N_{\max} \|\Gamma\|_2 \varsigma(t)}{c\lambda_2(L \otimes \Gamma) - L_f}, \quad (14)$$

or, in compact form $\epsilon = \epsilon_\varsigma \varsigma(t)$, in order to emphasize that the value of the bound of the global synchronization error $e(t)$ depends on $\varsigma(t)$ times a finite positive constant. As $\lim_{t \rightarrow +\infty} \varsigma(t) = \bar{\varsigma} > 0$, bounded synchronization is then ensured.

Step 2. Let us consider the dynamics of the error between a generic connected pair of nodes $(i, h) \in \mathcal{E}$. Such dynamics can be expressed as $\dot{e}_{ih}(t) = \dot{x}_h(t) - \dot{x}_i(t)$ thus,

$$\begin{aligned} \dot{e}_{ih} &= f(t, x_h) + c \sum_{j=1}^N a_{hj} \Gamma e_{hj}(t) + c \sum_{j=1}^N a_{hj} \Gamma \tilde{e}_{hj}(t) - \\ & f(t, x_i) - c \sum_{j=1}^N a_{ij} \Gamma e_{ij}(t) - c \sum_{j=1}^N a_{ij} \Gamma \tilde{e}_{ij}(t). \end{aligned}$$

Then, we can write

$$\begin{aligned} \|\dot{e}_{ih}\|_2 &\leq \|f(t, x_h) - f(t, x_i)\|_2 + \\ & c \sum_{j=1}^N a_{hj} \|\Gamma\|_2 \|e_{hj}(t)\|_2 + c \sum_{j=1}^N a_{hj} \|\Gamma\|_2 \|\tilde{e}_{hj}(t)\|_2 + \\ & c \sum_{j=1}^N a_{ij} \|\Gamma\|_2 \|e_{ij}(t)\|_2 + c \sum_{j=1}^N a_{ij} \|\Gamma\|_2 \|\tilde{e}_{ij}(t)\|_2. \end{aligned} \quad (15)$$

Now, taking into account that f is Lipschitz we have

$$\|e_{ih}(t)\|_2 \leq 2\|e(t)\|_2 \leq 2 \sup_{t' \in [t, +\infty)} \|e(t')\|_2 \leq 2\tilde{b}(t), \quad (16)$$

where $\tilde{b}(t)$ is the nonincreasing piecewise smooth continuous function defined as

$$\tilde{b}(t) = \begin{cases} \|e(t)\|_2 & \text{if } \|e(t)\|_2 > \epsilon_\varsigma \varsigma(t) \\ \epsilon_\varsigma \varsigma(t) & \text{if } \|e(t)\|_2 \leq \epsilon_\varsigma \varsigma(t). \end{cases} \quad (17)$$

Therefore, we can rewrite inequality (15) as

$$\|\dot{e}_{ih}(t)\|_2 \leq 2[L_f + c\|\Gamma\|_2(N_h + N_i)]\tilde{b}(t) + c\|\Gamma\|_2(N_h + N_i)\varsigma(t), \quad (18)$$

where N_i and N_h are the degrees of nodes i and h respectively. Let $p_1 = 2[L_f + c\|\Gamma\|_2(N_h + N_i)]$ and $p_2 = c\|\Gamma\|_2(N_h + N_i)$, at the last trigger event $t = t_l^{ih}$ we obtain from (18)

$$\|\dot{e}_{ih}(t)\|_2 \leq p_1 \tilde{b}(t_l^{ih}) + p_2 \varsigma(t_l^{ih}), \quad \forall t \geq t_l^{ih}. \quad (19)$$

Now, in order to prove that Zeno behaviours do not occur in the network, we will show that for all triggering instants t_k^{ih} there exists a nonzero lower bound $\tau_{ih}(t_l^{ih})$ for the inter-event time between the last trigger event t_l^{ih} and the next t_{l+1}^{ih} for any generic pair (i, h) verifying the condition

$$t_{k+1}^{ih} - t_k^{ih} \geq \tau_{ih}(t_l^{ih}).$$

To do this, let us consider the dynamics of the triggering error $\tilde{e}_{ih}(t)$ at time instants $t > t_l^{ih}$. Clearly, the following considerations will be valid not only for the last event instant t_l^{ih} but for all instants t_k^{ih} , since the sequence $\{t_k^{ih}\}_{k=0}^{\infty}$ is implicitly defined by the sequence of the last events. We can write

$$\begin{aligned} \|\tilde{e}_{ih}(t)\|_2 &\leq \int_{t_l^{ih}}^t \|\dot{\tilde{e}}_{ih}(s)\|_2 ds = \\ &\int_{t_l^{ih}}^t \|\dot{e}_{ih}(s)\|_2 ds = \int_{t_l^{ih}}^t \|\dot{e}_{ih}(s)\|_2 ds. \end{aligned} \quad (20)$$

Taking into account inequality (19) and considering $t = t_l^{ih} + \tau_{ih}(t_l^{ih})$ from the above formula we obtain the nonzero lower bound

$$\tau_{ih}(t_l^{ih}) = \frac{\bar{\varsigma}_{ih}}{p_1 \tilde{b}(t_l^{ih}) + p_2 \varsigma(t_l^{ih})}. \quad (21)$$

Remark 3. It is easy to note that Theorem 3 holds both for Algorithm 1 and for Algorithm 1' since the proof is independent from the choice of step 5 or step 5'. Despite this, since in Algorithm 1' all e_{ij} with $j \in \mathcal{N}_i$ are updated at the same time instant t_l^i and the corresponding errors \tilde{e}_{ij} reset, this leads to a non zero lower bound also between any two consecutive updating events of the control law. For this reason, Algorithm 1' can be implemented in all applications where criticality on actuators does not allow to change the control input arbitrarily fast.

Notice that if we suppose different choices for the threshold functions $\varsigma_{ij}(t)$ in the proposed event-triggered strategies, other goals could be considered, i.e. asymptotic synchronization. However, such study is beyond the scope of the paper and will be presented elsewhere.

4. NUMERICAL EXAMPLE

In order to show the effectiveness of the strategy proven in Theorem 3, we consider a network of Chua's circuits, see Matsumoto [1984]. The Chua's system is a well studied dynamical system and it is often taken in literature as a paradigm for chaos (de Magistris et al. [2012]) and for studying synchronization.

Specifically, we consider a connected graph of identical Chua's circuits, whose dynamical models $\dot{x}_i = f(x_i)$ have expression

$$\begin{aligned} \dot{x}_{i1} &= \alpha [x_{i2} - x_{i1} - \varphi(x_{i1})], \\ \dot{x}_{i2} &= x_{i1} - x_{i2} + x_{i3}, \\ \dot{x}_{i3} &= -\beta x_{i2}, \end{aligned}$$

where, following Matsumoto [1984] we set $\alpha = 10$, $\beta = 17.30$, and where $\varphi(x_{i1}) = bx_{i1} + (a - b)(|x_{i1} + 1| - |x_{i1} - 1|)/2$, with $a = -1.34$, $b = -0.73$. It is easy to notice that the vector field of the Chua system is Lipschitz and an upper bound for the Lipschitz constant is

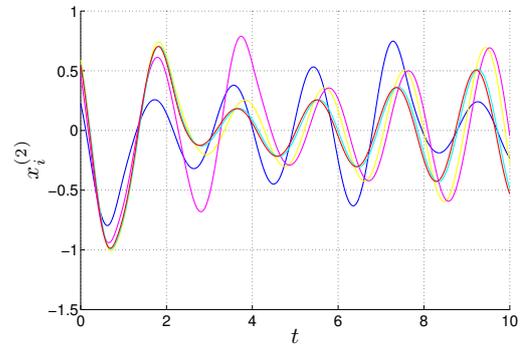
$$L_f = \left\| \begin{bmatrix} -\alpha & \alpha & 0 \\ 1 & -1 & 1 \\ 0 & -\beta & 0 \end{bmatrix} \right\|_2 + \alpha|a|,$$

which gives in this case $L_f = 34.2$.

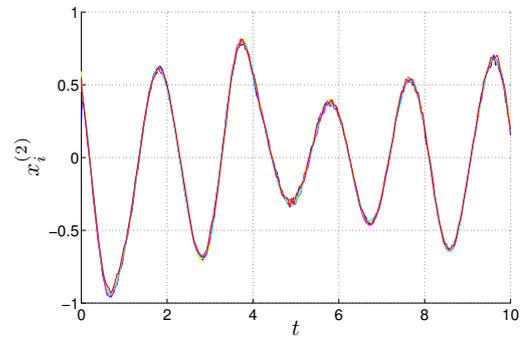
We simulate a network of five Chua systems with adjacency matrix

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix},$$

considering as matrix Γ the identity matrix and using the coupling strategy proposed in Algorithm 1. In order to guarantee inequality (11) these data lead to a minimum coupling $c = 13.7$.



(a)



(b)

Fig. 1. Time evolution of the state components $x_i^{(2)}(t)$ for the network of Chua systems with static thresholds: (a) uncoupled case; (b) coupled case.

Simulation in Fig. 1(a) shows the evolution of the uncoupled nodes considering random initial conditions in the domain of the chaotic attractor. From the same initial conditions and coupling the network it is possible to observe bounded synchronization in Fig. 1(b). Simulations have been performed applying Algorithm 1, setting an identical static threshold $\varsigma_{ij}(t) = \bar{\varsigma}$ for all connected pairs (i, j) , with $\bar{\varsigma} = 0.1$, and choosing to plot the second state components as representative of the whole state. The number of triggers for each node in time intervals of unitary length is reported in Tab. 1.

Table 1. Number of triggers in unitary intervals for the network of Chua systems with static thresholds

	[0, 1) s	[1, 2) s	[2, 3) s	[3, 4) s	[4, 5) s	[5, 6) s	[6, 7) s	[7, 8) s	[8, 9) s	[9, 10] s
node 1	32	33	26	25	22	24	18	12	17	28
node 2	35	25	27	39	33	30	29	32	24	38
node 3	29	25	19	22	23	25	20	16	18	21
node 4	32	24	22	27	36	39	27	12	36	35
node 5	15	15	14	21	23	32	30	24	30	22

5. CONCLUSIONS

We presented a model based event-triggered strategy for practical synchronization of networks of nonlinear dynamical agents. Trigger conditions are defined on the relative errors between connected pairs of agents. The knowledge of the dynamical model of the agents and the broadcasted information of the current value of the piecewise control signal of each node to its neighbours allow each agent to compute the dynamical flow of its neighbourhood and to evaluate the trigger condition. The proposed strategy guarantees practical synchronization with an adjustable value of states' mismatch among the agents. Furthermore, the Zeno behaviour is excluded. The results of the paper are supported through a numerical example. The case of different choices of the threshold functions in order to achieve asymptotic synchronization and the problem of robustness to model uncertainties are more cumbersome tasks and will be presented in a journal paper currently in preparation.

REFERENCES

- R. P. Agarwal and V. Lakshmikantham. *Uniqueness and Nonuniqueness Criteria for Ordinary Differential Equations*. Singapore: World Scientific, 1993.
- M. Arcak. Passivity as a design tool for group coordination. *IEEE Transactions on Automatic Control*, 52:1380–1390, 2007.
- K. J. Åström and B. Wittenmark. *Computer Controlled Systems*. Prentice Hall, 1997.
- S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. U. Hwang. Complex networks: structure and dynamics. *Physics Reports*, 424:175–308, 2006.
- M. de Magistris, M. di Bernardo, E. Di Tucci, and S. Manfredi. Synchronization of Networks of Non-identical Chua Circuits: analysis and experiments. *IEEE Transactions on Circuits and Systems*, 59:1029–1041, 2012.
- P. DeLellis, M. diBernardo, T. E. Gorochoowski, and G. Russo. Synchronization and control of complex networks via contraction, adaptation and evolution. *IEEE Circuits and System Magazine*, 3:64–82, 2010.
- O. Demir and J. Lunze. Event-based synchronisation of multi-agent systems. In *Proceedings of the 4th IFAC Conference on Analysis and Design of Hybrid Systems*, 2012.
- D. V. Dimarogonas and K. H. Johansson. Event-triggered Control for Multi-Agent Systems. In *Proceedings of the 48th Conference on Decision and Control*, 2009.
- D. V. Dimarogonas and K. J. Kyriakopoulos. A connection between formation infeasibility and velocity alignment in kinematic multi-agent systems. *Automatica*, 44:2648–2654, 2008.
- D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson. Distributed Event-Triggered Control for Multi-Agent Systems. *IEEE Transactions of Automatic Control*, 57:1291–1297, 2012.
- R. Goebel and A. R. Teel. Solutions to hybrid inclusions via set and graphical convergence with stability theory applications. *Automatica*, 47:573–587, 2006.
- M. Guinaldo, D. V. Dimarogonas, K. H. Johansson, J. Sánchez, and S. Dormido. Distributed Event-Based Control for Interconnected Linear Systems. In *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, 2011.
- K. H. Johansson, M. Egerstedt, J. Lygeros, and S. S. Sastry. On the regularization of zeno hybrid automata. *Systems and Control Letters*, 38:141–150, 1999.
- T. Matsumoto. A chaotic attractor from Chua's circuit. *IEEE Transaction on Circuits and Systems*, 31:1055–1058, 1984.
- R. Olfati-Saber and R. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49:1521–1533, 2004.
- J. Sun, E. M. Bollt, and T. Nishikawa. Master stability functions for coupled nearly identical dynamical systems. *Europhysics Letters*, 85:60011, 2009.
- P. Tabuada. Event-Triggered Real-Time Scheduling of Stabilizing Control Tasks. *IEEE Transactions on Automatic Control*, 52:1680–1684, 2007.
- J. Zhao, D. J. Hill, and T. Liu. Synchronization of dynamical networks with nonidentical nodes: Criteria and control. *IEEE Transactions on Circuits and Systems I*, 58:584–594, 2011. ISSN 1549-8328. doi: 10.1109/TCSI.2010.2072330.