

Global Synchronization of Clocks in Directed Rooted Acyclic Graphs: A Hybrid Systems Approach

Muhammad U. Javed*, Jorge I. Poveda, and Xudong Chen

Abstract—In this paper, we study the problem of robust global synchronization of resetting clocks in multi-agent networked systems, where by *robust global synchronization* we mean synchronization that can be achieved from *all* initial conditions and is insensitive to small perturbations. In particular, we address the following question: Given a set of homogeneous agents with periodic clocks, what kind of information flow topologies will guarantee that the resulting networked systems can achieve robust global synchronization? To address the question, we rely on the use of robust hybrid dynamical systems. Using the hybrid-system approach, we provide a partial solution to the question: Specifically, we show that one can achieve robust global synchronization if the underlying information flow topology is a rooted acyclic digraph. Such a result is complementary to the existing results in [1] and [2] by Poveda & Teel, in which strongly connected digraphs are considered as the underlying information flow topologies of the networked systems. We have further computed an upper bound on the convergence time for a networked system to reach global synchronization. In particular, the computation reveals the relationship between convergence time and the structure of the underlying digraph. We illustrate our theoretical findings via numerical simulations toward the end of the paper.

I. INTRODUCTION

In recent years, the problem of coordination and control of networked multi-agent systems (MAS) has been a major research area in control theory. For instance, analytical tools for MAS were studied in [3], [4], [5], [6], [7] in the context of networked systems, event-triggered control, and synchronization of clocks. Multi-agent networked systems have also been analyzed using graph-theoretic methods in [8], and geometric approaches in [9], [10]. In this work, we focus on one particular control problem that emerges in MAS, namely the global synchronization of a collection of homogeneous agents (i.e., agents with identical structures and parameters) with periodic behaviors. We assume that these agents can only communicate with their local neighbors, for which we will use a directed graph to describe the information flow topology. Such a problem finds applications in many areas where a globally synchronized periodic behavior is needed, but only local interactions between agents are allowed. These applications include power systems, biological systems, and sampled-data systems where the synchronization problems emerges in a natural way. This has motivated the development of several deterministic and stochastic synchronization algorithms in [7], [11], [12] and [13], to just name a few.

*Corresponding author. M. U. Javed, J. I. Poveda, and X. Chen are with the Department of Electrical, Computer and Energy Engineering, at the University of Colorado, Boulder, USA. Emails: {muhammad.javed, jorge.poveda, xudong.chen}@colorado.edu. X. Chen would like to acknowledge the support of grant NSF ECCS-1809315.

In this paper, we focus on a particular phase synchronization problem, where a group of agents with periodic clocks aims to eventually aligned their phase position in a distributed way. This problem is equivalent to the synchronization of identical oscillators flowing on the unit circle $\mathbb{S}^1$, a problem that is known for its infeasibility of achieving global synchronization that is robust against perturbation by using smooth feedback control laws [14], [15]. On the other hand, under certain assumptions, robust global synchronization can be achieved if one implements a hybrid controller in a cyclic graph [16], or, alternatively, if there exists a global cue in the network [17]. More recently, it was shown in [1], [2] that a *hybrid set-valued resetting algorithm (HSRA)* achieves robust global synchronization in any network whose underlying information flow topology is characterized by a directed strongly connected graph, provided that some mild assumptions on tunable parameters of the agents are satisfied.

The goal of this paper is to extend existing results [1], [2] and address digraphs that are beyond the class of strongly connected ones. The general question we aim to address is the following: *Given a set of homogeneous agents with periodic clocks sharing the same frequency, what kind of information flow topologies will guarantee that the resulting networked system can achieve robust global synchronization?*

We provide in the paper a partial solution to the above question by characterizing a new class of digraphs, namely rooted acyclic digraphs, for which robust global synchronization can be achieved using the framework of HSRA. Moreover, we provide variations of existing algorithms in [1] and [2] that can yield a better performance in terms of convergence time in some specific cases. We summarize below the contributions of the paper:

- 1) We provide a negative result which says that if the underlying network topology is not a rooted digraph, then the entire network system cannot achieve global synchronization under the HSRA in any case.
- 2) We show that the HSRA can be used to achieve robust global synchronization if the underlying digraph is rooted acyclic. We also compute the convergence time for the networked system to reach synchronization: It is bounded above by the multiplication of the *depth* of the rooted acyclic digraph (see Def. 2.2 in the next section) and ω^{-1} (where ω is the frequency of the clock).
- 3) We further show that in a special case where the common parameter shared by all agents is set to be certain extreme value, global synchronization can be achieved if and only if the digraph is rooted acyclic. Moreover, we show that in this special case, global

synchronization can be achieved in at most ω^{-1} units of time, which is the smallest upper bound on the convergence time in any case.

The remainder of the paper is organized as follows: In Section II, we present preliminaries and definitions related to graph theory and hybrid dynamical systems. In Section III, we first introduce the model for a network of resetting clocks and then formulate the synchronization problem. We also present the main result as well as a sketch of a proof in the section. Numerical simulations are given in Section IV. We provide conclusions at the end.

II. PRELIMINARIES

A. Notation

The set of (nonnegative) real numbers is denoted by $(\mathbb{R}_{\geq 0})$. The set of (nonnegative) integers is denoted by $(\mathbb{Z}_{\geq 0})$. $\mathbb{Z}$. A set-valued mapping $\mathcal{M} : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ is *outer semi-continuous (OSC)* at $x \in \mathbb{R}^m$ if for all sequences $x_i \rightarrow x$ and $y_i \in \mathcal{M}(x_i)$ such that $y_i \rightarrow y$ we have $y \in \mathcal{M}(x)$. A set-valued mapping $\mathcal{M} : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ is said to be *locally bounded (LB)* at $x \in \mathbb{R}^m$ if there exists a neighborhood K_x of x such that $\mathcal{M}(K_x) \subset \mathbb{R}^n$ is bounded. Given a set $\mathcal{X} \subset \mathbb{R}^m$, the mapping $\mathcal{M}$ is OSC and LB relative to $\mathcal{X}$ if the set-valued mapping from $\mathbb{R}^m$ to $\mathbb{R}^n$ defined by $\mathcal{M}(x)$ for $x \in \mathcal{X}$, and by $\emptyset$ for $x \notin \mathcal{X}$, is OSC and LB at each $x \in \mathcal{X}$. The *outer semi-continuous (OSC) hull* of $\mathcal{M}$ is the unique set-valued mapping $\mathcal{M}_2 : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ satisfying $\text{graph}(\mathcal{M}_2) = \text{cl}(\text{graph}(\mathcal{M}))$, where $\text{graph}(\mathcal{M}) := \{(x, y) \in \mathbb{R}^m \times \mathbb{R}^n : y \in \mathcal{M}(x)\}$. Given a compact set $\mathcal{A} \subset \mathbb{R}^n$ and a vector $x \in \mathbb{R}^n$, we define $|x|_{\mathcal{A}} := \min_{y \in \mathcal{A}} |x - y|$, and we use $|\cdot|$ to denote the standard Euclidean norm. Also, we denote a vector of ones by $\mathbf{1}_n \in \mathbb{R}^n$. A continuous function $\alpha : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is said to be of class $\mathcal{K}$ if it is strictly increasing and satisfies $\alpha(0) = 0$. A continuous function $\beta : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is said to be of class $\mathcal{KL}$, if (i) for each $t \in \mathbb{R}_{\geq 0}$, the mapping $\beta(\cdot, t)$ belongs to class $\mathcal{K}$; (ii) for each fixed $s \in \mathbb{R}_{\geq 0}$, the function $\beta(s, \cdot)$ is decreasing to zero as its argument increases. Further, the cardinality of a finite set is denoted as $\text{card}(\cdot)$. We denote as $\mathbb{S}^1$ the unit circle in $\mathbb{R}^2$.

B. Graph Theory:

We characterize a directed graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ (or simply a digraph) by a vertex set $\mathcal{V}$ and an edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. We denote by $v_i v_j$ an edge from v_i to v_j in $\mathbb{G}$ or equivalently, $(i, j) \in \mathcal{E}$, and we say that v_i is an in-neighbor of v_j and v_j is an out-neighbor of v_i .

Let v_i and v_j be two vertices of $\mathbb{G}$. A *walk* from v_i to v_j , denoted by w_{ij} , is a sequence $v_{i_0} v_{i_1} \cdots v_{i_m}$ (with $v_{i_0} = v_i$ and $v_{i_m} = v_j$) in which $v_{i_k} v_{i_{k+1}}$ is an edge of $\mathbb{G}$ for all $k \in \{0, 1, \dots, m-1\}$. A walk is said to be a *path* if all the vertices in the walk are pairwise distinct. A walk is said to be a *cycle* if there is no repetition of vertices in the walk other than the repetition of the starting and ending vertex. The *length* of a path/cycle/walk is defined to be the number of edges in that path/cycle/walk.

We next introduce the following definition:

Definition 2.1: Let $\mathbb{G}$ be a digraph. A vertex v_i of $\mathbb{G}$ is said to be a **root** if for any other vertex v_j , there exists a path from v_i to v_j . If $\mathbb{G}$ contains a root, then it is called a **rooted digraph**. If $\mathbb{G}$ is a rooted digraph that does not contain any cycle, then we call $\mathbb{G}$ a **rooted acyclic digraph**. $\square$

We note here that if $\mathbb{G}$ is rooted acyclic, then there is a unique root v^* . We also need the following definition:

Definition 2.2: Let $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ be rooted and acyclic and v^* be the root. The **depth of a vertex** v_i , denoted by $\text{dep}(v_i)$, is the minimum length of a path from v^* to v_i . The depth of v^* is by default 0. Further, we define the **depth of $\mathbb{G}$** as $\text{dep}(\mathbb{G}) := \max_{v_i \in \mathcal{V}} \text{dep}(v_i)$. $\square$

With the above definition, we can decompose the vertex set $\mathcal{V}$ of $\mathbb{G}$ as follows: $\mathcal{V} = \bigcup_{l=0}^{\text{dep}(\mathbb{G})} \mathcal{V}_l$ where $\mathcal{V}_l$ is comprised of all vertices of depth l .

C. Hybrid Dynamical Systems:

In this paper, we will model the clocks and the synchronization algorithms as hybrid dynamical systems (HDS) that combine continuous-time dynamics and discrete-time dynamics [18]. A HDS with state $x \in \mathbb{R}^n$ can be described by the following dynamics:

$$\dot{x} = F(x), \quad x \in C \quad (1a)$$

$$x^+ \in G(x), \quad x \in D, \quad (1b)$$

where $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a continuous function describing the continuous-time dynamics (or flows), $G : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is an outer-semicontinuous and locally bounded set-valued mapping describing the discrete-time dynamics (or jumps), $C \subset \text{dom}(F)$ is a closed set describing the points in the space where the system can flow, and $D \subset \text{dom}(G)$ is a closed set describing the points in the space where the system can jump. Under this definition of the data $\mathcal{H} := \{C, F, D, G\}$ the HDS is said to be *well-posed*. Solutions x to (1) are defined on hybrid time domains¹, i.e., x depends on a continuous-time index t , and a discrete-time index j . Solutions x with unbounded time domain are called complete.

In this paper, we will use the following stability notion to characterize the synchronization problem.

Definition 2.3: [18] Consider a hybrid system $\mathcal{H}$ with state $x \in \mathbb{R}^n$. A compact set $\mathcal{A} \subset \mathbb{R}^n$ is said to be *uniformly globally asymptotically stable (UGAS)* if there exists a $\mathcal{KL}$ function β such that all solutions of system (1) satisfy the bound

$$|x(t, j)| \leq \beta(|x(0, 0)|, t + j), \quad (2)$$

for all $(t, j) \in \text{dom}(x)$. $\square$

III. PROBLEM FORMULATION AND MAIN RESULT

In this section, we will first introduce our model of the network of resetting clocks, and a class of hybrid synchronization dynamics based on the framework of HSRA. After this, we will formally state our problem setting, as well as our main results for rooted acyclic digraphs in Theorem 3.4.

¹We refer the reader to [18, Ch. 2] for a comprehensive introduction to hybrid time domains and the notion of solutions to hybrid systems.

A. System Model

Consider a network of N clocks or agents characterized by a digraph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$. All agents have the same resetting period $T := 1/\omega$, and individual continuous-time dynamics given by:

$$\dot{\tau}_i = \omega, \quad \tau_i \in [0, 1]. \quad (3)$$

Whenever the clock of each agent i satisfies $\tau_i = 1$, agent i resets its own clock according to the update

$$\tau_i^+ = 0, \quad (4)$$

and signals its *out-neighbours* j to update their clock as follows:

$$\tau_j^+ \in \begin{cases} 0 & \tau_j \in [0, r) \\ \{0, 1\} & \tau_j = r \\ 1 & \tau_j \in (r, 1] \end{cases}, \quad (5)$$

where $r \geq 0$ is a homogenous parameter of the network that partitions the decision rule on the unit interval $[0, 1]$. Agents that are not out-neighbours of agent i keep their state constant i.e. $\tau_j^+ = \tau_j$.

Since the dynamics of the clocks combine continuous-time updates and discrete-time updates, the overall system is a hybrid dynamical system of the form (1). However, finding the data $\mathcal{H} := \{C, F, D, G\}$ such that the overall system is well-posed is not trivial. Indeed, for a HDS (1) to be well-posed in the sense of [18], we expect that the behavior of a sequence of solutions x_k with initial conditions $x_k(0, 0)$ satisfying $\lim_{k \rightarrow \infty} x_k(0, 0) = x(0, 0)$, should approach the behavior of the limiting solution x from $x(0, 0)$, where the limits should be understood in the graphical sense [18, Ch. 5]. For the clock synchronization problem, this implies that if $\{\tau_k\}_{k=1}^\infty$ is a sequence of solutions to the hybrid system $\mathcal{H}$ with initial conditions $\tau_k(0, 0)$ satisfying $0 < \tau_{1,k}(0, 0) < \tau_{2,k}(0, 0) < \dots < \tau_{N,k}(0, 0) < 1$, for all $k \in \mathbb{Z}_{\geq 0}$, and $\lim_{k \rightarrow \infty} \tau_{1,k}(0, 0) = \lim_{k \rightarrow \infty} \tau_{2,k}(0, 0) = \dots = \lim_{k \rightarrow \infty} \tau_{N,k}(0, 0) = 1$, which generates solutions with sequential jumps with smaller and smaller times between jumps, then the limiting behavior of the solution τ from $\tau(0, 0) = [\tau_1(0, 0), \tau_2(0, 0), \dots, \tau_N(0, 0)]^\top = \mathbf{1}_N$ should generate also sequential jumps, in this case with no time between jumps [1], [2]. This implies that whenever two or more agents satisfy the condition $\tau_i = 1$, a well-posed model of the hybrid synchronization mechanism should generate non-unique solutions, since the system should capture all the possible combinations of sequential jumps that could emerge from this point.

Remark 3.1: The power of establishing desirable stability and convergence properties for well-posed HDS resides in the fact that well-posed HDS retain their stability properties (in a semi-global practical sense) under small additive bounded disturbances acting on the states and dynamics [18, Thm. 7.21]. Therefore, by designing stable clock synchronization mechanisms modeled by well-posed HDS, we are also inherently guaranteeing that the synchronization will be robust with respect to small bounded disturbances unavoidable in practical applications. $\square$

In order to obtain a well-posed hybrid model for the synchronization algorithm, for each $r \in [0, 1]$ let $G_r^0 : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be a set-valued map that is nonempty only when $i = 1$ for some $i \in \{1, \dots, N\}$ and $j \in [0, 1]$ for $j \neq i$. This mapping is defined as

$$G_r^0(\tau) := \{g \in \mathbb{R}^N : g_i = 0, g_j \in R_j(\tau), \forall j \neq i\},$$

where the set-valued map $R_j(\tau)$ is given by

$$R_j(\tau) \in \begin{cases} 0 & \tau_j \in [0, r), (i, j) \in \mathcal{E} \\ \{0, 1\} & \tau_j = r, (i, j) \in \mathcal{E} \\ 1 & \tau_j \in (r, 1], (i, j) \in \mathcal{E} \\ \tau_j & (i, j) \notin \mathcal{E}. \end{cases}$$

Note that G_r^0 captures the updates described by equations (4) and (5) which occur whenever agent i satisfies $\tau_i = 1$. To capture all the possible jumps that could emerge in the network whenever more than one agent satisfies this condition, we define the overall jump map $G_r : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ to be the outer semi-continuous hull of the mapping $G_r^0(\tau)$ [19, pp. 154-155]. Using this definition and combining all N clocks into the vector τ , we obtain a HDS with state $\tau \in [0, 1]^N \subset \mathbb{R}^n$ and hybrid dynamics

$$\dot{\tau} = F(\tau) = \omega \mathbf{1}_N, \quad C_\tau = [0, 1]^N, \quad (6a)$$

$$\tau^+ \in G_r(\tau), \quad D_\tau = \left\{ \tau \in C_\tau : \max_i \tau_i = 1 \right\}. \quad (6b)$$

This HDS is well-posed and describes the HSRA [1], [2]. The robust global clock synchronization problem can then be cast as studying the asymptotic stability properties of system (6) with respect to the following compact set:

$$\mathcal{A}_\tau := ([0, 1] \cdot \mathbf{1}_N) \cup \{0, 1\}^N. \quad (7)$$

In particular, we are interested in settings where the HDS (6) renders the set $\mathcal{A}_\tau$ UGAS, and does not generate purely or eventually discrete solutions that never flow according to (3). In such settings, the following Lemma will immediately provide robustness guarantees for the synchronization dynamics.

Lemma 3.1: Let $r \in [0, 1]$ and suppose that the HDS (6) renders the set $\mathcal{A}_\tau$ UGAS. Then, for each $\delta > 0$ there exists $\varepsilon^* > 0$ and $T^* > 0$ such that for all measurable signals $e : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^N$ satisfying $\sup_{t \geq 0} |e(t)| \leq \varepsilon^*$, all solutions τ of the perturbed system

$$\dot{\tau} = F(\tau + e) + e, \quad \tau + e \in C_\tau \quad (8a)$$

$$\tau^+ \in G_r(\tau + e) + e, \quad \tau + e \in D_\tau \quad (8b)$$

satisfy $\tau(t, j) \in \mathcal{A}_\tau + \delta \mathbb{B}$ for all $(t, j) \in \text{dom}(\tau)$ such that $t + j \geq T^*$. $\square$

We note that the perturbation e acting on the states and dynamics in (8) does not have to be the same, and it can model noisy or corrupted measurements, as well as adversarial signals bounded in norm by ε^* . The synchronization algorithm realized from the HDS (6) is summarized as **Algorithm 1**.

Algorithm 1 Distributed Clock Synchronization

```

1: procedure SYNCHRONIZATION
2:   Inputs:  $r$  and  $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ ,  $\mathcal{E} \leftarrow \text{Edge Set}$ 
3:    $\{\tau_i(0, 0)\}_{i=1}^N \leftarrow \text{Initial phase of } N\text{-Clocks}$ 
4:   Each agent  $i \in \mathcal{V}$  receives information from all its
     in-neighbours  $J$  and does the following:
5:   while  $\tau_i \in [0, 1]$  do
6:     if  $\tau_i = 1$  then  $\tau_i^+ = 0$ 
7:     if  $\max_{j \in J} \tau_j = 1$  then
8:       if  $0 \leq \tau_i < r$  then  $\tau_i^+ = 0$ 
9:       if  $\tau_i > r$  then  $\tau_i^+ = 1$ .
10:      if  $\tau_i = r$  then  $\tau_i^+ \in \{0, 1\}$ .
11:       $\dot{\tau}_i = \omega$ 

```

B. Problem Formulation

Given a homogeneous parameter $r \in [0, 1]$ in (5), we denote as $\mathcal{G}(r)$ the collection of digraphs for which the HDS (6) satisfies the following two properties: 1) It renders UGAS the compact set (7). 2) It does not generate solutions that are purely or eventually discrete ([18], Def. 2.5), i.e., solutions without intervals of flow. We aim to characterize such a set $\mathcal{G}(r)$ for any $r \in [0, 1]$. Since the HSRA guarantees that whenever synchronization happens it occurs in finite time, we also aim to compute the convergence time for any given graph $\mathbb{G} \in \mathcal{G}(r)$, i.e., we compute an upper bound of the time for the entire networked system to reach synchronization starting with any initial condition. We denote such an upper bound by $T_s(\mathbb{G}, r)$ (or simply T_s if there is no ambiguity).

Before stating the main result of the paper (Theorem 3.4), we first introduce a known positive result and a new negative result. We start with the following fact established in [1]:

Lemma 3.2: If $\mathbb{G}$ is strongly connected then $\mathbb{G} \in \mathcal{G}(r)$ for any $r \in (0, 1/N)$, where N is the number of agents in the network. $\square$

We next have the following result:

Lemma 3.3: If a digraph $\mathbb{G}$ is not rooted, then $\mathbb{G} \notin \mathcal{G}(r)$ for any $r \in [0, 1]$. $\square$

Proof: We apply the strongly connected component decomposition [9] to the digraph $\mathbb{G}$ (see [9, Sec. II-A] for detail). Since $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ is not rooted, there exist at least two (disjoint) strongly connected components of $\mathbb{G}$, denoted by $\mathbb{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathbb{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$, such that either $\mathbb{G}_1$ or $\mathbb{G}_2$ does not have any incoming neighbor. More precisely, for a given $i = 1, 2$, if v is any vertex in $\mathcal{V}_i$ and v' is any vertex in $\mathcal{V} \setminus \mathcal{V}_i$, then there does not exist a path from v' to v . Now, let the initial condition of any vertex in $\mathbb{G}_1$ (resp. $\mathbb{G}_2$) be given by $\tau(0, 0) := 0$ (resp. $\tau(0, 0) := 1/2$). Because the initial conditions of all vertices in $\mathbb{G}_i$, for $i = 1, 2$, are the same, and because each $\mathbb{G}_i$ does not have any incoming neighbor, the dynamics of any vertex in any $\mathbb{G}_i$ flows as if the vertex is isolated from the others. But then, vertices of $\mathbb{G}_1$ will never be synchronized with the vertices of $\mathbb{G}_2$. $\blacksquare$

We note that the converse of Lemma 3.3 is not true i.e. a rooted digraph $\mathbb{G}$ does not necessarily belong to $\mathcal{G}(r)$ for any $r \in [0, 1]$. Here is an example:

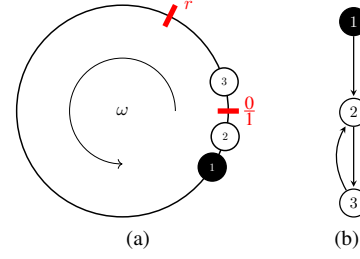


Fig. 1: Three resetting clocks with initial condition, $(\tau_1, \tau_2, \tau_3)(0, 0) = ([r, 1], [r, 1], (0, r))$ and $r \in (0, 1)$.

Example 3.1: Consider the HDS in Figure 1, where the dynamics of the agents are given by (6). Due to the periodic nature of the problem, agents are characterized on a unit circle $\mathbb{S}^1$ rotating counterclockwise at a frequency ω with 0 and 1 identified as the same point. First, note that if $r = 0$, then as soon as τ_2 hits one there exists a solution that enters the jump set and never leaves it, with τ_2 and τ_3 switching between 0 and 1. Thus, in this case, the network cannot reach synchronization. We next consider the case where $r \in (0, 1)$. We let the initial positions of the three vertices be such that $\tau_1(0, 0), \tau_2(0, 0) \in [r, 1]$ and $\tau_3(0, 0) \in (0, r)$. Then, the following events will occur subsequently:

- 1) τ_2 hits 1 and $\tau_3 \in [0, r)$, so τ_2 and τ_3 jump to zero.
- 2) τ_1 hits 1 and $\tau_2 \in [0, r)$, so τ_1 and τ_2 jump to zero.
- 3) The clocks flow for less than T seconds until τ_3 hits 1 and $\tau_2 \in [r, 1)$, so τ_2 and τ_3 jump to one.

Since all the clocks have the same resetting frequencies, from this point forward events 2) and 3) will repeat and τ_2 will oscillate between the positions of τ_1 and τ_3 . Hence, τ_1 , τ_2 and τ_3 cannot achieve synchronization. $\square$

A key feature that prevents the hybrid system (6) in the above example to render UGAS the compact set (7) is that vertices other than the root vertex form a strongly connected component. The existence of the counter-example indicates that the problem we posed at the beginning of the section is nontrivial in a sense that one cannot simply reach the conclusion that $\mathcal{G}(r)$ is the class of rooted digraphs as a straightforward extension of Lemma 3.2.

C. Main result and sketch of proof

We now take in the paper the first step to characterize $\mathcal{G}(r)$: We focus on a relatively simple class of rooted digraphs, namely, the class of rooted acyclic digraphs. We will now state the main result of the paper:

Theorem 3.4: The following statements hold:

- 1) If $\mathbb{G}$ is a rooted acyclic digraph, then $\mathbb{G} \in \mathcal{G}(r)$ for any $r \in [0, 1]$. Moreover, for any $r \in (0, 1)$, the synchronization time of $\mathcal{G}(r)$ is upper bounded by $T_s = (\text{dep}(\mathbb{G}) + 1)T$.
- 2) If $r = 0$, then $\mathcal{G}(0)$ is exactly the class of rooted acyclic digraphs. Moreover, the convergence time of $\mathcal{G}(0)$ is upper bounded by $T_s = T$. $\square$

Due to space limitations, the complete proof of Theorem 3.4 is omitted and can be found in [20].

We provide below an example to illustrate global synchronization of resetting clocks over rooted acyclic digraphs.

Example 3.2: Consider Figure 2 where there are four agents in the network and the depth of the underlying rooted acyclic graph is two. The set of all agents is further partitioned into three disjoint subsets based on their depths as follows: $\mathcal{V}_0 = \{\tau_1\}$, $\mathcal{V}_1 = \{\tau_2, \tau_3\}$ and $\mathcal{V}_2 = \{\tau_4\}$. Now following Algorithm 1, as $\mathcal{V}_0$ hits 1, it jumps to 0 and a subset of its out-neighbour agents $\mathcal{V}_1$ jumps to 0 while the other jumps to 1. Indeed, after this first step, the agents $\mathcal{V}_0$ and $\mathcal{V}_1$ are synchronized together and they will remain synchronized since all agents flow with the same frequency. Next, the agent $\mathcal{V}_2$ hits 1 and jumps to zero without affecting any other agent in $\mathcal{V}_0$ or $\mathcal{V}_1$. Finally, the agents $\mathcal{V}_0 \cup \mathcal{V}_1$, which are already synchronized, will hit 1 and eventually trigger the event that agent $\mathcal{V}_2$ jumps to either 0 or 1. Thus, all four agents flow and jump in a synchronized manner. $\square$

An outline of the proof is given below:

Sketch of Proof of Theorem 3.4: As in [1], [2], we follow a Lyapunov-based approach for hybrid systems, and we invoke the Hybrid Invariance Principle [21]. The proof consists of three main arguments:

- 1) We define the Lyapunov function $V_\tau : [0, 1]^N \rightarrow \mathbb{R}_{\geq 0}$ such that V_τ is the infimum of the lengths of all arcs touching all agents τ_i , where the points 0 and 1 on the interval $[0, 1]$ are identified to be the same, to form a circle $\mathbb{S}^1$. Such Lyapunov function is positive definite with respect to the set $\mathcal{A}_\tau$.
- 2) We show that the Lyapunov function remains constant during the flows and does not increase during the jumps.
- 3) We build our argument on the fact that the root agent is not influenced by any other agent in the rooted acyclic digraph. Then, we prove by induction the following fact: When an agent of depth k in the graph reaches $\tau_i = 1$, a condition that is guaranteed to happen in rooted acyclic digraphs, it will force the agents on the next level, i.e., agents of depth $(k+1)$, to be synchronized, this is also illustrated in Example 3.2. These jumps will necessarily decrease the Lyapunov function unless all the agents in depths k and $k+1$ are already synchronized. By moving forward on k we will eventually exhaust the depths of the graph, implying that the Lyapunov function $V(z)$ converges to zero in finite time, by construction this implies finite time synchronization.

We then appeal to the Hybrid Invariance Principle (see [21]) to establish that system (6) indeed renders UGAS the compact set (7). For an arbitrary initial condition, the worst-case synchronization time corresponds to the case when every agent at each depth k jumps to zero. Hence, by repeatedly applying the above arguments (starting with $k = 0$), we will show that the entire networked system will be synchronized in at most $(\text{dep}(\mathbb{G}) + 1)T$ time, where T was defined as the natural resetting period of the agents.

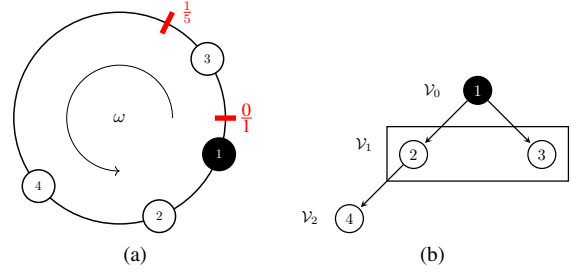


Fig. 2: Example 3.2: Synchronization of resetting clocks over rooted acyclic digraph

IV. SIMULATION RESULTS

Consider the HDS (6) for five resetting clocks. The tunable parameter is chosen as $r = 0.2$ for all agents, and the frequency of the clocks is selected as $\omega = 1$. We implement this system with an underlying rooted acyclic graph. We provide three examples of such graphs as shown in Figure 3. For each example, we validate Theorem 3.4 by showing that: (i) the hybrid system (6) synchronizes, (ii) the convergence time is related to the depth of the graph, and (iii) no solution remains in the jump set for ever, i.e., all solutions eventually have to flow.

For digraphs (a) and (b) in Figure 3, we used the same initial conditions to implement Algorithm 1. The simulation results over these networks are then illustrated in Figure 4. Observe that for the network topology (a), as soon as τ_2 hits 1, it synchronizes with its out-neighbors τ_3 and τ_5 , after which τ_3 forces τ_4 to synchronize. However, τ_1 remains unaffected. Soon after that, τ_1 hits 1 and synchronizes $\tau_2, \tau_3, \tau_4, \tau_5$ with itself in $t = 1$ second. From this point forward, all the clocks remain synchronized and there is no solution that stays in the jump set forever. Note that in this case, the agents synchronize in 1 second which is less than $T_s = 3$ seconds. Similarly, for the network topology (b), as soon as τ_2 hits 1, it synchronizes τ_5 with itself. Soon after that τ_3 hits 1 and τ_3, τ_4 synchronize together. Finally, as τ_1 hits 1, it synchronizes all the agents in the network. Again, note that all the clocks still synchronize in 1 second which is less than $T_s = 3$ seconds. Observe that even though the network synchronization behavior of case (a) is different from case (b), the overall network of clocks converges to the set $\mathcal{A}$ in no more than 1 second. This is due to the fact that in both graphs the number of edges are different but the depths are the same.

Now, consider the network topology (c) as shown in Figure 3. We set the initial conditions of each clock very close to zero and we apply Algorithm 1 to obtain the plots shown in Figure 5. For this case, observe that when τ_1 hits 1, it synchronizes τ_2 with itself. Then τ_1, τ_2 both hit 1 and synchronize τ_3 with them and so on. When $r = 0.2$ the network converges in 3.6 seconds which is less than the theoretical worst-case synchronization time of $T_s = 5$ seconds. Similarly, if we homogeneously set the value of $r = 0$, the network will synchronize in 0.9 seconds which is less than the theoretical worst-case convergence time of

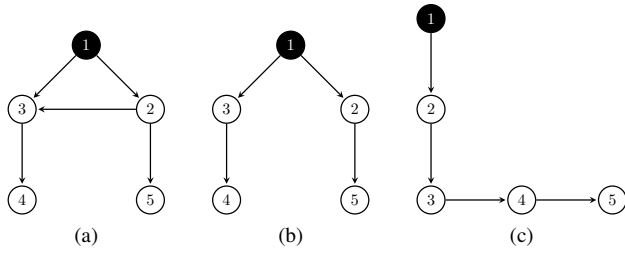


Fig. 3: Rooted Acyclic Graphs with depths a) two b) two c) four. (Black indicates the root vertex)

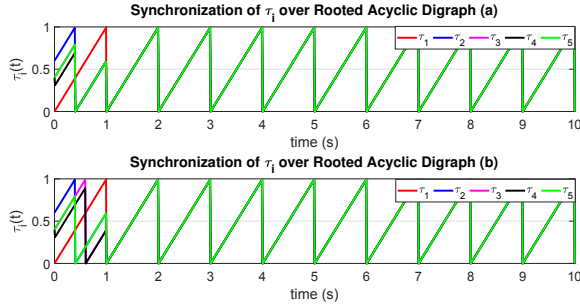


Fig. 4: Resetting Clocks Synchronization over graphs (a)-(b) in Figure 3.

$T_s = 1$ second. Hence, the simulation results validate the following aspects: (i) For rooted acyclic graphs, the hybrid system (6) achieves synchronization, (ii) the convergence time is related to the depth of the digraph by Theorem 3.4, and (iii) the hybrid dynamics generate no purely or eventually discrete-time solutions, i.e., complete solutions that remain in the jump set.

V. CONCLUSIONS

In this paper, we have introduced the set $\mathcal{G}(r)$, defined as the collection of the *information flow topologies* that can render robust global synchronization of the resulting networks of resetting clocks. While a complete characterization of the set still remains open, we established in the paper the following facts: (1) Rooted acyclic digraphs always belong to $\mathcal{G}(r)$ for any $r \in [0, 1)$, and (2) If $r = 0$, then $\mathcal{G}(r)$ is exactly the class of rooted acyclic digraphs. We also computed an upper bound on the synchronization time, which relates the convergence time to the depth of the underlying rooted acyclic digraph. Finally, we have provided numerical results that illustrate the main points of this paper. Future extensions will explore the role of random graphs in the synchronization properties of the synchronization dynamics.

REFERENCES

- [1] A. R. Teel and J. I. Poveda, "A hybrid systems approach to global synchronization and coordination of multi-agent sampled-data systems," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 123 – 128, 2015. Analysis and Design of Hybrid Systems ADHS.
- [2] J. I. Poveda and A. R. Teel, "Hybrid mechanisms for robust synchronization and coordination of multi-agent networked sampled-data systems," *Automatica*, vol. 99, pp. 41–53, 2019.

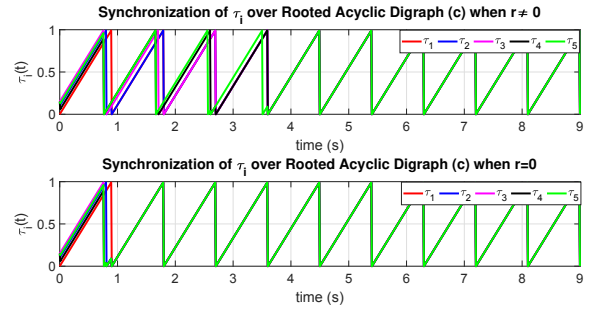


Fig. 5: Resetting Clocks Synchronization over graph (c) in Figure 3.

- [3] D. Nesic and D. Liberzon, "A unified framework for design and analysis of networked and quantized control systems," *IEEE Transactions on Automatic Control*, vol. 54, pp. 732–747, April 2009.
- [4] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 57, pp. 1291–1297, May 2012.
- [5] C. De Persis and P. Frasca, "Robust self-triggered coordination with ternary controllers," *IEEE Transactions on Automatic Control*, vol. 58, pp. 3024–3038, Dec 2013.
- [6] C. De Persis and R. Postoyan, "A Lyapunov redesign of coordination algorithms for cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 62, Feb 2017.
- [7] C. Lenzen, T. Locher, P. Sommer, and R. Wattenhofer, "Clock synchronization: Open problems in theory and practice," *In Proc. of International Conference on Current Trends in Theory and Practice of Computer Science*, pp. 61–70, 2010.
- [8] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
- [9] X. Chen, M.-A. Belabbas, and T. Başar, "Controllability of formations over directed time-varying graphs," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 3, pp. 407–416, 2017.
- [10] X. Chen, "Controllability of continuum ensemble of formation systems over directed graphs," *Automatica*, vol. 108, 2019. appeared online.
- [11] A. Sarlette, S. Tuna, V. Blondel, and R. Sepulchre, "Global synchronization on the circle," *In Proc. of 17th IFAC World Congress*, pp. 9045–9050, 2008.
- [12] M. Hartman, A. Subbaraman, and A. R. Teel, "Robust global almost sure synchronization on a circle via stochastic hybrid control," in *Control of Cyber-Physical Systems* (D. C. Tarraf, ed.), vol. 449 of *Lecture Notes in Control and Information Sciences*, pp. 3–21, Springer, 2013.
- [13] A. V. Proskurnikov and M. Cao, "Synchronization of pulse-coupled oscillators and clocks under minimal connectivity assumptions," *IEEE Transactions on Automatic and Control*, DOI 10.1109/TAC.2016.2644538.
- [14] E. D. Sontag, "Clocks and insensitivity to small measurement errors," *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 4, no. 4, pp. 537–557, 1999.
- [15] E. Sontag, "Stability and stabilization: Discontinuities and the effect of disturbances," in *Nonlinear Analysis, Differential Equations and Control*, vol. 528 of *NATO Science Series*, pp. 551–598, 1999.
- [16] F. Nunez, Y. Wang, and F. J. D. III, "Global synchronization of pulse-coupled oscillators interacting on cycle graphs," *Automatica*, vol. 52, pp. 202–209, 02015.
- [17] F. Nunez, Y. Wang, A. Teel, and F. J. D. III, "Synchronization of pulse-coupled oscillators to a global pacemaker," *Systems & Control Letters*, vol. 88, pp. 75–80, 2016.
- [18] R. Goebel, R. Sanfelice, and A. R. Teel, *Hybrid Dynamical System*. Princeton, NJ: Princeton University Press, 2012.
- [19] R. T. Rockafellar and R. J. Wets, *Variational Analysis*. Springer, 2009.
- [20] M. U. Javed, J. I. Poveda, and X. Chen, "Global Synchronization of Clocks in Directed Rooted Acyclic Graphs: A Hybrid Systems Approach." <https://arxiv.org/pdf/1904.03566.pdf>, 2019.
- [21] R. G. Sanfelice, R. Goebel, and A. R. Teel, "Invariance principles for hybrid systems with connections to detectability and asymptotic stability," *IEEE Trans. Autom. Control*, vol. 52, pp. 2282–2297, 2007.