Phase Desynchronization: A New Approach and Theory Using Pulse-Based Interaction

Timothy Anglea, *Student Member, IEEE*, and Yongqiang Wang, *Senior Member, IEEE*

***Abstract*—In this paper, we present a novel approach for achieving phase desynchronization in a pulse-coupled oscillator network. Ensuring phase desynchronization is a difficult problem, and existing results are constrained to a completely interconnected network and a fixed number of nodes. Our approach is more robust than previous approaches, removing the constraint of a fixed number of nodes. The removal of this constraint is significant because it allows the network to receive and drop nodes freely without any change to the phase update strategy. Also, to our knowledge, our approach is the first to prove the convergence to the desynchronized state for a topology that is more general than the all-to-all topology. More specifically, our approach is applicable to any circulant, symmetric network topology, including the circulant symmetric ring topology. Rigorous mathematical proofs are provided to support the result that any circulant, symmetric network with ordered phases under our proposed algorithm will converge to uniform phase desynchronization. Simulation results are also presented to demonstrate the algorithm's performance.**

***Index Terms*—Pulse-coupled oscillator, desynchronization, phase desynchronization, splay state.**

I. Introduction

THE study and analysis of pulse coupled oscillators (PCOs) is a currently active field of engineering research. Initially proposed to model the synchronization of many biological systems, such as the flashing of fireflies [1], [2] and firing neurons [3], [4], the model has been expanded to encompass more complex behaviors [5], [6]. One such behavior is desynchronization, a more difficult problem and the inverse problem to synchronization. Desynchronization has many important applications, primarily in the area of decentralized network communication [7]–[9], but also in the design of analog-to-digital converters and the control of traffic intersection flow [7]. Compared with the increased research on achieving phase synchronization [10]–[14], there is much less research on desynchronization.

In this paper we propose a desynchronization strategy and algorithm that we believe to be unique to the literature for desynchronizing pulse coupled oscillator (PCO) networks. A PCO network is one of connected oscillators, or nodes, each with an associated phase that continuously cycles from 0 to 2π. A node fires a pulse to all of its neighboring nodes when it reaches a threshold value, often 2π. Based upon the timing at which these pulses are received and fired, the PCOs can respond in a desirable way. The great benefits of a PCO network are that the communication between nodes is a single, identical pulse, minimizing latency and data corruption, and that the control of the network can be decentralized.

Many algorithms and approaches have been proposed to achieve desynchronization. Some control strategies ensure weak desynchronization, where the timing between firings is constant, but the phases are constantly shifting [7], [15]. Other control strategies ensure strong, or phase, desynchronization, but must assume a completely interconnected, or all-to-all, network and a fixed number of nodes [7]–[9], [16], [17]. Some have tried to relax the constraint of an all-to-all connection topology, but instead require either a continuously coupled, rather than pulse coupled, network [6], [18], or the use of additional types of nodes and pulse transmissions [19]. Work has also been done to improve the speed of convergence assuming an all-to-all topological network [20], [21]. In this paper, we propose an approach that relaxes the constraint of complete interconnection and removes the requirement of a fixed number of nodes to achieve complete phase desynchronization. To the best of our knowledge, our approach is the first to prove that a network topology other than the all-to-all topology can be used to achieve uniform phase desynchronization with identical PCOs.

In Section II, we will describe our proposed algorithm to achieve desynchronization for circulant symmetric networks with ordered nodes. We note that the algorithm is independent of the number of nodes in the network, resulting in a more robust control, allowing for nodes to join and leave the network freely without affecting the update rule. In Section III, we will provide rigorous proof for guaranteed desynchronization under the all-to-all topology using our algorithm. In Section IV, we will relax the constraint of the all-to-all topology, showing that any circulant symmetric network topology, and specifically the circulant symmetric ring topology, will desynchronize using the proposed algorithm. In Section V, we will provide simulation results showing how the network desynchronizes under our proposed algorithm and analyze the convergence rate of our algorithm. In Section VI, we will conclude and present areas of further research.

II. Network Dynamics

It is our belief that the algorithm used in this paper is unique in the literature on PCOs. The goal of desynchronization is to spread the phases of all oscillators, or nodes, evenly around the unit circle. This state of desynchronization is sometimes termed

Manuscript received June 11, 2016; revised September 15, 2016 and October 21, 2016; accepted November 10, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Subhrakanti Dey. The work was supported in part by the U.S. Army Engineer Research & Development Center's Basic Research Program 455337 and the Institute for Collaborative Biotechnologies through grant W911NF-09-0001.

The authors are with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634 USA (e-mail: tbangle@clemson.edu; yongqiw@clemson.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2016.2633246

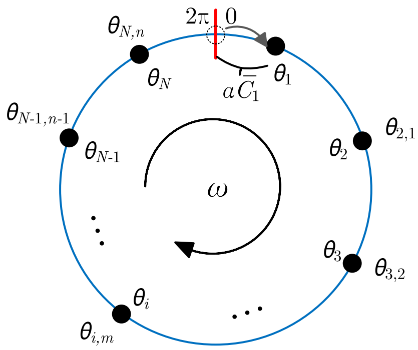


Fig. 1. A snapshot of current and recorded phases when a node fires for a network of N pulse coupled oscillators with fundamental frequency ω. When a node fires, all connected nodes record the m^{th} instance of their current phase. For example, when node 1 fires, node 2 records $\theta_{2,1}$ and node 3 records $\theta_{3,2}$. Node 3 recorded $\theta_{3,1}$ when node 2 fired previously.

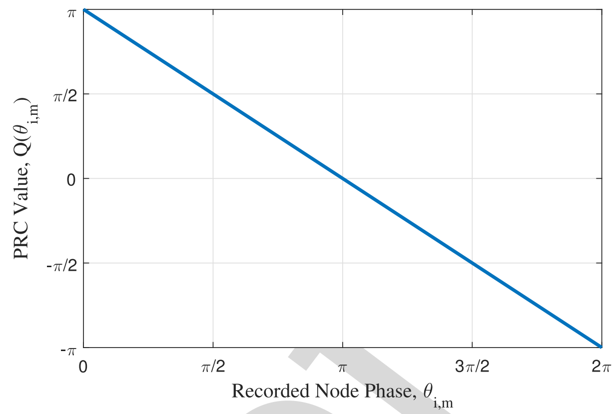


Fig. 2. Phase Response Curve, Q, as given in (1) for the proposed algorithm.

the splay state. Inspiration for the proposed algorithm is based on the DESYNC and DESYNC-STALE algorithms proposed in [7]. However, different from [7], which only allows nodes to react based on the nodes directly ahead of and behind it in phase, our approach allows nodes to react to all nodes with which it is connected. A mechanical analogy can be made, where each node, spread around the unit circle, is connected to other nodes by compressed springs [12]. The closer the nodes are in phase, the more force the nodes will exert in moving the other away. An equilibrium is achieved when all of the nodes are as far away from each other in phase as possible, the desired state of desynchronization.

A. General Algorithm Definitions

Let us assume a network of N pulse coupled oscillators, as shown in Fig. 1. Let $\theta_i \in [0, 2\pi]$ be the associated phase of oscillator, or node, $i \in \mathcal{V} = \{1, 2, \cdots, N\}$, with ω as the fundamental frequency for all oscillators. The state of the network $\vec{\theta} = [\theta_i]$ is a column vector representing the phase of the N nodes.

We define $\mathcal{A}_{\mathcal{R}} = [a_{ij}] \in \mathbb{R}^{N \times N}$ to be the adjacency matrix for the network, where element a_{ij} is 1 if there is a connection from node i to node j, and 0 if there is no connection. The adjacency matrix defines the topology for the network. $\mathcal{A}_{\mathcal{R}}$ is zero diagonal ($a_{ii} = 0\ \forall i \in \mathcal{V}$), and is symmetrical if connections between nodes are bidirectional. For each node i, we let $\mathcal{N}_i$ denote the set of nodes connected to node i, where $\mathcal{N}_i \subset \mathcal{V}$, and $|\mathcal{N}_i| = n_i$.

Our proposed algorithm can be described as follows.

1) A node fires a pulse when its phase reaches 2π. When the node fires, all connected nodes receive the pulse.
2) When a node receives a pulse, the node records the value of *its own* current phase, $\theta_{i,m}$. This value is stored to be used when the node fires, i.e., when its phase reaches 2π.
3) When a node fires, emits a pulse, and resets its phase to zero, it jumps, and the amount that the node jumps is dependent upon all of the recorded phase values due to the firing of the other nodes. The node computes the phase response curve (PRC) value, $Q(\theta_{i,m})$, for each of its own recorded phase values, and then computes the change value, C_i, or the amount that the node needs to jump, from the PRC values. If the change value is positive, the node will jump by a proportional amount, αC_i, where α is a positive real number; otherwise, the node will not jump. Thus, the node will only jump forward in phase.

The phase update for the network is asynchronous. Each node updates its phase when it fires, and does not wait for other nodes. However, the amount that each node jumps is determined by the recorded phase values found from the previous firing cycle. In other words, the network relies on "stale" information obtained from the timing of received pulses, as in the DESYNC-STALE algorithm in [7]. In this paper, the invariance of the firing sequence of the nodes is assumed, which will be proven later.

Let us denote the stroboscopic recorded phase values when a node receives a pulse as $\theta_{i,m}$, where i is the node that received the pulse, and $m \in \{1, 2, \ldots, n_i\}$ is the index of the recorded phase value of that node due to the pulses received from neighboring nodes in $\mathcal{N}_i$. Thus, $0 < \theta_{i,1} < \theta_{i,2} < \ldots < \theta_{i,n_i} < 2\pi$.

We define the PRC of the network, Q, in the following way:

$$Q(\theta_{i,m}) = -(\theta_{i,m}) + \pi \tag{1}$$

An interpretation for the PRC function in Fig. 2 is that if two nodes are close in phase, they will exert a force on each other that causes them to move farther away from each other. Otherwise, when the nodes are as far apart as possible ($\theta_{i,m} = \pi$), they will exert no force on each other.

B. Pulse Dynamics

The amount that a node jumps is dependent on the relative position and total number of other nodes from which it has received pulses. For our algorithm, the node will compute the average amount of the calculated PRC values from its recorded phase values. Let us, then, define the change value, C_i, to be this average amount:

$$C_i = \frac{1}{n_i} \sum_{m=1}^{n_i} Q(\theta_{i,m}) \tag{2}$$

As mentioned earlier, when a node fires, it computes its change value and then jumps by a portion of that amount if the change value is positive. Otherwise, the node does not jump. This jump rule is described mathematically as

$$\theta_i^+ = \begin{cases} \alpha C_i & \text{if } C_i > 0 \\ 0 & \text{if } C_i \leq 0 \end{cases} \tag{3}$$

To summarize, a node will jump in phase only when it fires, and then only if it will jump forward in phase.

Remark 1: In application, the phase could be allowed to jump backward, allowing for faster convergence. However, there will exist the problem of having the state, in theory, be less than zero for a period of time. Care will need to be taken to ensure the proper behavior of the network and to maintain the proper firing sequence.

Let us define the change vector, C, as the column vector of the phase change values corresponding to all nodes in the network:

$$C = [C_i] \in \mathbb{R}^N \tag{4}$$

C. Network Equilibrium

In order for us to show that the network will desynchronize, we need to show that the splay state is the only equilibrium point for the network under our proposed algorithm. For this paper we will show that the splay state is the equilibrium point for any circulant symmetric network with ordered nodes.

Definition 1: Nodes in a PCO network are considered ordered when they are initially numbered in order by their phase magnitude, i.e., $0 \leq \theta_1 < \theta_2 < \ldots < \theta_{N-1} < \theta_N \leq 2\pi$.

Definition 2: A network with ordered nodes is called circulant symmetric if the resulting adjacency matrix $\mathcal{A}_{\mathcal{R}}$ for the ordered nodes is both circulant and symmetric.

Remark 2: For our paper, a circulant symmetric network assumes ordered nodes.

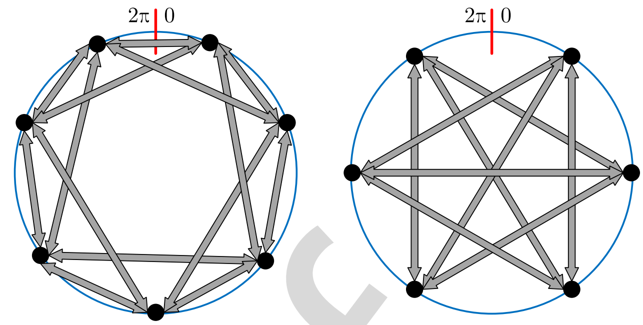
Remark 3: For circulant symmetric networks, all nodes have the same number of neighbors, or $\mathcal{N}_i = n \; \forall i \in \mathcal{V}$. For the all-to-all network topology, $n = N - 1$, and for the ring topology, $n = 2$.

Proposition 1: For any circulant symmetric network under the proposed algorithm, the network is desynchronized if and only if the change vector in (4) is the zero vector.

Proof: When the network is in the state of desynchronization, all nodes are as far apart from each other in phase as possible. In this state, the phases are equally spaced around the region $[0, 2\pi]$. Since the network is circulant and symmetric, for each node in the network, there will be equal force pushing the node forward in phase as there is pushing it backward in phase. Thus, over a cycle of firings, C_i will be zero for all nodes, and no node will jump in phase.

If the network is not desynchronized, then since the network is circulant and symmetric, there will be an imbalance in the network, causing the change vector to be non-zero. ■

Proposition 1 indicates that the desynchronized state is the only equilibrium state under our setup. This result gives us insight into how to determine when the network has reached desynchronization.



(a) N = 7 (b) N = 6

Fig. 3. Examples of circulant symmetric networks that will achieve desynchronization under the proposed algorithm. (a) Seven node network with each node connected to two nodes ahead in phase and two nodes behind in phase. (b) Six node network where each node is without connections to the nodes directly ahead and behind it in phase.

D. Network Energy Measure

Let us define an energy function for the network using the change vector after the k^{th} firing cycle:

$$p(k) = \frac{1}{2} C(k)^T C(k) = \frac{1}{2} \sum_{i=1}^{N} C_i(k)^2 \tag{5}$$

By the definition, the energy is non-negative everywhere, and zero only when the change vector from (4) is the zero vector. Thus, (5) is a potential "Lyapunov" function. The energy is computed after every firing cycle, when all nodes have fired once in the sequence.

We also need to define the change in the energy as a function of the cycle:

$$\Delta p(k) = p(k+1) - p(k) \tag{6}$$

Proposition 2: For any circulant symmetric network under the proposed algorithm, the network will desynchronize if $\Delta p(k) < 0$ for any k when $p(k) > 0$.

Using LaSalle's theorem given in [22], Proposition 2 follows easily from the defined energy function and Proposition 1.

In the next sections, we will prove that the condition in Proposition 2 holds for the all-to-all topology and circulant symmetric topologies in general, specifically for the circulant symmetric ring topology.

Remark 4: The all-to-all and circulant symmetric ring topologies are special cases of circulant symmetric networks. Networks with adjacency matrices that are both symmetric and circulant are a subset of highly symmetrical networks, as described in [12].

III. All-to-All Case

A. Preliminaries

In order for the network to desynchronize, we need to ensure that the firing sequence of the nodes remains unchanged between cycles. If nodes switch order after a cycle, then the resulting network with ordered nodes may no longer be circulant.

Lemma 1: The firing sequence for any circulant symmetric network is invariant between cycles under the proposed algorithm for any change ratio $\alpha \in (0, 1]$.

See Appendix for proof.

To simplify our proofs, we will utilize the following property of the change values C_i.

Lemma 2: For a circulant symmetric network under the proposed algorithm, $\sum_{i=1}^{N} C_i = 0$ holds if no nodes have jumped during the previous cycle.

See Appendix for proof.

Remark 5: For both Lemma 1 and Lemma 2, it is necessary that the network be circulant symmetric. Thus, we restrict our analysis to this class of networks.

B. Convergence of the All-to-All Topology

Let us first consider a PCO network with an all-to-all topology. This network has been shown to converge under other desynchronization algorithms [7], so we will show that the network will also converge under our proposed algorithm.

Theorem 1: A network with an all-to-all topology under the proposed algorithm will converge to the desynchronized state.

Proof: We can prove Theorem 1 if we can show that $\Delta p(k) < 0$ for a general k when $p(k)$ is non-zero.

Let us first note that the expression for the change in energy can be rewritten by combining (5) and (6) and simplifying:

$$\Delta p(k) = \frac{1}{2}\sum_{i=1}^{N}\left[C_i(k+1)^2 - C_i(k)^2\right] \tag{7}$$

By substituting the PRC in (1) into (2), we can write an expression for the change value for a node at a given cycle:

$$C_i(k) = \frac{1}{n}\sum_{m=1}^{n}\big(-\theta_{i,m}(k)+\pi\big) \tag{8}$$

Let $j \in \mathcal{N}_i$ be an element in the set of nodes that are connected to node i. For the all-to-all topology, $\mathcal{N}_i$ includes all of the network nodes except node i. The recorded phase values can be expressed as

$$\theta_{i,m}(k) = 2\pi - \big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\big), \tag{9}$$

where j corresponds to the node that resulted in the m^{th} recorded phase value. We can thus express (8) in terms of the node phases during the current cycle as follows:

$$C_i(k) = \frac{1}{n}\sum_{j\in\mathcal{N}_i}\left[\big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\big) - \pi\right] \tag{10}$$

Similarly, the change value for the same node at the next cycle can also be expressed in terms of the node phases during the next cycle:

$$\begin{aligned} C_i(k+1) &= \frac{1}{n}\sum_{m=1}^{n}\big(-\theta_{i,m}(k+1)+\pi\big)\\ &= \frac{1}{n}\sum_{j\in\mathcal{N}_i}\left[\big(\big(\theta_j(k+1)-\theta_i(k+1)\big) \mod 2\pi\big) - \pi\right] \end{aligned} \tag{11}$$

We now need an expression for the recorded phase value for the next cycle. We know that, in the next cycle, the phases may have jumped a fractional amount of C_i, which may have been non-positive, from the previous cycle following the update rule given in (3). The amount that node i jumps during the current cycle, which we will denote as $\bar{C}_i$, can be mathematically described as the maximum of zero and C_i, or

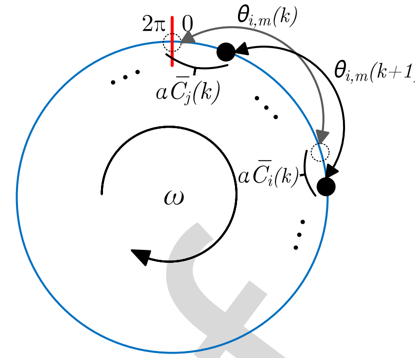


Fig. 4. Illustration for (13), showing the change in $\theta_{i,m}$ from one cycle to the next.

$$\bar{C}_i = \max\{0, C_i\} \tag{12}$$

Eqn. (12) allows us to write an expression for the next recorded phase value based on the previous recorded phase value and the amount that each node jumped in the previous cycle, as illustrated in Fig. 4.

$$\theta_{i,m}(k+1) = \theta_{i,m}(k) + \alpha\big(\bar{C}_i(k) - \bar{C}_j(k)\big) \tag{13}$$

Using the expression in (9), we can rewrite (13) as

$$\begin{aligned} &2\pi - \big(\big(\theta_j(k+1)-\theta_i(k+1)\big) \mod 2\pi\big)\\ &= 2\pi - \big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\big) + \alpha\big(\bar{C}_i(k)-\bar{C}_j(k)\big)\\ &\implies\\ &\big(\big(\theta_j(k+1)-\theta_i(k+1)\big) \mod 2\pi\big)\\ &= \big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\big) + \alpha\big(\bar{C}_j(k)-\bar{C}_i(k)\big) \end{aligned} \tag{14}$$

Thus, the expression for the next change value, by substituting (14) into (11), can be written as

$$\begin{aligned} C_i(k+1) = &\frac{1}{n}\sum_{j\in\mathcal{N}_i}\Big[\big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\big)\\ &+ \alpha\big(\bar{C}_j(k)-\bar{C}_i(k)\big) - \pi\Big]\\ = &\frac{1}{n}\sum_{j\in\mathcal{N}_i}\left[\big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\big) - \pi\right]\\ &+ \frac{\alpha}{n}\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k)-\bar{C}_i(k)\right] \end{aligned} \tag{15}$$

Recognizing the expression for the change value from (10), we can simplify (15) as follows:

$$C_i(k+1) = C_i(k) + \frac{\alpha}{n}\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k)-\bar{C}_i(k)\right] \tag{16}$$

Squaring both sides of (16) and subtracting $C_i(k)^2$ from both sides give us the following expression for the summation term in (7):

$$C_i(k+1)^2 - C_i(k)^2 = \frac{2\alpha}{n} C_i(k) \sum_{j\in\mathcal{N}_i} \left[\bar{C}_j(k) - \bar{C}_i(k)\right] + \left(\frac{\alpha}{n} \sum_{j\in\mathcal{N}_i} \left[\bar{C}_j(k) - \bar{C}_i(k)\right]\right)^2 \quad (17)$$

We can rearrange and simplify (17) as follows:

$$\begin{aligned} &C_i(k+1)^2 - C_i(k)^2 \\ &= \frac{2\alpha}{n}\left[C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) - C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_i(k)\right] \\ &\quad + \frac{\alpha^2}{n^2}\left(\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k) - \bar{C}_i(k)\right]\right)^2 \\ &= \frac{2\alpha}{n}\left[C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) - nC_i(k)\bar{C}_i(k)\right] \\ &\quad + \frac{\alpha^2}{n^2}\left(\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k)\right] - n\bar{C}_i(k)\right)^2 \end{aligned} \quad (18)$$

We can use the fact that $C_i(k)\bar{C}_i(k) = \bar{C}_i(k)^2$, and then expand and recombine (18) to achieve the following:

$$\begin{aligned} &C_i(k+1)^2 - C_i(k)^2 \\ &= \frac{2\alpha}{n}C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) - 2\alpha\bar{C}_i(k)^2 \\ &\quad + \frac{\alpha^2}{n^2}\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j(k)\right]^2 - \frac{2\alpha^2}{n}\bar{C}_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) + \alpha^2\bar{C}_i(k)^2 \\ &= (\alpha^2 - 2\alpha)\left(\bar{C}_i(k)^2\right) + \frac{\alpha^2}{n^2}\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j(k)\right]^2 \\ &\quad + \left(\frac{2\alpha}{n}C_i(k) - \frac{2\alpha^2}{n}\bar{C}_i(k)\right)\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j(k)\right] \end{aligned} \quad (19)$$

Substituting (19) into (7) gives us an expression for the change in network energy, where we drop the cycle (k) notation for clarity:

$$\begin{aligned} \Delta p &= \frac{1}{2}\sum_{i=1}^{N}\left[(\alpha^2 - 2\alpha)(\bar{C}_i^2) + \frac{\alpha^2}{n^2}\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)^2 \right. \\ &\quad \left. + \left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right] \\ &= \frac{1}{2}\left((\alpha^2 - 2\alpha)\sum_{i=1}^{N}(\bar{C}_i^2) + \frac{\alpha^2}{n^2}\sum_{i=1}^{N}\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j\right]^2 \right. \\ &\quad \left. + \sum_{i=1}^{N}\left[\left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \end{aligned} \quad (20)$$

We can further simplify the second term in (20) by expanding and recombining terms as follows:

$$\sum_{i=1}^{N}\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)^2 = n\sum_{i=1}^{N}\bar{C}_i^2 + (n-1)\left(\sum_{i=1}^{N}\bar{C}_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right) \quad (21)$$

Substituting (21) into (20) and recombining terms give the following result:

$$\begin{aligned} \Delta p &= \frac{1}{2}\left((\alpha^2 - 2\alpha)\sum_{i=1}^{N}(\bar{C}_i^2) \right. \\ &\quad + \frac{\alpha^2}{n^2}\left(n\sum_{i=1}^{N}\bar{C}_i^2 + (n-1)\left[\sum_{i=1}^{N}\bar{C}_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \\ &\quad \left. + \sum_{i=1}^{N}\left[\left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \\ &= \frac{1}{2}\left((\alpha^2 - 2\alpha)\sum_{i=1}^{N}(\bar{C}_i^2) + \frac{\alpha^2}{n}\sum_{i=1}^{N}(\bar{C}_i^2) \right. \\ &\quad + \sum_{i=1}^{N}\left[\frac{\alpha^2(n-1)}{n^2}\bar{C}_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right] \\ &\quad \left. + \sum_{i=1}^{N}\left[\left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \end{aligned} \quad (22)$$

Further recombination and simplification lead to the following final result:

$$\begin{aligned} \Delta p &= \left(\frac{n+1}{2n}\alpha^2 - \alpha\right)\sum_{i=1}^{N}(\bar{C}_i^2) \\ &\quad - \frac{1}{n}\sum_{i=1}^{N}\left[\left(\frac{n+1}{2n}\alpha^2\bar{C}_i - \alpha C_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right] \end{aligned} \quad (23)$$

It now leaves to show that (23) is always non-positive, and only stays zero when the network is desynchronized. Let us denote Δp_i as the i^{th} term of Δp, corresponding to the i^{th} change value, such that

$$\Delta p = \sum_{i=1}^{N}\Delta p_i. \quad (24)$$

Let us arrange the Δp_i terms into two groups: 1) Set $\mathcal{M}$ corresponding to when C_i is non-positive, and 2) Set $\mathcal{S}$ when it is positive.

1) If $C_i \leq 0$, then, from (12), $\bar{C}_i = 0$, and $\Delta p_i \in \mathcal{M}$ can be simplified as follows:

$$\Delta p_i = \frac{\alpha}{n}C_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right) \quad (25)$$

Eqn. (25) is non-positive for any positive α, and the sum of all of the $\Delta p_i \in \mathcal{M}$ will also be non-positive.

2) If $C_i > 0$, then, from (12), $C_i = \bar{C}_i$, and the sum of all $\Delta p_i \in \mathcal{S}$ can be simplified as follows:

$$\sum_{i\in\mathcal{S}} \Delta p_i = \Big(\frac{n+1}{2n}\alpha^2 - \alpha\Big)\sum_{i\in\mathcal{S}}\left[\bar{C}_i^2 - \frac{1}{n}\left(\sum_{j\in\mathcal{N}_i}\bar{C}_i\bar{C}_j\right)\right] \tag{26}$$

For the summation term in brackets, there will be $|\mathcal{S}| = s$ square terms and at most $\frac{sn}{n} = s$ non-zero pairwise terms, causing the expression to be strictly non-negative. Thus, (26) will be non-positive for any $\alpha \in (0, \frac{2n}{n+1})$.

Therefore, the sum of non-positive values results in the fact that (23) is always non-positive for any $\alpha \in (0, \frac{2n}{n+1})$.

We therefore must deal with the cases where Δp is zero and desynchronization has not yet been achieved. There are two such cases: 1) All change values are equal and positive ($C_1 = C_2 = \ldots = C_N > 0$), and 2) all change values are non-positive ($C_i \leq 0\ \forall i$).

1) It is trivial to show that the first case cannot happen. If all change values are positive, then $\theta_i - \theta_{i-1} < \theta_{i+1} - \theta_i < \ldots < \theta_{i-1} - \theta_{i-2} < \theta_i - \theta_{i-1}$, which is a contradiction.
2) The second case can be true for one cycle since the nodes can sense only the "stale" phase from each node. However, this case will not be true for the next cycle. If no nodes move in a cycle, each node will then be observing the true phase of its connecting nodes, and not the stale values, in the next cycle. According to Lemma 2, $\sum_{i=1}^N C_i = 0$ holds in the next cycle. If desynchronization has not yet been achieved (i.e., not all change values are zero), then at least one node will have a positive change value, and will move forward in phase, causing the change in energy to be negative according to (23).

Therefore, the energy will always decrease and converge to zero. According to Proposition 2, the network will converge to the state of desynchronization. ■

Remark 6: As determined from the derivation of (23), a choice of change ratio $\alpha \in (0, \frac{2n}{n+1})$ will ensure that the network energy will decrease with each cycle. An α larger than $\frac{2n}{n+1}$ cannot be used to guarantee that the network energy will converge to zero, and thus cause the network to desynchronize. Because we want to choose α such that it can guarantee desynchronization independent of network size, the largest range of α, then, that can ensure desynchronization is $\alpha \in (0, 1]$, since $\min_{n\in\mathbb{N}} \frac{2n}{n+1} = 1$.

IV. Circulant Symmetric Case

We now relax the constraint of an all-to-all topology. We will show that any circulant symmetric network will desynchronize under our algorithm, specifically for the circulant symmetric ring topology.

A. Convergence of the Ring Topology

Let us now consider the case of the circulant symmetric ring topology, where every node is connected to the node directly ahead of and behind it in phase.

Theorem 2: A network with the circulant symmetric ring topology, where every node is connected to the node directly ahead of and behind it in phase, under the proposed algorithm will converge to the desynchronized state.

Proof: We can prove Theorem 2 if we can show that $\Delta p(k) < 0$ for a general k. The proof follows a similar approach to Theorem 1.

Using the result from (16) in Theorem 1, and knowing that $n = 2$ for the ring topology, we can write an expression for the change value during the next cycle as follows:

$$C_i(k+1) = C_i(k) + \alpha\left(\frac{\bar{C}_{i-1}(k) + \bar{C}_{i+1}(k)}{2} - \bar{C}_i(k)\right) \tag{27}$$

Squaring both sides, subtracting $C_i(k)^2$ from both sides, and using the fact that $C_i(k)\bar{C}_i(k) = \bar{C}_i(k)^2$ give us the term in the summation of (7):

$$\begin{aligned} &C_i(k+1)^2 - C_i(k)^2 \\ &\quad = \alpha\big(C_i(k)\bar{C}_{i-1}(k) + C_i(k)\bar{C}_{i+1}(k) - 2\bar{C}_i(k)^2\big) \\ &\qquad + \frac{\alpha^2}{4}\big(\bar{C}_{i-1}(k) + \bar{C}_{i+1}(k) - 2\bar{C}_i(k)\big)^2 \end{aligned} \tag{28}$$

Substituting (28) into (7) and then splitting the summation give us the following relationship, where we drop the cycle (k) notation for clarity:

$$\begin{aligned} \Delta p = \frac{1}{2}\Bigg(&\alpha\sum_{i=1}^N\big(C_i\bar{C}_{i-1} + C_i\bar{C}_{i+1} - 2\bar{C}_i^2\big) \\ &+ \frac{\alpha^2}{4}\sum_{i=1}^N\big(\bar{C}_{i-1} + \bar{C}_{i+1} - 2\bar{C}_i\big)^2\Bigg) \end{aligned} \tag{29}$$

Through expanding and recombining, the first summation term in (29) can be simplified, resulting in a convenient shifting of indices as follows:

$$\begin{aligned} &\alpha\sum_{i=1}^N\big(C_i\bar{C}_{i-1} + C_i\bar{C}_{i+1} - 2\bar{C}_i^2\big) \\ &\quad = -2\alpha\left(\sum_{i=1}^N\left[\bar{C}_i^2 - \frac{1}{2}(C_i\bar{C}_{i+1} + C_{i+1}\bar{C}_i)\right]\right) \end{aligned} \tag{30}$$

In a similar way, we can expand and recombine the second summation term in (29), also resulting in a convenient shifting of indices:

$$\begin{aligned} &\frac{\alpha^2}{4}\sum_{i=1}^N\big(\bar{C}_{i-1} + \bar{C}_{i+1} - 2\bar{C}_i\big)^2 \\ &\quad = \frac{\alpha^2}{4}\sum_{i=1}^N\big(\bar{C}_{i-1}^2 + \bar{C}_{i+1}^2 + 4\bar{C}_i^2 \\ &\quad - 4\bar{C}_i\bar{C}_{i-1} - 4\bar{C}_i\bar{C}_{i+1} + 2\bar{C}_{i-1}\bar{C}_{i+1}\big) \\ &\quad = \frac{\alpha^2}{4}\left(6\sum_{i=1}^N[\bar{C}_i^2] - 8\sum_{i=1}^N[\bar{C}_i\bar{C}_{i+1}] + 2\sum_{i=1}^N[\bar{C}_i\bar{C}_{i+2}]\right) \end{aligned} \tag{31}$$

By adding zero in a clever way, we can further simplify (31) as follows:

$$\frac{\alpha^2}{4}\left(6\sum_{i=1}^{N}[\bar{C}_i^2] - 8\sum_{i=1}^{N}[\bar{C}_i\bar{C}_{i+1}] + 2\sum_{i=1}^{N}[\bar{C}_i\bar{C}_{i+2}]\right)$$
$$+ \frac{\alpha^2}{4}\left(2\sum_{i=1}^{N}[\bar{C}_i^2] - 2\sum_{i=1}^{N}[\bar{C}_i^2]\right)$$
$$= 2\alpha^2\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+1}]\right) - \frac{\alpha^2}{2}\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+2}]\right) \tag{32}$$

We can now substitute (30) and (32) into (29):

$$\Delta p = \left(-\alpha\left[\sum_{i=1}^{N}\left(\bar{C}_i^2 - \frac{1}{2}[C_i\bar{C}_{i+1} + C_{i+1}\bar{C}_i]\right)\right]\right.$$
$$\left. + \alpha^2\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+1}]\right) - \frac{\alpha^2}{4}\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+2}]\right)\right) \tag{33}$$

Further simplification gives us the following final result:

$$\Delta p = (\alpha^2 - \alpha)\left[\sum_{i=1}^{N}(\bar{C}_i^2)\right] - \frac{\alpha^2}{4}\left[\sum_{i=1}^{N}(\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+2})\right]$$
$$- \sum_{i=1}^{N}\left(\alpha^2[\bar{C}_i\bar{C}_{i+1}] - \alpha\left[\frac{1}{2}(C_i\bar{C}_{i+1} + C_{i+1}\bar{C}_i)\right]\right) \tag{34}$$

It now leaves to show that (34) is always non-positive. We first notice that the first term is non-positive for $\alpha \in (0, 1)$, and the second term is non-positive for any positive α. These results now leave to show that the third term causes the entire expression to always be non-positive.

We will consider the following three cases separately: 1) Both C_i and C_{i+1} are positive, 2) one is positive, and the other non-positive, and 3) both are non-positive.

1) In the first case, the terms in brackets are equivalent, allowing us to combine the first and third term of (34):

$$(\alpha^2 - \alpha)\left[\sum_{i=1}^{N}(\bar{C}_i^2)\right] - \sum_{i=1}^{N}\left(\alpha^2[\bar{C}_i\bar{C}_{i+1}] - \alpha[\bar{C}_i\bar{C}_{i+1}]\right)$$
$$= (\alpha^2 - \alpha)\left[\sum_{i=1}^{N}(\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+1})\right] \tag{35}$$

   Eqn. (35) is non-positive for $\alpha \in (0, 1)$.
2) In the second case, the first term in brackets is zero, and the second term in brackets is non-positive. Thus, the whole term will be non-positive for any positive α.
3) In the third case, both terms in brackets are zero, making the whole term zero, and thus non-positive.

We next deal with the cases where Δp is zero and desynchronization has not yet been achieved. There are two such cases: 1) All change values are equal and positive ($C_1 = C_2 = \ldots = C_N > 0$), and 2) all change values are non-positive ($C_i \leq 0\, \forall i$). These are the same cases as for the all-to-all topology.

1) It is trivial to show that the first case cannot happen. If all change values are positive, then $\theta_i - \theta_{i-1} < \theta_{i+1} - \theta_i < \ldots < \theta_{i-1} - \theta_{i-2} < \theta_i - \theta_{i-1}$, which is a contradiction.
2) The second case can be true for one cycle since the nodes can sense only the "stale" phase from each node. However, this case will not be true for the next cycle. If no nodes move in a cycle, each node will then be observing the true phase of its connecting nodes, and not the stale values, in the next cycle. According to Lemma 2, $\sum_{i=1}^{N} C_i = 0$ holds in the next cycle. If desynchronization has not yet been achieved (i.e., not all change values are zero), then at least one node will have a positive change value, and will move forward in phase, causing the change in energy to be negative according to (34).

Therefore, the energy will always decrease and converge to zero for a choice of $\alpha \in (0, 1]$. According to Proposition 2, the network will converge to the desynchronized state. ■

B. Convergence of Circulant Symmetric Topologies

We can further prove the convergence of any network that can be described with an adjacency matrix that is both circulant and symmetric to the desynchronized state.

Theorem 3: A network with any circulant symmetric topology under the proposed algorithm will converge to the desynchronized state.

Proof: The proof follows an identical structure as the proof for Theorem 2, and hence will be omitted here, due to tedious bookkeeping. Given the associated adjacency matrix for the network, we can derive the change in energy after each cycle. It will be non-positive for $\alpha \in (0, 1]$, and cannot stay at zero unless the network is desynchronized. Thus, the network energy will approach zero, and, according to Proposition 2, the network will converge to the state of desynchronization. ■

Remark 7: The definition of circulant symmetric effectively limits the region of initial conditions under which convergence can be achieved. Enforcing this condition on the initial phases for large networks may deserve separate investigation.

Remark 8: Other types of topologies may also desynchronize under our algorithm, but we cannot prove that from the theory presented in this paper. One would need to show that the splay state is the only equilibrium point for the network under the proposed algorithm, and that the energy for that network topology decreases with each firing cycle.

V. Simulation Results

We will now simulate how the network evolves over time using our proposed algorithm. We offer comparisons between the all-to-all topology and circulant symmetric ring topology. The results of these simulations confirm our theoretical results from the previous sections. All simulations were done using MATLAB, with all PCOs having identical fundamental frequency, $\omega = 2\pi$, and identical period of one second.

A. Proposed Algorithm Results

We consider a network of six PCOs, with randomly chosen initial phases from the interval $[0, 2\pi]$. For the circulant symmetric ring topology, we order the initial phases first, and then

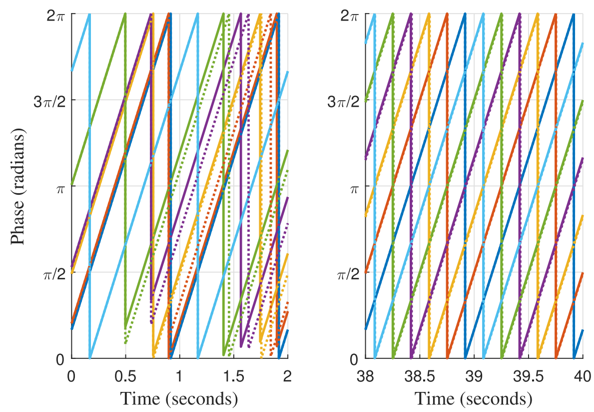


Fig. 5. Initial and final phase evolution for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict behavior under the all-to-all topology. Dotted lines depict behavior under the circulant symmetric ring topology. The network converges to the state of desynchronization.

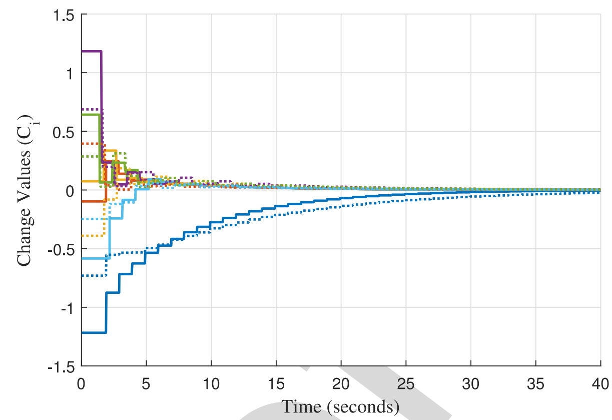


Fig. 7. The evolution of the change value C_i for each node. Solid lines depict behavior under the all-to-all topology. Dotted lines depict behavior under the circulant symmetric ring topology. All change values converge to zero.

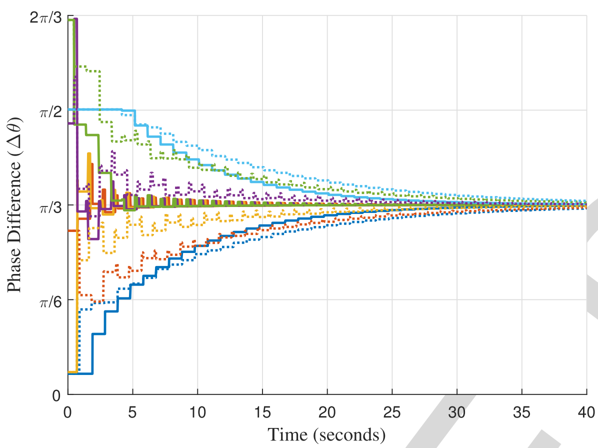


Fig. 6. Phase differences for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict values under the all-to-all topology. Dotted lines depict values under the circulant symmetric ring topology. The phase differences converge to the same value, one sixth of the unit circle distance, and all nodes become evenly spread out around the unit circle.

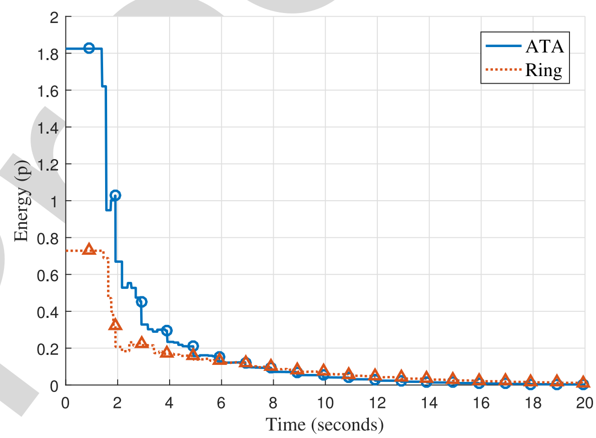


Fig. 8. Network energy as a function of time for six nodes with random initial state and $\alpha = 0.9$ for both the all-to-all (ATA) and circulant symmetric ring topologies. Markers indicate the measured energy after a complete firing cycle. The energy converges to zero as the network desynchronizes.

create the network such that every node is connected to the node directly ahead of and behind it in phase.

The evolution of the phases under our algorithm in the all-to-all topology and the circulant symmetric ring topology is given in Fig. 5. To visualize the evolution of the network toward desynchronization more easily, we also measure the phase difference between adjacent nodes, which we define in the following manner:

$$\Delta\theta_i = (\theta_{i+1} - \theta_i) \mod 2\pi, \text{ where } \theta_{N+1} = \theta_1 \tag{36}$$

Fig. 6 shows the plot of this phase distance between adjacent nodes, and confirms that desynchronization is indeed achieved.

In Fig. 7, we plot the change values for each node, which converge to zero with time. This behavior is to be expected, since once the network desynchronizes, no more jumps in phase need to be made.

In Fig. 8, the network energy as defined in (5) is shown for both the all-to-all and ring topology. It is important to note that the energy is not always strictly decreasing with regard to time. This result is primarily caused by the asymmetric nature of the update algorithm and secondarily due to the restriction that only forward jumps in phase can be made. However, the energy does strictly decrease while observing the value after every firing cycle, as was determined in our theoretical derivation. This result and our theoretical derivation show the importance of analyzing the evolution of the energy based on complete firing cycles instead of individual firing instants. Our treatment of desynchronization in this paper is different than the treatment of pulse-based desynchronization in the literature, which analyzes convergence to the splay state based on individual firing instances and will fail if applied directly to analyzing desynchronization in this paper.

As shown in Fig. 9, the rate at which the network energy converges is shown to be dependent on the network topology. More node connections cause the network to desynchronize more quickly than with fewer connections.

It is also important to note that the initial network energies under different network topologies are different. This difference is due to the increased number of connections in the all-to-all topology, making comparisons between the energy of the network and the closeness to desynchronization more difficult for

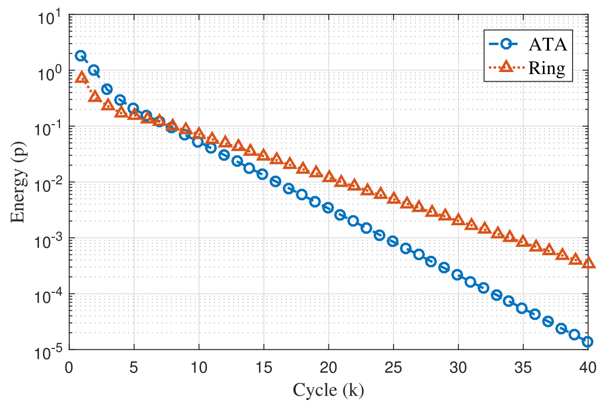


Fig. 9. Network energy after a complete firing cycle for six nodes with random initial state and $\alpha = 0.9$ for both the all-to-all (ATA) and circulant symmetric ring topologies. The energy strictly decreases when measured after a complete firing cycle. The energy convergence rate depends on the network topology.

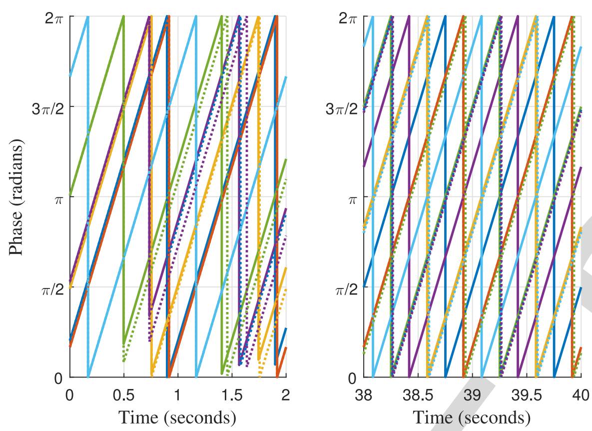


Fig. 10. Initial and final phase evolution for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict behavior under the all-to-all topology. Dotted lines depict behavior under the circulant symmetric ring topology. When unordered, the nodes in the circulant symmetric ring topology converge into groups of two, rather than spreading out evenly. The all-to-all topology is not affected by the reordering of the nodes.

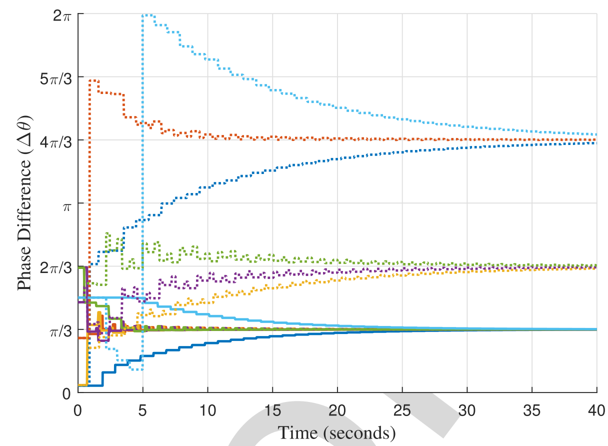


Fig. 11. Phase differences for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict values under the all-to-all topology. Dotted lines depict values under the circulant symmetric ring topology. The ring topology does not properly desynchronize or maintain the initial firing sequence when the nodes are unordered, and the resulting adjacency matrix is non-circulant.

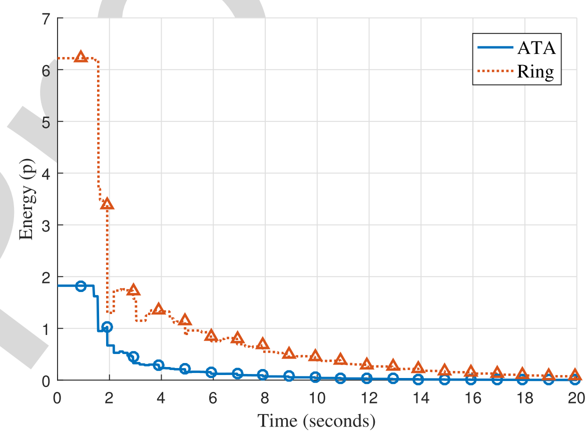


Fig. 12. Network energy as a function of time for six nodes with random initial state and $\alpha = 0.9$ for both the all-to-all (ATA) and ring topologies. Markers indicate the measured energy after a complete firing cycle. The energy of the ring topology still approaches zero, but does not indicate desynchronization, since the ordered adjacency matrix is non-circulant.

networks of different topologies and sizes. Normalizing the energy curves to have the same initial value aids in this comparison.

B. *Example of Insufficient Initial Condition*

Although our proposed algorithm relaxes the constraint of an all-to-all network topology, our theory only expands the set of potential network topologies to circulant symmetric networks with ordered nodes. If the network does not satisfy the conditions of being circulant or symmetric when the nodes are ordered, then our theoretical results do not guarantee that the network will desynchronize.

As an example, we take the initial node phases from the previous example and swap the phases of node 1 and node 2, keeping the same network connections from before. The all-to-all topology, with the swapped nodes, still has a circulant symmetric adjacency matrix, and thus meets the conditions for our proposed algorithm. However, the resulting node ordering for the circulant symmetric ring topology causes the network to have a symmetric, but non-circulant, adjacency matrix.

As seen in Fig. 10, the nodes in the all-to-all topology achieve phase desynchronization. However, the nodes of the ring topology do not achieve phase desynchronization, but rather converge into groups of two, and do not maintain the initial firing sequence. Fig. 11 more apparently shows the effect of the ring topology not achieving desynchronization, and also shows the shifting of relative node positions.

Even though the network energy approaches zero, as shown in Fig. 12, the required initial conditions were not met for this topology, and the energy does not indicate desynchronization.

C. *Comparison with DESYNC-STALE [7]*

Our proposed algorithm can be thought of as a variant and generalization of the DESYNC-STALE algorithm first given in [7]. The primary advantage that our proposed algorithm has over the DESYNC-STALE algorithm is the relaxation of the all-to-all topology constraint. Whereas the DESYNC-STALE algorithm assumes an all-to-all topology, our proposed algo-

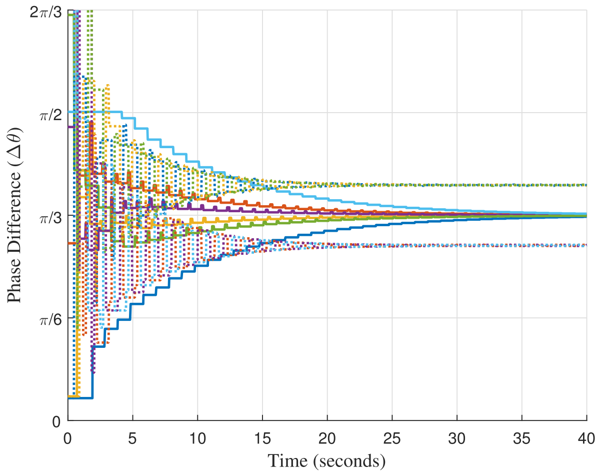


Fig. 13. Phase Differences for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict values under our proposed algorithm. Dotted lines depict values under the DESYNC-STALE algorithm. Both algorithms use the circulant symmetric network topology shown in Fig. 3b. The DESYNC-STALE algorithm is unable to completely desynchronize.

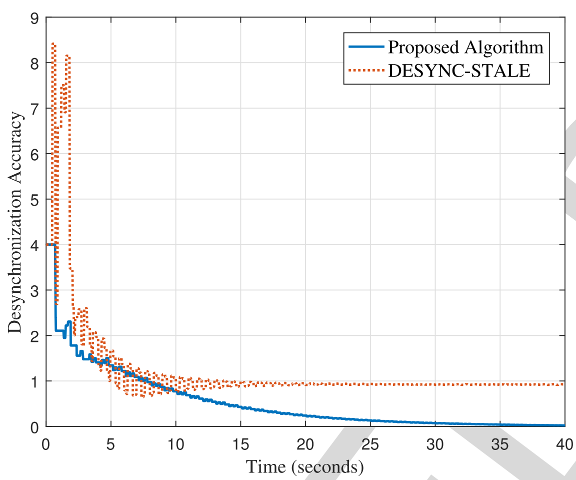


Fig. 14. Desynchronization accuracy of the network as defined in [7] as a function of time for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict behavior under our proposed algorithm. Dotted lines depict behavior under the DESYNC-STALE algorithm. The DESYNC-STALE algorithm fails to achieve network desynchronization.

rithm can generalize to any circulant symmetric network. Our proposed algorithm thus allows for a wider range of network topologies that can achieve desynchronization.

To illustrate the advantages of our proposed algorithm, we use the network topology shown in Fig. 3b, and use both algorithms to try to achieve the state of desynchronization. In the DESYNC-STALE algorithm, a node moves based on the nearest phase neighbor directly ahead of and behind it, whereas in our proposed algorithm the node moves based on all connected nodes.

In Fig. 13, we see that our proposed algorithm achieves evenly spaced node phases, whereas the DESYNC-STALE algorithm is unable to achieve desynchronization, as a result of each node ignoring the phase of one node to which it is connected.

Fig. 14 shows that our algorithm is able to accurately desynchronize under the given network topology, but the DESYNC-STALE algorithm is unable to accurately desynchronize. The

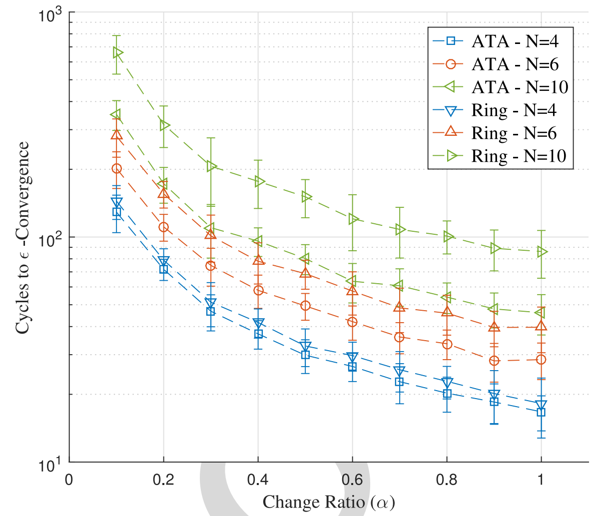


Fig. 15. The convergence rate of the network toward desynchronization for both the all-to-all (ATA) and circulant symmetric ring topologies. Points represent the mean of 25 runs at random initial conditions for each change ratio to reach ϵ-desynchrony for $\epsilon = 1 * 10^{-4}$. Error bars show one standard deviation about the mean.

desynchronization accuracy of the network $|\vec{\Delta}|$, defined in [7], is given to be "the sum of the absolute deviations from perfect desynchrony."

$$|\vec{\Delta}| = \sum_{i=1}^{N} \left| \Delta\theta_i - \frac{2\pi}{N} \right| \tag{37}$$

D. *Proposed Algorithm Convergence*

We now evaluate the speed of convergence of our algorithm for various sizes and types of topologies. To do so, we need some measure for when the network has desynchronized. We say that a network is ϵ-desynchronized after k cycles when the network energy satisfies $p(k) < \epsilon$.

In Fig. 15, we compare the relative convergence speed for various network topologies and sizes under different change ratios. We run 25 simulations at random initial conditions for each set of parameters, and plot the mean and standard deviation for when the network becomes ϵ-desynchronized. We see that more node connections cause the network to desynchronize more quickly for a given change ratio. Also, larger sizes of networks take more time to desynchronize. It is interesting to note that the ratio between the mean amount of cycles to ϵ-desynchronization for identical network size is approximately constant with regard to the change ratio.

VI. Conclusion

In this paper we present a novel control algorithm for achieving phase desynchronization in pulse coupled oscillators. Our proposed algorithm is robust, allowing the control of the network to be independent of the number of nodes in the network. Having the control independent of the size of the network allows for the network to add and remove nodes freely without modifying the update strategy. This robustness presents itself useful in many applications, where the network may have nodes dynamically entering and leaving the network.

It is our belief that this paper is the first to present a proof for the convergence to the desynchronized state for a pulse coupled oscillator network topology other than the all-to-all network. Specifically, we show that the network, under our proposed dynamics, will converge to the state of uniform phase desynchronization for any circulant symmetric network with ordered nodes, which includes the all-to-all and circulant symmetric ring topologies as special cases. Relaxing the constraint of an all-to-all topology makes our proposed algorithm desirable for various applications, where direct communication may not be available between every pair of nodes in the network.

Further analysis can still be made. Specifically, theoretically analyzing the convergence speed of the network is desirable so that the network can be tuned by an appropriate choice of topology and change ratio. This analysis will be complex, since it will be dependent on the type of network topology and initial conditions of the nodes. Also, further research can be done to see if additional types of networks can desynchronize under the proposed algorithm beyond the class of circulant symmetric networks.

Appendix

Proof for Lemma 1:

Proof: To prove this lemma, we need only to show that the firing sequence is invariant for one node, and then it will be true for all other nodes. To show the firing invariance for one node, we need only to show that the node will not pass the node directly ahead of it in phase within a firing cycle, regardless of whether the two nodes are connected.

Without loss of generality, let us consider node N, and show that it will not pass node 1 in phase during a firing cycle. We can describe this relationship mathematically with the following equation:

$$\alpha C_1 + \big((\theta_1 - \theta_N) \mod 2\pi\big) > \alpha C_N \tag{38}$$

We need only to consider the case when node N jumps forward in phase ($C_N > 0$), since the node will not pass the node directly in front of it if it does not jump. Without loss of generality, let us assume that the network has evolved to the instant right before node N fires ($\theta_N = 2\pi$). Thus, (38) can be rewritten as follows:

$$(\alpha C_1 + \theta_1) > \alpha C_N \tag{39}$$

We can express the amount that node N will jump when it fires as a function of the phases connected to it,

$$\begin{aligned} C_N &= \frac{1}{n}\sum_{l\in\mathcal{N}_N}\Big(-\big(2\pi-\big((\theta_l-\theta_N) \mod 2\pi\big)\big)+\pi\Big)\\ &= \frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_l\big)-\pi \end{aligned} \tag{40}$$

where $l \in \mathcal{N}_N$ corresponds to the node that caused the m^{th} instance of node N's recorded phase.

Similarly, we can express the amount that node 1 will jump,

$$\begin{aligned} C_1 &= \frac{1}{n}\sum_{j\in\mathcal{N}_1}\Big(-\big(2\pi-\big((\theta_j-\theta_1) \mod 2\pi\big)\big)+\pi\Big)\\ &= \frac{1}{n}\sum_{j\in\mathcal{N}_1}\big(\theta_j\big)-\theta_1-\pi \end{aligned} \tag{41}$$

where $j \in \mathcal{N}_1$ corresponds to the node that caused the m^{th} instance of node 1's recorded phase.

Since the nodes are ordered and the network is circulant, node 1 is connected to the nodes that are directly ahead in phase of the nodes that are connected to node N. Thus, we can express (41) in terms of the neighbors of node N:

$$C_1 = \frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_{l+1}\big)-\theta_1-\pi \tag{42}$$

By substituting (40) and (42) into (39), we achieve the following relationship:

$$\begin{aligned} &\alpha\left(\frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_{l+1}\big)-\theta_1-\pi\right)+\theta_1 > \alpha\left(\frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_l\big)-\pi\right)\\ &\quad\Rightarrow \frac{\alpha}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_{l+1}\big)+(1-\alpha)\theta_1 > \frac{\alpha}{n}\sum_{l\in\mathcal{N}_N}\theta_l \end{aligned} \tag{43}$$

Since the nodes are ordered, then we know that $\theta_{l+1} > \theta_l\ \forall l \in \mathcal{N}_N$. Thus, for any $\alpha \in (0, 1]$, the expression in (43) will be true, and after a cycle the relative node positions and the resulting firing sequence will be maintained.

An issue arises if $\alpha C_N > \theta_1$ holds when node N fires for the first time. When this condition is true, then, due to the asynchronous update of the algorithm, node N will fire again before node 1, breaking the invariance of the firing sequence. Even though the amount that node 1 will change would put it past node N again, node N will fire first. However, this problem does not occur in practice where the nodes begin with no knowledge of the positions of neighboring nodes in the network. Each node senses the location of neighboring nodes based on received pulses as the network evolves. Specifically, node N has the largest initial phase and will fire first. When it fires, it will not have received any pulses from its neighboring nodes, and thus it has $C_N = 0$, and will not jump. As the network evolves, node $(N-1)$ will fire next. When it fires, it will only have received a pulse from node N if $N \in \mathcal{N}_{N-1}$, and, from the phase response function in (1), any jump that node $(N-1)$ makes will not break invariance. A similar result occurs for node $(N-2)$ to node 2. When node 1 fires, it will have full knowledge of the positions of all neighboring nodes, and from the analysis above, will jump and maintain firing invariance. Thus, when node N fires the second time, Eq. (39) holds, and invariance is maintained. ■

Proof for Lemma 2:

Proof: From (1) and (2), we can write that

$$\begin{aligned} n_i \sum_{i=1}^{N} C_i &= \sum_{i=1}^{N} \sum_{m=1}^{n_i} Q(\theta_{i,m}) \\ &= \sum_{i=1}^{N} \sum_{l \in \mathcal{N}_i} \big(- [(\theta_i - \theta_l) \mod 2\pi] + \pi \big) \quad (44) \end{aligned}$$

where $l \in \mathcal{N}_i$ corresponds to the node that caused the m^{th} instance of node i's recorded phase.

Since the network is symmetric, then for two nodes j and k with a connection between them (assuming $j > k$), node j will record the m_1^{th} instance of its phase (denoted θ_{j,m_1}) when node k fires, and node k will record the m_2^{th} instance of its phase (denoted θ_{k,m_2}) when node j fires. Note that θ_{j,m_1} represents the phase difference between nodes k and j, and θ_{k,m_2} represents the phase difference between nodes j and k. Since no nodes have jumped in phase, the phase difference measured by these two nodes is the true phase difference between them and is equivalent for both nodes. Combining these two terms causes them to cancel.

$$\begin{aligned} &Q(\theta_{j,m_1}) + Q(\theta_{k,m_2}) \\ &= \big(- (\theta_j - \theta_k) + \pi \big) + \big(- (\theta_k - \theta_j + 2\pi) + \pi \big) = 0 \quad (45) \end{aligned}$$

Since the entire network is symmetric, and no nodes have jumped in the cycle, the whole expression can be recombined and canceled.

$$n_i \sum_{i=1}^{N} C_i = 0 \Rightarrow \sum_{i=1}^{N} C_i = 0 \quad (46)$$

Thus, the desired result is achieved. ■

Acknowledgment

The authors would like to thank those who reviewed the initial drafts of this paper.

References

[1] R. Mirollo and S. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, 1990.

[2] P. Goel and B. Ermentrout, "Synchrony, stability, and firing patterns in pulse-coupled oscillators," *Phys. D, Nonlinear Phenom.*, vol. 163, no. 3, pp. 191–216, 2002.

[3] C. S. Peskin, *Mathematical Aspects of Heart Physiology*. New York, NY, USA: Courant Inst. of Math. Sci., New York Univ., New York University, 1975.

[4] B. Ermentrout, "Type I membranes, phase resetting curves, and synchrony," *Neural Comput.*, vol. 8, no. 5, pp. 979–1001, 1996.

[5] H. Gao and Y. Q. Wang, "Integrated communication and control design for decentralized collective motion coordination," Submitted.

[6] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion: All-to-all communication," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 811–824, May 2007.

[7] A. Patel, J. Degesys, and R. Nagpal, "Desynchronization: The theory of self-organizing algorithms for round-robin scheduling," in *Proc. 1st Int. Conf. Self-Adapt. Self-Organizing Syst.*, Cambridge, MA, USA, Jul. 2007, pp. 87–96.

[8] J. Degesys and R. Nagpal, "Towards desynchronization of multi-hop topologies," in *Proc. 2nd IEEE Int. Conf. Self-Adapt. Self-Organizing Syst.*, Oct. 2008, pp. 129–138.

[9] R. Pagliari, Y. W. P. Hong, and A. Scaglione, "Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 4, pp. 564–575, May 2010.

[10] F. Núñez, Y. Q. Wang, and F. J. Doyle III, "Global synchronization of pulse-coupled oscillators interacting on cycle graphs," *Automatica*, vol. 52, pp. 202–209, 2015.

[11] F. Núñez, Y. Q. Wang, and F. J. Doyle III, "Synchronization of pulse-coupled oscillators on (strongly) connected graphs," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1710–1715, Jun. 2015.

[12] F. Dörfler and F. Bullo, "Synchronization in complex networks of phase oscillators: A survey," *Automatica*, vol. 50, no. 6, pp. 1539–1564, 2014.

[13] Y. Q. Wang and F. J. Doyle III, "Optimal phase response functions for fast pulse-coupled synchronization in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5583–5588, Oct. 2012.

[14] Y. Q. Wang, F. Núñez, and F. J. Doyle III, "Energy-efficient pulse-coupled synchronization strategy design for wireless sensor networks through reduced idle listening," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5293–5306, Oct. 2012.

[15] S. Phillips and R. G. Sanfelice, "Results on the asymptotic stability properties of desynchronization in impulse-coupled oscillators," in *Proc. Amer. Control Conf.*, Washington, DC, USA, Jun. 2013, pp. 3272–3277.

[16] N. Deligiannis, J. F. C. Mota, G. Smart, and Y. Andreopoulos, "Decentralized multichannel medium access control: Viewing desynchronization as a convex optimization method," in *Proc. 14th Int. Conf. Inf. Process. Sensor Netw.*, New York, NY, USA, 2015, pp. 13–24.

[17] H. Gao and Y. Q. Wang, "Analysis and design of phase desynchronization in pulse-coupled oscillators." (2016). [Online]. Available: http://arxiv.org/pdf/1603.03313v1.pdf

[18] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion with limited communication," *IEEE Trans. Autom. Control*, vol. 53, no. 3, pp. 706–719, Apr. 2008.

[19] R. Gentz, A. Scaglione, L. Ferrari, and Y. W. P. Hong, "Pulsess: A pulse-coupled synchronization and scheduling protocol for clustered wireless sensor networks," *IEEE Internet Things J.*, in press, 2016.

[20] P. Taechalertpaisarn, S. Choochaisri, and C. Intanagonwiwat, "An orthodontics-inspired desynchronization algorithm for wireless sensor networks," in *Proc. 2011 IEEE 13th Int. Conf. Commun. Technol.*, Sep. 2011, pp. 631–636.

[21] D. Buranapanichkit, N. Deligiannis, and Y. Andreopoulos, "Convergence of desynchronization primitives in wireless sensor networks: A stochastic modeling approach," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 221–233, Jan. 2015.

[22] H. K. Khalil, *Nonlinear Systems*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

Timothy Anglea (S'14) received the B.S. degree in engineering from Bob Jones University, Greenville, SC, USA, in 2015. He is currently working toward the M.S. degree in electrical engineering in the Electrical and Computer Engineering Department, Clemson University, Clemson, SC, USA, with a focus in intelligent systems.

He received first place in the 2015 IEEE SouthEastCon student paper competition with research on a simple control strategy for autonomous vehicles. His current research focuses on the dynamics and applications of pulse-coupled oscillators.

Yongqiang Wang (SM'13) was born in Shandong, China. He received the B.S. degree in electrical engineering and automation and the B.S. degree in computer science and technology from Xi'an Jiaotong University, Shaanxi, China, in 2004. He received the M.Sc. and Ph.D. degrees in control science and engineering from Tsinghua University, Beijing, China, in 2009.

From 2007 to 2008, he was with the University of Duisburg-Essen, Duisburg, Germany, as a Visiting Student. He was a Project Scientist at the University of California, Santa Barbara. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA. His research interests include cooperative and networked control, synchronization of wireless sensor networks, systems modeling and analysis of biochemical oscillator networks, and model-based fault diagnosis. He received the 2008 Young Author Prize from IFAC Japan Foundation for a paper presented at the 17th IFAC World Congress in Seoul.

Phase Desynchronization: A New Approach and Theory Using Pulse-Based Interaction

Timothy Anglea, *Student Member, IEEE*, and Yongqiang Wang, *Senior Member, IEEE*

***Abstract*—In this paper, we present a novel approach for achieving phase desynchronization in a pulse-coupled oscillator network. Ensuring phase desynchronization is a difficult problem, and existing results are constrained to a completely interconnected network and a fixed number of nodes. Our approach is more robust than previous approaches, removing the constraint of a fixed number of nodes. The removal of this constraint is significant because it allows the network to receive and drop nodes freely without any change to the phase update strategy. Also, to our knowledge, our approach is the first to prove the convergence to the desynchronized state for a topology that is more general than the all-to-all topology. More specifically, our approach is applicable to any circulant, symmetric network topology, including the circulant symmetric ring topology. Rigorous mathematical proofs are provided to support the result that any circulant, symmetric network with ordered phases under our proposed algorithm will converge to uniform phase desynchronization. Simulation results are also presented to demonstrate the algorithm's performance.**

***Index Terms*—Pulse-coupled oscillator, desynchronization, phase desynchronization, splay state.**

I. Introduction

THE study and analysis of pulse coupled oscillators (PCOs) is a currently active field of engineering research. Initially proposed to model the synchronization of many biological systems, such as the flashing of fireflies [1], [2] and firing neurons [3], [4], the model has been expanded to encompass more complex behaviors [5], [6]. One such behavior is desynchronization, a more difficult problem and the inverse problem to synchronization. Desynchronization has many important applications, primarily in the area of decentralized network communication [7]–[9], but also in the design of analog-to-digital converters and the control of traffic intersection flow [7]. Compared with the increased research on achieving phase synchronization [10]–[14], there is much less research on desynchronization.

In this paper we propose a desynchronization strategy and algorithm that we believe to be unique to the literature for desynchronizing pulse coupled oscillator (PCO) networks. A PCO network is one of connected oscillators, or nodes, each with an associated phase that continuously cycles from 0 to 2π. A node fires a pulse to all of its neighboring nodes when it reaches a threshold value, often 2π. Based upon the timing at which these pulses are received and fired, the PCOs can respond in a desirable way. The great benefits of a PCO network are that the communication between nodes is a single, identical pulse, minimizing latency and data corruption, and that the control of the network can be decentralized.

Many algorithms and approaches have been proposed to achieve desynchronization. Some control strategies ensure weak desynchronization, where the timing between firings is constant, but the phases are constantly shifting [7], [15]. Other control strategies ensure strong, or phase, desynchronization, but must assume a completely interconnected, or all-to-all, network and a fixed number of nodes [7]–[9], [16], [17]. Some have tried to relax the constraint of an all-to-all connection topology, but instead require either a continuously coupled, rather than pulse coupled, network [6], [18], or the use of additional types of nodes and pulse transmissions [19]. Work has also been done to improve the speed of convergence assuming an all-to-all topological network [20], [21]. In this paper, we propose an approach that relaxes the constraint of complete interconnection and removes the requirement of a fixed number of nodes to achieve complete phase desynchronization. To the best of our knowledge, our approach is the first to prove that a network topology other than the all-to-all topology can be used to achieve uniform phase desynchronization with identical PCOs.

In Section II, we will describe our proposed algorithm to achieve desynchronization for circulant symmetric networks with ordered nodes. We note that the algorithm is independent of the number of nodes in the network, resulting in a more robust control, allowing for nodes to join and leave the network freely without affecting the update rule. In Section III, we will provide rigorous proof for guaranteed desynchronization under the all-to-all topology using our algorithm. In Section IV, we will relax the constraint of the all-to-all topology, showing that any circulant symmetric network topology, and specifically the circulant symmetric ring topology, will desynchronize using the proposed algorithm. In Section V, we will provide simulation results showing how the network desynchronizes under our proposed algorithm and analyze the convergence rate of our algorithm. In Section VI, we will conclude and present areas of further research.

II. Network Dynamics

It is our belief that the algorithm used in this paper is unique in the literature on PCOs. The goal of desynchronization is to spread the phases of all oscillators, or nodes, evenly around the unit circle. This state of desynchronization is sometimes termed

Manuscript received June 11, 2016; revised September 15, 2016 and October 21, 2016; accepted November 10, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Subhrakanti Dey. The work was supported in part by the U.S. Army Engineer Research & Development Center's Basic Research Program 455337 and the Institute for Collaborative Biotechnologies through grant W911NF-09-0001.

The authors are with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634 USA (e-mail: tbangle@clemson.edu; yongqiw@clemson.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2016.2633246

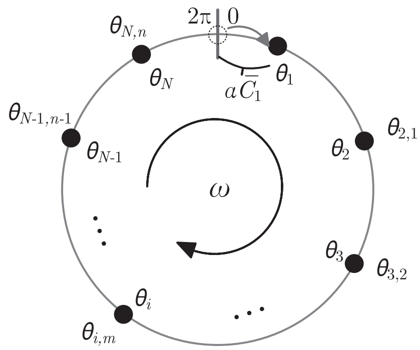


Fig. 1. A snapshot of current and recorded phases when a node fires for a network of N pulse coupled oscillators with fundamental frequency ω. When a node fires, all connected nodes record the m^{th} instance of their current phase. For example, when node 1 fires, node 2 records $\theta_{2,1}$ and node 3 records $\theta_{3,2}$. Node 3 recorded $\theta_{3,1}$ when node 2 fired previously.

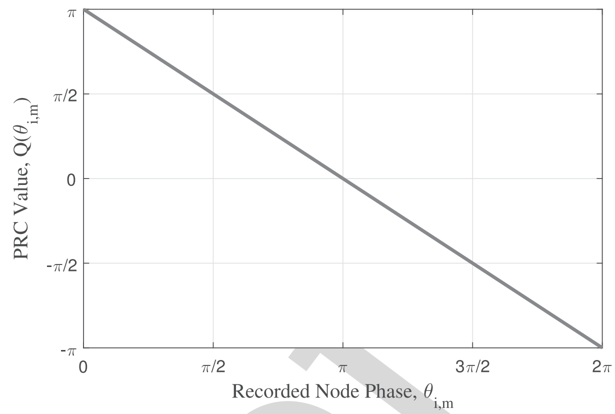


Fig. 2. Phase Response Curve, Q, as given in (1) for the proposed algorithm.

the splay state. Inspiration for the proposed algorithm is based on the DESYNC and DESYNC-STALE algorithms proposed in [7]. However, different from [7], which only allows nodes to react based on the nodes directly ahead of and behind it in phase, our approach allows nodes to react to all nodes with which it is connected. A mechanical analogy can be made, where each node, spread around the unit circle, is connected to other nodes by compressed springs [12]. The closer the nodes are in phase, the more force the nodes will exert in moving the other away. An equilibrium is achieved when all of the nodes are as far away from each other in phase as possible, the desired state of desynchronization.

A. General Algorithm Definitions

Let us assume a network of N pulse coupled oscillators, as shown in Fig. 1. Let $\theta_i \in [0, 2\pi]$ be the associated phase of oscillator, or node, $i \in \mathcal{V} = \{1, 2, \cdots, N\}$, with ω as the fundamental frequency for all oscillators. The state of the network $\vec{\theta} = [\theta_i]$ is a column vector representing the phase of the N nodes.

We define $\mathcal{A}_{\mathcal{R}} = [a_{ij}] \in \mathbb{R}^{N \times N}$ to be the adjacency matrix for the network, where element a_{ij} is 1 if there is a connection from node i to node j, and 0 if there is no connection. The adjacency matrix defines the topology for the network. $\mathcal{A}_{\mathcal{R}}$ is zero diagonal ($a_{ii} = 0\ \forall i \in \mathcal{V}$), and is symmetrical if connections between nodes are bidirectional. For each node i, we let $\mathcal{N}_i$ denote the set of nodes connected to node i, where $\mathcal{N}_i \subset \mathcal{V}$, and $|\mathcal{N}_i| = n_i$.

Our proposed algorithm can be described as follows.

1) A node fires a pulse when its phase reaches 2π. When the node fires, all connected nodes receive the pulse.
2) When a node receives a pulse, the node records the value of *its own* current phase, $\theta_{i,m}$. This value is stored to be used when the node fires, i.e., when its phase reaches 2π.
3) When a node fires, emits a pulse, and resets its phase to zero, it jumps, and the amount that the node jumps is dependent upon all of the recorded phase values due to the firing of the other nodes. The node computes the phase response curve (PRC) value, $Q(\theta_{i,m})$, for each of its own recorded phase values, and then computes the change value, C_i, or the amount that the node needs to jump, from the PRC values. If the change value is positive, the node will jump by a proportional amount, αC_i, where α is a positive real number; otherwise, the node will not jump. Thus, the node will only jump forward in phase.

The phase update for the network is asynchronous. Each node updates its phase when it fires, and does not wait for other nodes. However, the amount that each node jumps is determined by the recorded phase values found from the previous firing cycle. In other words, the network relies on "stale" information obtained from the timing of received pulses, as in the DESYNC-STALE algorithm in [7]. In this paper, the invariance of the firing sequence of the nodes is assumed, which will be proven later.

Let us denote the stroboscopic recorded phase values when a node receives a pulse as $\theta_{i,m}$, where i is the node that received the pulse, and $m \in \{1, 2, \ldots, n_i\}$ is the index of the recorded phase value of that node due to the pulses received from neighboring nodes in $\mathcal{N}_i$. Thus, $0 < \theta_{i,1} < \theta_{i,2} < \ldots < \theta_{i,n_i} < 2\pi$.

We define the PRC of the network, Q, in the following way:

$$Q(\theta_{i,m}) = -(\theta_{i,m}) + \pi \tag{1}$$

An interpretation for the PRC function in Fig. 2 is that if two nodes are close in phase, they will exert a force on each other that causes them to move farther away from each other. Otherwise, when the nodes are as far apart as possible ($\theta_{i,m} = \pi$), they will exert no force on each other.

B. Pulse Dynamics

The amount that a node jumps is dependent on the relative position and total number of other nodes from which it has received pulses. For our algorithm, the node will compute the average amount of the calculated PRC values from its recorded phase values. Let us, then, define the change value, C_i, to be this average amount:

$$C_i = \frac{1}{n_i} \sum_{m=1}^{n_i} Q(\theta_{i,m}) \tag{2}$$

As mentioned earlier, when a node fires, it computes its change value and then jumps by a portion of that amount if the change value is positive. Otherwise, the node does not jump. This jump rule is described mathematically as

$$\theta_i^+ = \begin{cases} \alpha C_i & \text{if } C_i > 0 \\ 0 & \text{if } C_i \leq 0 \end{cases} \tag{3}$$

To summarize, a node will jump in phase only when it fires, and then only if it will jump forward in phase.

Remark 1: In application, the phase could be allowed to jump backward, allowing for faster convergence. However, there will exist the problem of having the state, in theory, be less than zero for a period of time. Care will need to be taken to ensure the proper behavior of the network and to maintain the proper firing sequence.

Let us define the change vector, C, as the column vector of the phase change values corresponding to all nodes in the network:

$$C = [C_i] \in \mathbb{R}^N \tag{4}$$

C. Network Equilibrium

In order for us to show that the network will desynchronize, we need to show that the splay state is the only equilibrium point for the network under our proposed algorithm. For this paper we will show that the splay state is the equilibrium point for any circulant symmetric network with ordered nodes.

Definition 1: Nodes in a PCO network are considered ordered when they are initially numbered in order by their phase magnitude, i.e., $0 \leq \theta_1 < \theta_2 < \ldots < \theta_{N-1} < \theta_N \leq 2\pi$.

Definition 2: A network with ordered nodes is called circulant symmetric if the resulting adjacency matrix $\mathcal{A}_{\mathcal{R}}$ for the ordered nodes is both circulant and symmetric.

Remark 2: For our paper, a circulant symmetric network assumes ordered nodes.

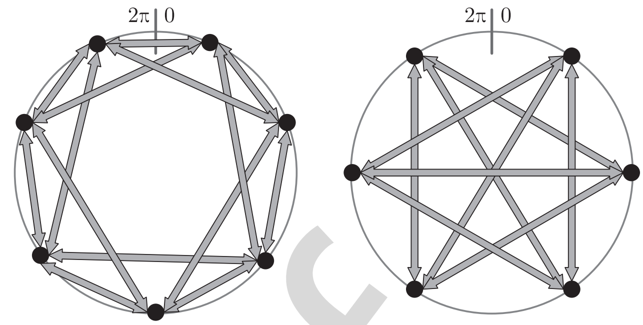
Remark 3: For circulant symmetric networks, all nodes have the same number of neighbors, or $\mathcal{N}_i = n \; \forall i \in \mathcal{V}$. For the all-to-all network topology, $n = N - 1$, and for the ring topology, $n = 2$.

Proposition 1: For any circulant symmetric network under the proposed algorithm, the network is desynchronized if and only if the change vector in (4) is the zero vector.

Proof: When the network is in the state of desynchronization, all nodes are as far apart from each other in phase as possible. In this state, the phases are equally spaced around the region $[0, 2\pi]$. Since the network is circulant and symmetric, for each node in the network, there will be equal force pushing the node forward in phase as there is pushing it backward in phase. Thus, over a cycle of firings, C_i will be zero for all nodes, and no node will jump in phase.

If the network is not desynchronized, then since the network is circulant and symmetric, there will be an imbalance in the network, causing the change vector to be non-zero. ■

Proposition 1 indicates that the desynchronized state is the only equilibrium state under our setup. This result gives us insight into how to determine when the network has reached desynchronization.



(a) N = 7 (b) N = 6

Fig. 3. Examples of circulant symmetric networks that will achieve desynchronization under the proposed algorithm. (a) Seven node network with each node connected to two nodes ahead in phase and two nodes behind in phase. (b) Six node network where each node is without connections to the nodes directly ahead and behind it in phase.

D. Network Energy Measure

Let us define an energy function for the network using the change vector after the k^{th} firing cycle:

$$p(k) = \frac{1}{2} C(k)^T C(k) = \frac{1}{2} \sum_{i=1}^{N} C_i(k)^2 \tag{5}$$

By the definition, the energy is non-negative everywhere, and zero only when the change vector from (4) is the zero vector. Thus, (5) is a potential "Lyapunov" function. The energy is computed after every firing cycle, when all nodes have fired once in the sequence.

We also need to define the change in the energy as a function of the cycle:

$$\Delta p(k) = p(k+1) - p(k) \tag{6}$$

Proposition 2: For any circulant symmetric network under the proposed algorithm, the network will desynchronize if $\Delta p(k) < 0$ for any k when $p(k) > 0$.

Using LaSalle's theorem given in [22], Proposition 2 follows easily from the defined energy function and Proposition 1.

In the next sections, we will prove that the condition in Proposition 2 holds for the all-to-all topology and circulant symmetric topologies in general, specifically for the circulant symmetric ring topology.

Remark 4: The all-to-all and circulant symmetric ring topologies are special cases of circulant symmetric networks. Networks with adjacency matrices that are both symmetric and circulant are a subset of highly symmetrical networks, as described in [12].

III. All-to-All Case

A. Preliminaries

In order for the network to desynchronize, we need to ensure that the firing sequence of the nodes remains unchanged between cycles. If nodes switch order after a cycle, then the resulting network with ordered nodes may no longer be circulant.

Lemma 1: The firing sequence for any circulant symmetric network is invariant between cycles under the proposed algorithm for any change ratio $\alpha \in (0, 1]$.

See Appendix for proof.

To simplify our proofs, we will utilize the following property of the change values C_i.

Lemma 2: For a circulant symmetric network under the proposed algorithm, $\sum_{i=1}^{N} C_i = 0$ holds if no nodes have jumped during the previous cycle.

See Appendix for proof.

Remark 5: For both Lemma 1 and Lemma 2, it is necessary that the network be circulant symmetric. Thus, we restrict our analysis to this class of networks.

B. Convergence of the All-to-All Topology

Let us first consider a PCO network with an all-to-all topology. This network has been shown to converge under other desynchronization algorithms [7], so we will show that the network will also converge under our proposed algorithm.

Theorem 1: A network with an all-to-all topology under the proposed algorithm will converge to the desynchronized state.

Proof: We can prove Theorem 1 if we can show that $\Delta p(k) < 0$ for a general k when $p(k)$ is non-zero.

Let us first note that the expression for the change in energy can be rewritten by combining (5) and (6) and simplifying:

$$\Delta p(k) = \frac{1}{2}\sum_{i=1}^{N}\left[C_i(k+1)^2 - C_i(k)^2\right] \tag{7}$$

By substituting the PRC in (1) into (2), we can write an expression for the change value for a node at a given cycle:

$$C_i(k) = \frac{1}{n}\sum_{m=1}^{n}\big(-\theta_{i,m}(k)+\pi\big) \tag{8}$$

Let $j \in \mathcal{N}_i$ be an element in the set of nodes that are connected to node i. For the all-to-all topology, $\mathcal{N}_i$ includes all of the network nodes except node i. The recorded phase values can be expressed as

$$\theta_{i,m}(k) = 2\pi - \Big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\Big), \tag{9}$$

where j corresponds to the node that resulted in the m^{th} recorded phase value. We can thus express (8) in terms of the node phases during the current cycle as follows:

$$C_i(k) = \frac{1}{n}\sum_{j\in\mathcal{N}_i}\Big[\Big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\Big) - \pi\Big] \tag{10}$$

Similarly, the change value for the same node at the next cycle can also be expressed in terms of the node phases during the next cycle:

$$\begin{aligned} C_i(k+1) &= \frac{1}{n}\sum_{m=1}^{n}\big(-\theta_{i,m}(k+1)+\pi\big) \\ &= \frac{1}{n}\sum_{j\in\mathcal{N}_i}\Big[\Big(\big(\theta_j(k+1)-\theta_i(k+1)\big) \mod 2\pi\Big) - \pi\Big] \end{aligned} \tag{11}$$

We now need an expression for the recorded phase value for the next cycle. We know that, in the next cycle, the phases may have jumped a fractional amount of C_i, which may have been non-positive, from the previous cycle following the update rule given in (3). The amount that node i jumps during the current cycle, which we will denote as $\bar{C}_i$, can be mathematically described as the maximum of zero and C_i, or

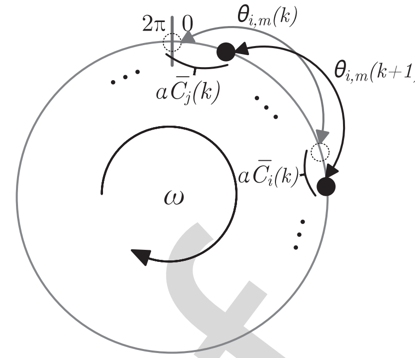


Fig. 4. Illustration for (13), showing the change in $\theta_{i,m}$ from one cycle to the next.

$$\bar{C}_i = \max\{0, C_i\} \tag{12}$$

Eqn. (12) allows us to write an expression for the next recorded phase value based on the previous recorded phase value and the amount that each node jumped in the previous cycle, as illustrated in Fig. 4.

$$\theta_{i,m}(k+1) = \theta_{i,m}(k) + \alpha\big(\bar{C}_i(k) - \bar{C}_j(k)\big) \tag{13}$$

Using the expression in (9), we can rewrite (13) as

$$\begin{aligned} &2\pi - \Big(\big(\theta_j(k+1)-\theta_i(k+1)\big) \mod 2\pi\Big) \\ &= 2\pi - \Big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\Big) + \alpha\big(\bar{C}_i(k)-\bar{C}_j(k)\big) \\ &\qquad \Longrightarrow \\ &\Big(\big(\theta_j(k+1)-\theta_i(k+1)\big) \mod 2\pi\Big) \\ &= \Big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\Big) + \alpha\big(\bar{C}_j(k)-\bar{C}_i(k)\big) \end{aligned} \tag{14}$$

Thus, the expression for the next change value, by substituting (14) into (11), can be written as

$$\begin{aligned} C_i(k+1) &= \frac{1}{n}\sum_{j\in\mathcal{N}_i}\Big[\Big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\Big) \\ &\qquad + \alpha\big(\bar{C}_j(k)-\bar{C}_i(k)\big) - \pi\Big] \\ &= \frac{1}{n}\sum_{j\in\mathcal{N}_i}\Big[\Big(\big(\theta_j(k)-\theta_i(k)\big) \mod 2\pi\Big) - \pi\Big] \\ &\qquad + \frac{\alpha}{n}\sum_{j\in\mathcal{N}_i}\big[\bar{C}_j(k)-\bar{C}_i(k)\big] \end{aligned} \tag{15}$$

Recognizing the expression for the change value from (10), we can simplify (15) as follows:

$$C_i(k+1) = C_i(k) + \frac{\alpha}{n}\sum_{j\in\mathcal{N}_i}\big[\bar{C}_j(k)-\bar{C}_i(k)\big] \tag{16}$$

Squaring both sides of (16) and subtracting $C_i(k)^2$ from both sides give us the following expression for the summation term in (7):

$$C_i(k+1)^2 - C_i(k)^2 = \frac{2\alpha}{n} C_i(k) \sum_{j\in\mathcal{N}_i} \left[\bar{C}_j(k) - \bar{C}_i(k)\right] + \left(\frac{\alpha}{n}\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k) - \bar{C}_i(k)\right]\right)^2 \quad (17)$$

We can rearrange and simplify (17) as follows:

$$\begin{aligned} &C_i(k+1)^2 - C_i(k)^2 \\ &= \frac{2\alpha}{n}\left[C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) - C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_i(k)\right] \\ &\quad + \frac{\alpha^2}{n^2}\left(\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k) - \bar{C}_i(k)\right]\right)^2 \\ &= \frac{2\alpha}{n}\left[C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) - nC_i(k)\bar{C}_i(k)\right] \\ &\quad + \frac{\alpha^2}{n^2}\left(\sum_{j\in\mathcal{N}_i}\left[\bar{C}_j(k)\right] - n\bar{C}_i(k)\right)^2 \end{aligned} \quad (18)$$

We can use the fact that $C_i(k)\bar{C}_i(k) = \bar{C}_i(k)^2$, and then expand and recombine (18) to achieve the following:

$$\begin{aligned} &C_i(k+1)^2 - C_i(k)^2 \\ &= \frac{2\alpha}{n}C_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) - 2\alpha\bar{C}_i(k)^2 \\ &\quad + \frac{\alpha^2}{n^2}\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j(k)\right]^2 - \frac{2\alpha^2}{n}\bar{C}_i(k)\sum_{j\in\mathcal{N}_i}\bar{C}_j(k) + \alpha^2\bar{C}_i(k)^2 \\ &= (\alpha^2 - 2\alpha)\left(\bar{C}_i(k)^2\right) + \frac{\alpha^2}{n^2}\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j(k)\right]^2 \\ &\quad + \left(\frac{2\alpha}{n}C_i(k) - \frac{2\alpha^2}{n}\bar{C}_i(k)\right)\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j(k)\right] \end{aligned} \quad (19)$$

Substituting (19) into (7) gives us an expression for the change in network energy, where we drop the cycle (k) notation for clarity:

$$\begin{aligned} \Delta p &= \frac{1}{2}\sum_{i=1}^{N}\left[(\alpha^2 - 2\alpha)(\bar{C}_i^2) + \frac{\alpha^2}{n^2}\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)^2 \right. \\ &\quad \left. + \left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right] \\ &= \frac{1}{2}\left((\alpha^2 - 2\alpha)\sum_{i=1}^{N}(\bar{C}_i^2) + \frac{\alpha^2}{n^2}\sum_{i=1}^{N}\left[\sum_{j\in\mathcal{N}_i}\bar{C}_j\right]^2 \right. \\ &\quad \left. + \sum_{i=1}^{N}\left[\left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \end{aligned} \quad (20)$$

We can further simplify the second term in (20) by expanding and recombining terms as follows:

$$\sum_{i=1}^{N}\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)^2 = n\sum_{i=1}^{N}\bar{C}_i^2 + (n-1)\left(\sum_{i=1}^{N}\bar{C}_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right) \quad (21)$$

Substituting (21) into (20) and recombining terms give the following result:

$$\begin{aligned} \Delta p &= \frac{1}{2}\left((\alpha^2 - 2\alpha)\sum_{i=1}^{N}(\bar{C}_i^2) \right. \\ &\quad + \frac{\alpha^2}{n^2}\left(n\sum_{i=1}^{N}\bar{C}_i^2 + (n-1)\left[\sum_{i=1}^{N}\bar{C}_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \\ &\quad \left. + \sum_{i=1}^{N}\left[\left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \\ &= \frac{1}{2}\left((\alpha^2 - 2\alpha)\sum_{i=1}^{N}(\bar{C}_i^2) + \frac{\alpha^2}{n}\sum_{i=1}^{N}(\bar{C}_i^2) \right. \\ &\quad + \sum_{i=1}^{N}\left[\frac{\alpha^2(n-1)}{n^2}\bar{C}_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right] \\ &\quad \left. + \sum_{i=1}^{N}\left[\left(\frac{2\alpha}{n}C_i - \frac{2\alpha^2}{n}\bar{C}_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right]\right) \end{aligned} \quad (22)$$

Further recombination and simplification lead to the following final result:

$$\begin{aligned} \Delta p &= \left(\frac{n+1}{2n}\alpha^2 - \alpha\right)\sum_{i=1}^{N}(\bar{C}_i^2) \\ &\quad - \frac{1}{n}\sum_{i=1}^{N}\left[\left(\frac{n+1}{2n}\alpha^2\bar{C}_i - \alpha C_i\right)\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right)\right] \end{aligned} \quad (23)$$

It now leaves to show that (23) is always non-positive, and only stays zero when the network is desynchronized. Let us denote Δp_i as the i^{th} term of Δp, corresponding to the i^{th} change value, such that

$$\Delta p = \sum_{i=1}^{N}\Delta p_i. \quad (24)$$

Let us arrange the Δp_i terms into two groups: 1) Set $\mathcal{M}$ corresponding to when C_i is non-positive, and 2) Set $\mathcal{S}$ when it is positive.

1) If $C_i \leq 0$, then, from (12), $\bar{C}_i = 0$, and $\Delta p_i \in \mathcal{M}$ can be simplified as follows:

$$\Delta p_i = \frac{\alpha}{n}C_i\left(\sum_{j\in\mathcal{N}_i}\bar{C}_j\right) \quad (25)$$

Eqn. (25) is non-positive for any positive α, and the sum of all of the $\Delta p_i \in \mathcal{M}$ will also be non-positive.

2) If $C_i > 0$, then, from (12), $C_i = \bar{C}_i$, and the sum of all $\Delta p_i \in \mathcal{S}$ can be simplified as follows:

$$\sum_{i\in\mathcal{S}} \Delta p_i = \Big(\frac{n+1}{2n}\alpha^2 - \alpha\Big)\sum_{i\in\mathcal{S}}\left[\bar{C}_i^2 - \frac{1}{n}\left(\sum_{j\in\mathcal{N}_i}\bar{C}_i\bar{C}_j\right)\right] \tag{26}$$

For the summation term in brackets, there will be $|\mathcal{S}| = s$ square terms and at most $\frac{sn}{n} = s$ non-zero pairwise terms, causing the expression to be strictly non-negative. Thus, (26) will be non-positive for any $\alpha \in (0, \frac{2n}{n+1})$.

Therefore, the sum of non-positive values results in the fact that (23) is always non-positive for any $\alpha \in (0, \frac{2n}{n+1})$.

We therefore must deal with the cases where Δp is zero and desynchronization has not yet been achieved. There are two such cases: 1) All change values are equal and positive ($C_1 = C_2 = \ldots = C_N > 0$), and 2) all change values are non-positive ($C_i \le 0\ \forall i$).

1) It is trivial to show that the first case cannot happen. If all change values are positive, then $\theta_i - \theta_{i-1} < \theta_{i+1} - \theta_i < \ldots < \theta_{i-1} - \theta_{i-2} < \theta_i - \theta_{i-1}$, which is a contradiction.
2) The second case can be true for one cycle since the nodes can sense only the "stale" phase from each node. However, this case will not be true for the next cycle. If no nodes move in a cycle, each node will then be observing the true phase of its connecting nodes, and not the stale values, in the next cycle. According to Lemma 2, $\sum_{i=1}^{N} C_i = 0$ holds in the next cycle. If desynchronization has not yet been achieved (i.e., not all change values are zero), then at least one node will have a positive change value, and will move forward in phase, causing the change in energy to be negative according to (23).

Therefore, the energy will always decrease and converge to zero. According to Proposition 2, the network will converge to the state of desynchronization. ∎

Remark 6: As determined from the derivation of (23), a choice of change ratio $\alpha \in (0, \frac{2n}{n+1})$ will ensure that the network energy will decrease with each cycle. An α larger than $\frac{2n}{n+1}$ cannot be used to guarantee that the network energy will converge to zero, and thus cause the network to desynchronize. Because we want to choose α such that it can guarantee desynchronization independent of network size, the largest range of α, then, that can ensure desynchronization is $\alpha \in (0, 1]$, since $\min_{n\in\mathbb{N}} \frac{2n}{n+1} = 1$.

IV. Circulant Symmetric Case

We now relax the constraint of an all-to-all topology. We will show that any circulant symmetric network will desynchronize under our algorithm, specifically for the circulant symmetric ring topology.

A. Convergence of the Ring Topology

Let us now consider the case of the circulant symmetric ring topology, where every node is connected to the node directly ahead of and behind it in phase.

Theorem 2: A network with the circulant symmetric ring topology, where every node is connected to the node directly ahead of and behind it in phase, under the proposed algorithm will converge to the desynchronized state.

Proof: We can prove Theorem 2 if we can show that $\Delta p(k) < 0$ for a general k. The proof follows a similar approach to Theorem 1.

Using the result from (16) in Theorem 1, and knowing that $n = 2$ for the ring topology, we can write an expression for the change value during the next cycle as follows:

$$C_i(k+1) = C_i(k) + \alpha\left(\frac{\bar{C}_{i-1}(k) + \bar{C}_{i+1}(k)}{2} - \bar{C}_i(k)\right) \tag{27}$$

Squaring both sides, subtracting $C_i(k)^2$ from both sides, and using the fact that $C_i(k)\bar{C}_i(k) = \bar{C}_i(k)^2$ give us the term in the summation of (7):

$$\begin{aligned} C_i(k+1)^2 - C_i(k)^2 &\\ = \alpha\big(C_i(k)\bar{C}_{i-1}(k) &+ C_i(k)\bar{C}_{i+1}(k) - 2\bar{C}_i(k)^2\big)\\ &+ \frac{\alpha^2}{4}\big(\bar{C}_{i-1}(k) + \bar{C}_{i+1}(k) - 2\bar{C}_i(k)\big)^2 \end{aligned} \tag{28}$$

Substituting (28) into (7) and then splitting the summation give us the following relationship, where we drop the cycle (k) notation for clarity:

$$\begin{aligned} \Delta p = \frac{1}{2}\Bigg(\alpha\sum_{i=1}^{N}\big(C_i\bar{C}_{i-1} + C_i\bar{C}_{i+1} - 2\bar{C}_i^2\big)&\\ + \frac{\alpha^2}{4}\sum_{i=1}^{N}\big(\bar{C}_{i-1} + \bar{C}_{i+1} - 2\bar{C}_i\big)^2\Bigg)& \end{aligned} \tag{29}$$

Through expanding and recombining, the first summation term in (29) can be simplified, resulting in a convenient shifting of indices as follows:

$$\begin{aligned} &\alpha\sum_{i=1}^{N}\big(C_i\bar{C}_{i-1} + C_i\bar{C}_{i+1} - 2\bar{C}_i^2\big)\\ &= -2\alpha\left(\sum_{i=1}^{N}\left[\bar{C}_i^2 - \frac{1}{2}(C_i\bar{C}_{i+1} + C_{i+1}\bar{C}_i)\right]\right) \end{aligned} \tag{30}$$

In a similar way, we can expand and recombine the second summation term in (29), also resulting in a convenient shifting of indices:

$$\begin{aligned} &\frac{\alpha^2}{4}\sum_{i=1}^{N}\big(\bar{C}_{i-1} + \bar{C}_{i+1} - 2\bar{C}_i\big)^2\\ &= \frac{\alpha^2}{4}\sum_{i=1}^{N}\big(\bar{C}_{i-1}^2 + \bar{C}_{i+1}^2 + 4\bar{C}_i^2\\ &- 4\bar{C}_i\bar{C}_{i-1} - 4\bar{C}_i\bar{C}_{i+1} + 2\bar{C}_{i-1}\bar{C}_{i+1}\big)\\ &= \frac{\alpha^2}{4}\left(6\sum_{i=1}^{N}[\bar{C}_i^2] - 8\sum_{i=1}^{N}[\bar{C}_i\bar{C}_{i+1}] + 2\sum_{i=1}^{N}[\bar{C}_i\bar{C}_{i+2}]\right) \end{aligned} \tag{31}$$

By adding zero in a clever way, we can further simplify (31) as follows:

$$\frac{\alpha^2}{4}\left(6\sum_{i=1}^{N}[\bar{C}_i^2] - 8\sum_{i=1}^{N}[\bar{C}_i\bar{C}_{i+1}] + 2\sum_{i=1}^{N}[\bar{C}_i\bar{C}_{i+2}]\right)$$
$$+\ \frac{\alpha^2}{4}\left(2\sum_{i=1}^{N}[\bar{C}_i^2] - 2\sum_{i=1}^{N}[\bar{C}_i^2]\right)$$
$$= 2\alpha^2\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+1}]\right) - \frac{\alpha^2}{2}\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+2}]\right) \tag{32}$$

We can now substitute (30) and (32) into (29):

$$\Delta p = \Bigg(-\alpha\left[\sum_{i=1}^{N}\left(\bar{C}_i^2 - \frac{1}{2}[C_i\bar{C}_{i+1} + C_{i+1}\bar{C}_i]\right)\right]$$
$$+\ \alpha^2\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+1}]\right) - \frac{\alpha^2}{4}\left(\sum_{i=1}^{N}[\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+2}]\right)\Bigg) \tag{33}$$

Further simplification gives us the following final result:

$$\Delta p = (\alpha^2 - \alpha)\left[\sum_{i=1}^{N}(\bar{C}_i^2)\right] - \frac{\alpha^2}{4}\left[\sum_{i=1}^{N}(\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+2})\right]$$
$$- \sum_{i=1}^{N}\left(\alpha^2[\bar{C}_i\bar{C}_{i+1}] - \alpha\left[\frac{1}{2}(C_i\bar{C}_{i+1} + C_{i+1}\bar{C}_i)\right]\right) \tag{34}$$

It now leaves to show that (34) is always non-positive. We first notice that the first term is non-positive for $\alpha \in (0, 1)$, and the second term is non-positive for any positive α. These results now leave to show that the third term causes the entire expression to always be non-positive.

We will consider the following three cases separately: 1) Both C_i and C_{i+1} are positive, 2) one is positive, and the other non-positive, and 3) both are non-positive.

1) In the first case, the terms in brackets are equivalent, allowing us to combine the first and third term of (34):

$$(\alpha^2 - \alpha)\left[\sum_{i=1}^{N}(\bar{C}_i^2)\right] - \sum_{i=1}^{N}\left(\alpha^2[\bar{C}_i\bar{C}_{i+1}] - \alpha[\bar{C}_i\bar{C}_{i+1}]\right)$$
$$= (\alpha^2 - \alpha)\left[\sum_{i=1}^{N}(\bar{C}_i^2 - \bar{C}_i\bar{C}_{i+1})\right] \tag{35}$$

Eqn. (35) is non-positive for $\alpha \in (0, 1)$.

2) In the second case, the first term in brackets is zero, and the second term in brackets is non-positive. Thus, the whole term will be non-positive for any positive α.
3) In the third case, both terms in brackets are zero, making the whole term zero, and thus non-positive.

We next deal with the cases where Δp is zero and desynchronization has not yet been achieved. There are two such cases: 1) All change values are equal and positive ($C_1 = C_2 = \ldots = C_N > 0$), and 2) all change values are non-positive ($C_i \le 0\,\forall i$). These are the same cases as for the all-to-all topology.

1) It is trivial to show that the first case cannot happen. If all change values are positive, then $\theta_i - \theta_{i-1} < \theta_{i+1} - \theta_i < \ldots < \theta_{i-1} - \theta_{i-2} < \theta_i - \theta_{i-1}$, which is a contradiction.
2) The second case can be true for one cycle since the nodes can sense only the "stale" phase from each node. However, this case will not be true for the next cycle. If no nodes move in a cycle, each node will then be observing the true phase of its connecting nodes, and not the stale values, in the next cycle. According to Lemma 2, $\sum_{i=1}^{N} C_i = 0$ holds in the next cycle. If desynchronization has not yet been achieved (i.e., not all change values are zero), then at least one node will have a positive change value, and will move forward in phase, causing the change in energy to be negative according to (34).

Therefore, the energy will always decrease and converge to zero for a choice of $\alpha \in (0, 1]$. According to Proposition 2, the network will converge to the desynchronized state. ■

B. Convergence of Circulant Symmetric Topologies

We can further prove the convergence of any network that can be described with an adjacency matrix that is both circulant and symmetric to the desynchronized state.

Theorem 3: A network with any circulant symmetric topology under the proposed algorithm will converge to the desynchronized state.

Proof: The proof follows an identical structure as the proof for Theorem 2, and hence will be omitted here, due to tedious bookkeeping. Given the associated adjacency matrix for the network, we can derive the change in energy after each cycle. It will be non-positive for $\alpha \in (0, 1]$, and cannot stay at zero unless the network is desynchronized. Thus, the network energy will approach zero, and, according to Proposition 2, the network will converge to the state of desynchronization. ■

Remark 7: The definition of circulant symmetric effectively limits the region of initial conditions under which convergence can be achieved. Enforcing this condition on the initial phases for large networks may deserve separate investigation.

Remark 8: Other types of topologies may also desynchronize under our algorithm, but we cannot prove that from the theory presented in this paper. One would need to show that the splay state is the only equilibrium point for the network under the proposed algorithm, and that the energy for that network topology decreases with each firing cycle.

V. Simulation Results

We will now simulate how the network evolves over time using our proposed algorithm. We offer comparisons between the all-to-all topology and circulant symmetric ring topology. The results of these simulations confirm our theoretical results from the previous sections. All simulations were done using MATLAB, with all PCOs having identical fundamental frequency, $\omega = 2\pi$, and identical period of one second.

A. Proposed Algorithm Results

We consider a network of six PCOs, with randomly chosen initial phases from the interval $[0, 2\pi]$. For the circulant symmetric ring topology, we order the initial phases first, and then

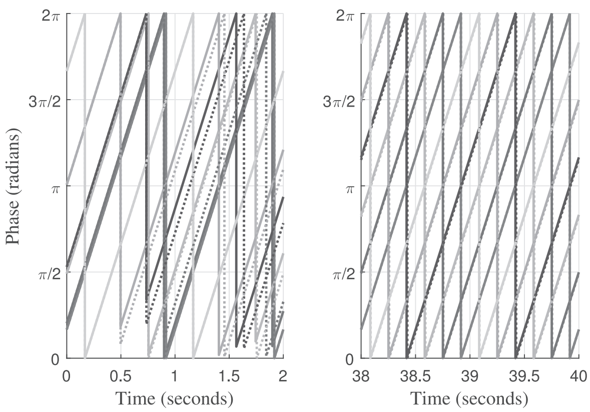


Fig. 5. Initial and final phase evolution for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict behavior under the all-to-all topology. Dotted lines depict behavior under the circulant symmetric ring topology. The network converges to the state of desynchronization.

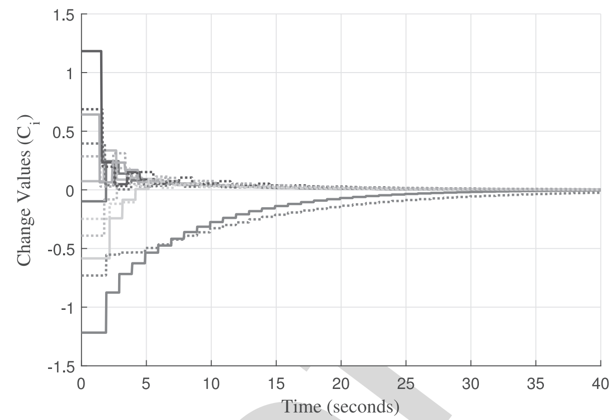


Fig. 7. The evolution of the change value C_i for each node. Solid lines depict behavior under the all-to-all topology. Dotted lines depict behavior under the circulant symmetric ring topology. All change values converge to zero.

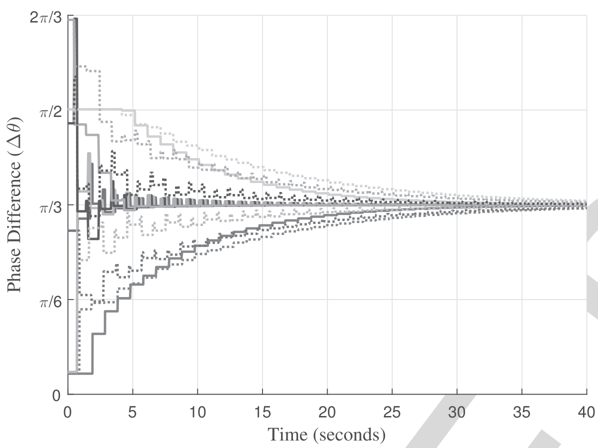


Fig. 6. Phase differences for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict values under the all-to-all topology. Dotted lines depict values under the circulant symmetric ring topology. The phase differences converge to the same value, one sixth of the unit circle distance, and all nodes become evenly spread out around the unit circle.

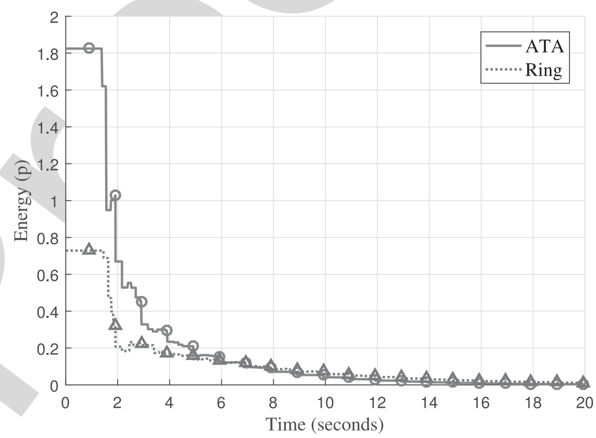


Fig. 8. Network energy as a function of time for six nodes with random initial state and $\alpha = 0.9$ for both the all-to-all (ATA) and circulant symmetric ring topologies. Markers indicate the measured energy after a complete firing cycle. The energy converges to zero as the network desynchronizes.

create the network such that every node is connected to the node directly ahead of and behind it in phase.

The evolution of the phases under our algorithm in the all-to-all topology and the circulant symmetric ring topology is given in Fig. 5. To visualize the evolution of the network toward desynchronization more easily, we also measure the phase difference between adjacent nodes, which we define in the following manner:

$$\Delta\theta_i = (\theta_{i+1} - \theta_i) \mod 2\pi, \text{ where } \theta_{N+1} = \theta_1 \tag{36}$$

Fig. 6 shows the plot of this phase distance between adjacent nodes, and confirms that desynchronization is indeed achieved.

In Fig. 7, we plot the change values for each node, which converge to zero with time. This behavior is to be expected, since once the network desynchronizes, no more jumps in phase need to be made.

In Fig. 8, the network energy as defined in (5) is shown for both the all-to-all and ring topology. It is important to note that the energy is not always strictly decreasing with regard to time. This result is primarily caused by the asymmetric nature of the update algorithm and secondarily due to the restriction that only forward jumps in phase can be made. However, the energy does strictly decrease while observing the value after every firing cycle, as was determined in our theoretical derivation. This result and our theoretical derivation show the importance of analyzing the evolution of the energy based on complete firing cycles instead of individual firing instants. Our treatment of desynchronization in this paper is different than the treatment of pulse-based desynchronization in the literature, which analyzes convergence to the splay state based on individual firing instances and will fail if applied directly to analyzing desynchronization in this paper.

As shown in Fig. 9, the rate at which the network energy converges is shown to be dependent on the network topology. More node connections cause the network to desynchronize more quickly than with fewer connections.

It is also important to note that the initial network energies under different network topologies are different. This difference is due to the increased number of connections in the all-to-all topology, making comparisons between the energy of the network and the closeness to desynchronization more difficult for

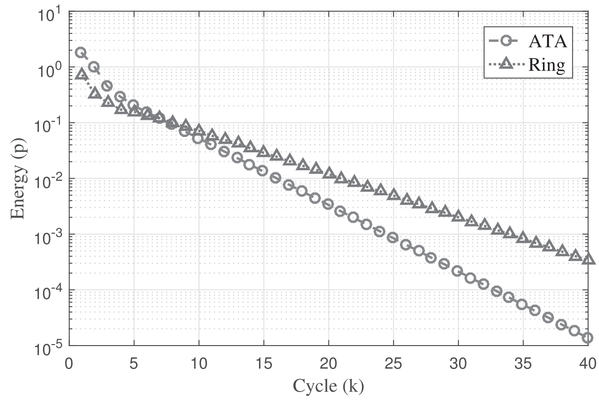


Fig. 9. Network energy after a complete firing cycle for six nodes with random initial state and $\alpha = 0.9$ for both the all-to-all (ATA) and circulant symmetric ring topologies. The energy strictly decreases when measured after a complete firing cycle. The energy convergence rate depends on the network topology.

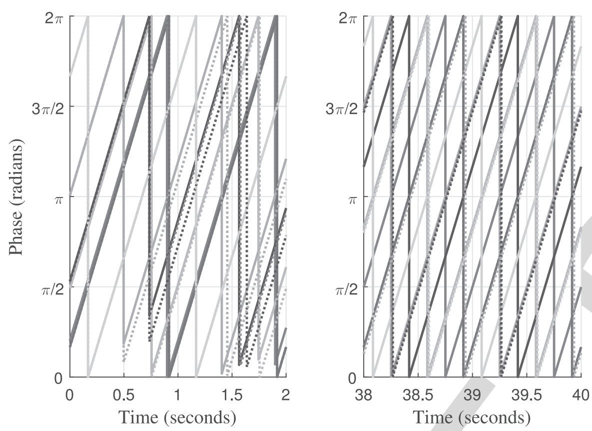


Fig. 10. Initial and final phase evolution for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict behavior under the all-to-all topology. Dotted lines depict behavior under the circulant symmetric ring topology. When unordered, the nodes in the circulant symmetric ring topology converge into groups of two, rather than spreading out evenly. The all-to-all topology is not affected by the reordering of the nodes.

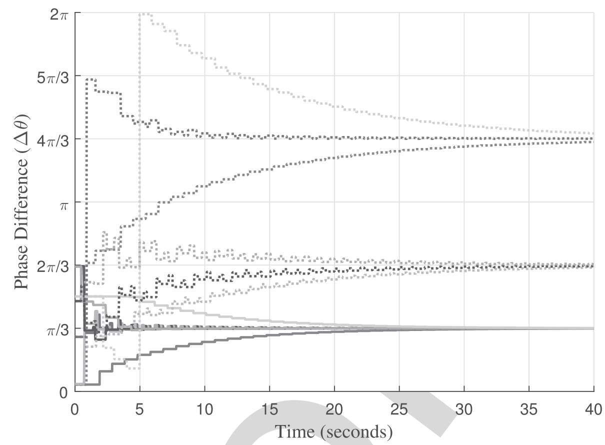


Fig. 11. Phase differences for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict values under the all-to-all topology. Dotted lines depict values under the circulant symmetric ring topology. The ring topology does not properly desynchronize or maintain the initial firing sequence when the nodes are unordered, and the resulting adjacency matrix is non-circulant.

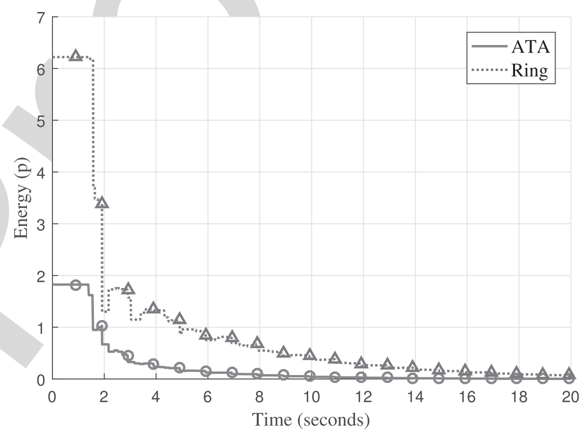


Fig. 12. Network energy as a function of time for six nodes with random initial state and $\alpha = 0.9$ for both the all-to-all (ATA) and ring topologies. Markers indicate the measured energy after a complete firing cycle. The energy of the ring topology still approaches zero, but does not indicate desynchronization, since the ordered adjacency matrix is non-circulant.

networks of different topologies and sizes. Normalizing the energy curves to have the same initial value aids in this comparison.

B. Example of Insufficient Initial Condition

Although our proposed algorithm relaxes the constraint of an all-to-all network topology, our theory only expands the set of potential network topologies to circulant symmetric networks with ordered nodes. If the network does not satisfy the conditions of being circulant or symmetric when the nodes are ordered, then our theoretical results do not guarantee that the network will desynchronize.

As an example, we take the initial node phases from the previous example and swap the phases of node 1 and node 2, keeping the same network connections from before. The all-to-all topology, with the swapped nodes, still has a circulant symmetric adjacency matrix, and thus meets the conditions for our proposed algorithm. However, the resulting node ordering for the circulant symmetric ring topology causes the network to have a symmetric, but non-circulant, adjacency matrix.

As seen in Fig. 10, the nodes in the all-to-all topology achieve phase desynchronization. However, the nodes of the ring topology do not achieve phase desynchronization, but rather converge into groups of two, and do not maintain the initial firing sequence. Fig. 11 more apparently shows the effect of the ring topology not achieving desynchronization, and also shows the shifting of relative node positions.

Even though the network energy approaches zero, as shown in Fig. 12, the required initial conditions were not met for this topology, and the energy does not indicate desynchronization.

C. Comparison with DESYNC-STALE [7]

Our proposed algorithm can be thought of as a variant and generalization of the DESYNC-STALE algorithm first given in [7]. The primary advantage that our proposed algorithm has over the DESYNC-STALE algorithm is the relaxation of the all-to-all topology constraint. Whereas the DESYNC-STALE algorithm assumes an all-to-all topology, our proposed algo-

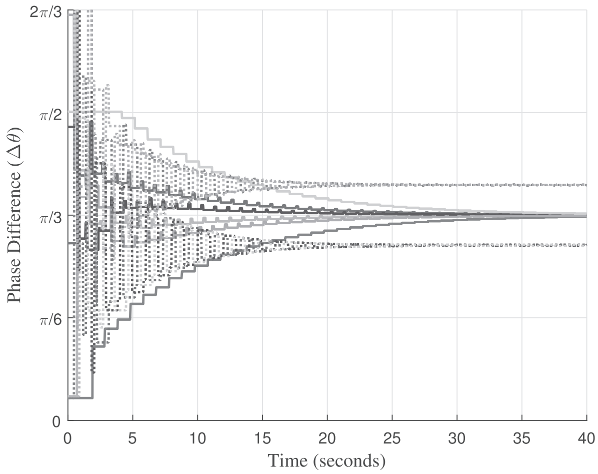


Fig. 13. Phase Differences for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict values under our proposed algorithm. Dotted lines depict values under the DESYNC-STALE algorithm. Both algorithms use the circulant symmetric network topology shown in Fig. 3b. The DESYNC-STALE algorithm is unable to completely desynchronize.

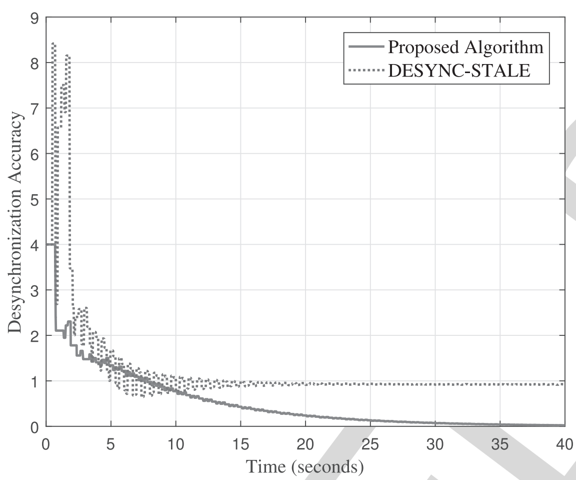


Fig. 14. Desynchronization accuracy of the network as defined in [7] as a function of time for six nodes with random initial state and $\alpha = 0.9$. Solid lines depict behavior under our proposed algorithm. Dotted lines depict behavior under the DESYNC-STALE algorithm. The DESYNC-STALE algorithm fails to achieve network desynchronization.

rithm can generalize to any circulant symmetric network. Our proposed algorithm thus allows for a wider range of network topologies that can achieve desynchronization.

To illustrate the advantages of our proposed algorithm, we use the network topology shown in Fig. 3b, and use both algorithms to try to achieve the state of desynchronization. In the DESYNC-STALE algorithm, a node moves based on the nearest phase neighbor directly ahead of and behind it, whereas in our proposed algorithm the node moves based on all connected nodes.

In Fig. 13, we see that our proposed algorithm achieves evenly spaced node phases, whereas the DESYNC-STALE algorithm is unable to achieve desynchronization, as a result of each node ignoring the phase of one node to which it is connected.

Fig. 14 shows that our algorithm is able to accurately desynchronize under the given network topology, but the DESYNC-STALE algorithm is unable to accurately desynchronize. The

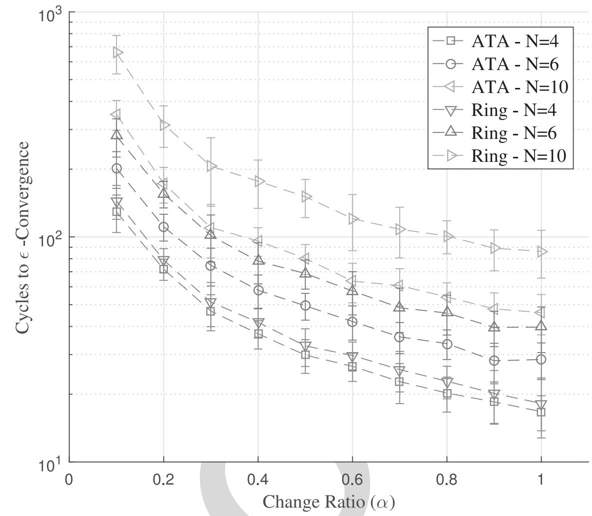


Fig. 15. The convergence rate of the network toward desynchronization for both the all-to-all (ATA) and circulant symmetric ring topologies. Points represent the mean of 25 runs at random initial conditions for each change ratio to reach ϵ-desynchrony for $\epsilon = 1 * 10^{-4}$. Error bars show one standard deviation about the mean.

desynchronization accuracy of the network $|\vec{\Delta}|$, defined in [7], is given to be "the sum of the absolute deviations from perfect desynchrony."

$$|\vec{\Delta}| = \sum_{i=1}^{N} \left| \Delta\theta_i - \frac{2\pi}{N} \right| \tag{37}$$

D. Proposed Algorithm Convergence

We now evaluate the speed of convergence of our algorithm for various sizes and types of topologies. To do so, we need some measure for when the network has desynchronized. We say that a network is ϵ-desynchronized after k cycles when the network energy satisfies $p(k) < \epsilon$.

In Fig. 15, we compare the relative convergence speed for various network topologies and sizes under different change ratios. We run 25 simulations at random initial conditions for each set of parameters, and plot the mean and standard deviation for when the network becomes ϵ-desynchronized. We see that more node connections cause the network to desynchronize more quickly for a given change ratio. Also, larger sizes of networks take more time to desynchronize. It is interesting to note that the ratio between the mean amount of cycles to ϵ-desynchronization for identical network size is approximately constant with regard to the change ratio.

VI. Conclusion

In this paper we present a novel control algorithm for achieving phase desynchronization in pulse coupled oscillators. Our proposed algorithm is robust, allowing the control of the network to be independent of the number of nodes in the network. Having the control independent of the size of the network allows for the network to add and remove nodes freely without modifying the update strategy. This robustness presents itself useful in many applications, where the network may have nodes dynamically entering and leaving the network.

It is our belief that this paper is the first to present a proof for the convergence to the desynchronized state for a pulse coupled oscillator network topology other than the all-to-all network. Specifically, we show that the network, under our proposed dynamics, will converge to the state of uniform phase desynchronization for any circulant symmetric network with ordered nodes, which includes the all-to-all and circulant symmetric ring topologies as special cases. Relaxing the constraint of an all-to-all topology makes our proposed algorithm desirable for various applications, where direct communication may not be available between every pair of nodes in the network.

Further analysis can still be made. Specifically, theoretically analyzing the convergence speed of the network is desirable so that the network can be tuned by an appropriate choice of topology and change ratio. This analysis will be complex, since it will be dependent on the type of network topology and initial conditions of the nodes. Also, further research can be done to see if additional types of networks can desynchronize under the proposed algorithm beyond the class of circulant symmetric networks.

Appendix

Proof for Lemma 1:

Proof: To prove this lemma, we need only to show that the firing sequence is invariant for one node, and then it will be true for all other nodes. To show the firing invariance for one node, we need only to show that the node will not pass the node directly ahead of it in phase within a firing cycle, regardless of whether the two nodes are connected.

Without loss of generality, let us consider node N, and show that it will not pass node 1 in phase during a firing cycle. We can describe this relationship mathematically with the following equation:

$$\alpha C_1 + \big((\theta_1 - \theta_N) \mod 2\pi\big) > \alpha C_N \tag{38}$$

We need only to consider the case when node N jumps forward in phase ($C_N > 0$), since the node will not pass the node directly in front of it if it does not jump. Without loss of generality, let us assume that the network has evolved to the instant right before node N fires ($\theta_N = 2\pi$). Thus, (38) can be rewritten as follows:

$$(\alpha C_1 + \theta_1) > \alpha C_N \tag{39}$$

We can express the amount that node N will jump when it fires as a function of the phases connected to it,

$$\begin{aligned} C_N &= \frac{1}{n}\sum_{l\in\mathcal{N}_N}\Big(-\big(2\pi - \big((\theta_l - \theta_N) \mod 2\pi\big)\big) + \pi\Big) \\ &= \frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_l\big) - \pi \end{aligned} \tag{40}$$

where $l \in \mathcal{N}_N$ corresponds to the node that caused the m^{th} instance of node N's recorded phase.

Similarly, we can express the amount that node 1 will jump,

$$\begin{aligned} C_1 &= \frac{1}{n}\sum_{j\in\mathcal{N}_1}\Big(-\big(2\pi - \big((\theta_j - \theta_1) \mod 2\pi\big)\big) + \pi\Big) \\ &= \frac{1}{n}\sum_{j\in\mathcal{N}_1}\big(\theta_j\big) - \theta_1 - \pi \end{aligned} \tag{41}$$

where $j \in \mathcal{N}_1$ corresponds to the node that caused the m^{th} instance of node 1's recorded phase.

Since the nodes are ordered and the network is circulant, node 1 is connected to the nodes that are directly ahead in phase of the nodes that are connected to node N. Thus, we can express (41) in terms of the neighbors of node N:

$$C_1 = \frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_{l+1}\big) - \theta_1 - \pi \tag{42}$$

By substituting (40) and (42) into (39), we achieve the following relationship:

$$\begin{aligned} &\alpha\left(\frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_{l+1}\big) - \theta_1 - \pi\right) + \theta_1 > \alpha\left(\frac{1}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_l\big) - \pi\right) \\ &\quad\Rightarrow \frac{\alpha}{n}\sum_{l\in\mathcal{N}_N}\big(\theta_{l+1}\big) + (1-\alpha)\theta_1 > \frac{\alpha}{n}\sum_{l\in\mathcal{N}_N}\theta_l \end{aligned} \tag{43}$$

Since the nodes are ordered, then we know that $\theta_{l+1} > \theta_l \ \forall l \in \mathcal{N}_N$. Thus, for any $\alpha \in (0, 1]$, the expression in (43) will be true, and after a cycle the relative node positions and the resulting firing sequence will be maintained.

An issue arises if $\alpha C_N > \theta_1$ holds when node N fires for the first time. When this condition is true, then, due to the asynchronous update of the algorithm, node N will fire again before node 1, breaking the invariance of the firing sequence. Even though the amount that node 1 will change would put it past node N again, node N will fire first. However, this problem does not occur in practice where the nodes begin with no knowledge of the positions of neighboring nodes in the network. Each node senses the location of neighboring nodes based on received pulses as the network evolves. Specifically, node N has the largest initial phase and will fire first. When it fires, it will not have received any pulses from its neighboring nodes, and thus it has $C_N = 0$, and will not jump. As the network evolves, node $(N-1)$ will fire next. When it fires, it will only have received a pulse from node N if $N \in \mathcal{N}_{N-1}$, and, from the phase response function in (1), any jump that node $(N-1)$ makes will not break invariance. A similar result occurs for node $(N-2)$ to node 2. When node 1 fires, it will have full knowledge of the positions of all neighboring nodes, and from the analysis above, will jump and maintain firing invariance. Thus, when node N fires the second time, Eq. (39) holds, and invariance is maintained. ■

Proof for Lemma 2:

Proof: From (1) and (2), we can write that

$$\begin{aligned} n_i \sum_{i=1}^{N} C_i &= \sum_{i=1}^{N} \sum_{m=1}^{n_i} Q(\theta_{i,m}) \\ &= \sum_{i=1}^{N} \sum_{l \in \mathcal{N}_i} \big(-[(\theta_i - \theta_l) \mod 2\pi] + \pi \big) \qquad (44) \end{aligned}$$

where $l \in \mathcal{N}_i$ corresponds to the node that caused the m^{th} instance of node i's recorded phase.

Since the network is symmetric, then for two nodes j and k with a connection between them (assuming $j > k$), node j will record the m_1^{th} instance of its phase (denoted θ_{j,m_1}) when node k fires, and node k will record the m_2^{th} instance of its phase (denoted θ_{k,m_2}) when node j fires. Note that θ_{j,m_1} represents the phase difference between nodes k and j, and θ_{k,m_2} represents the phase difference between nodes j and k. Since no nodes have jumped in phase, the phase difference measured by these two nodes is the true phase difference between them and is equivalent for both nodes. Combining these two terms causes them to cancel.

$$\begin{aligned} &Q(\theta_{j,m_1}) + Q(\theta_{k,m_2}) \\ &= \big(-(\theta_j - \theta_k) + \pi \big) + \big(-(\theta_k - \theta_j + 2\pi) + \pi \big) = 0 \quad (45) \end{aligned}$$

Since the entire network is symmetric, and no nodes have jumped in the cycle, the whole expression can be recombined and canceled.

$$n_i \sum_{i=1}^{N} C_i = 0 \Rightarrow \sum_{i=1}^{N} C_i = 0 \qquad (46)$$

Thus, the desired result is achieved. ■

ACKNOWLEDGMENT

The authors would like to thank those who reviewed the initial drafts of this paper.

REFERENCES

[1] R. Mirollo and S. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, 1990.

[2] P. Goel and B. Ermentrout, "Synchrony, stability, and firing patterns in pulse-coupled oscillators," *Phys. D, Nonlinear Phenom.*, vol. 163, no. 3, pp. 191–216, 2002.

[3] C. S. Peskin, *Mathematical Aspects of Heart Physiology*. New York, NY, USA: Courant Inst. of Math. Sci., New York Univ., New York University, 1975.

[4] B. Ermentrout, "Type I membranes, phase resetting curves, and synchrony," *Neural Comput.*, vol. 8, no. 5, pp. 979–1001, 1996.

[5] H. Gao and Y. Q. Wang, "Integrated communication and control design for decentralized collective motion coordination," Submitted.

[6] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion: All-to-all communication," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 811–824, May 2007.

[7] A. Patel, J. Degesys, and R. Nagpal, "Desynchronization: The theory of self-organizing algorithms for round-robin scheduling," in *Proc. 1st Int. Conf. Self-Adapt. Self-Organizing Syst.*, Cambridge, MA, USA, Jul. 2007, pp. 87–96.

[8] J. Degesys and R. Nagpal, "Towards desynchronization of multi-hop topologies," in *Proc. 2nd IEEE Int. Conf. Self-Adapt. Self-Organizing Syst.*, Oct. 2008, pp. 129–138.

[9] R. Pagliari, Y. W. P. Hong, and A. Scaglione, "Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 4, pp. 564–575, May 2010.

[10] F. Núñez, Y. Q. Wang, and F. J. Doyle III, "Global synchronization of pulse-coupled oscillators interacting on cycle graphs," *Automatica*, vol. 52, pp. 202–209, 2015.

[11] F. Núñez, Y. Q. Wang, and F. J. Doyle III, "Synchronization of pulse-coupled oscillators on (strongly) connected graphs," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1710–1715, Jun. 2015.

[12] F. Dörfler and F. Bullo, "Synchronization in complex networks of phase oscillators: A survey," *Automatica*, vol. 50, no. 6, pp. 1539–1564, 2014.

[13] Y. Q. Wang and F. J. Doyle III, "Optimal phase response functions for fast pulse-coupled synchronization in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5583–5588, Oct. 2012.

[14] Y. Q. Wang, F. Núñez, and F. J. Doyle III, "Energy-efficient pulse-coupled synchronization strategy design for wireless sensor networks through reduced idle listening," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5293–5306, Oct. 2012.

[15] S. Phillips and R. G. Sanfelice, "Results on the asymptotic stability properties of desynchronization in impulse-coupled oscillators," in *Proc. Amer. Control Conf.*, Washington, DC, USA, Jun. 2013, pp. 3272–3277.

[16] N. Deligiannis, J. F. C. Mota, G. Smart, and Y. Andreopoulos, "Decentralized multichannel medium access control: Viewing desynchronization as a convex optimization method," in *Proc. 14th Int. Conf. Inf. Process. Sensor Netw.*, New York, NY, USA, 2015, pp. 13–24.

[17] H. Gao and Y. Q. Wang, "Analysis and design of phase desynchronization in pulse-coupled oscillators." (2016). [Online]. Available: http://arxiv.org/pdf/1603.03313v1.pdf

[18] R. Sepulchre, D. A. Paley, and N. E. Leonard, "Stabilization of planar collective motion with limited communication," *IEEE Trans. Autom. Control*, vol. 53, no. 3, pp. 706–719, Apr. 2008.

[19] R. Gentz, A. Scaglione, L. Ferrari, and Y. W. P. Hong, "Pulsess: A pulse-coupled synchronization and scheduling protocol for clustered wireless sensor networks," *IEEE Internet Things J.*, in press, 2016.

[20] P. Taechalertpaisarn, S. Choochaisri, and C. Intanagonwiwat, "An orthodontics-inspired desynchronization algorithm for wireless sensor networks," in *Proc. 2011 IEEE 13th Int. Conf. Commun. Technol.*, Sep. 2011, pp. 631–636.

[21] D. Buranapanichkit, N. Deligiannis, and Y. Andreopoulos, "Convergence of desynchronization primitives in wireless sensor networks: A stochastic modeling approach," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 221–233, Jan. 2015.

[22] H. K. Khalil, *Nonlinear Systems*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

Timothy Anglea (S'14) received the B.S. degree in engineering from Bob Jones University, Greenville, SC, USA, in 2015. He is currently working toward the M.S. degree in electrical engineering in the Electrical and Computer Engineering Department, Clemson University, Clemson, SC, USA, with a focus in intelligent systems.

He received first place in the 2015 IEEE SouthEastCon student paper competition with research on a simple control strategy for autonomous vehicles. His current research focuses on the dynamics and applications of pulse-coupled oscillators.

Yongqiang Wang (SM'13) was born in Shandong, China. He received the B.S. degree in electrical engineering and automation and the B.S. degree in computer science and technology from Xi'an Jiaotong University, Shaanxi, China, in 2004. He received the M.Sc. and Ph.D. degrees in control science and engineering from Tsinghua University, Beijing, China, in 2009.

From 2007 to 2008, he was with the University of Duisburg-Essen, Duisburg, Germany, as a Visiting Student. He was a Project Scientist at the University of California, Santa Barbara. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA. His research interests include cooperative and networked control, synchronization of wireless sensor networks, systems modeling and analysis of biochemical oscillator networks, and model-based fault diagnosis. He received the 2008 Young Author Prize from IFAC Japan Foundation for a paper presented at the 17th IFAC World Congress in Seoul.