

Convergence of Desynchronization Primitives in Wireless Sensor Networks: A Stochastic Modeling Approach

Dujdow Buranapanichkit, Nikos Deligiannis, *Member, IEEE*, and Yiannis Andreopoulos*, *Senior Member, IEEE*

Abstract—Desynchronization approaches in wireless sensor networks converge to time-division multiple access (TDMA) of the shared medium without requiring clock synchronization amongst the wireless sensors, or indeed the presence of a central (coordinator) node. All such methods are based on the principle of *reactive listening* of periodic “fire” or “pulse” broadcasts: each node updates the time of its fire message broadcasts based on received fire messages from some of the remaining nodes sharing the given spectrum. In this paper, we present a novel framework to estimate the required iterations for convergence to fair TDMA scheduling. Our estimates are fundamentally different from previous conjectures or bounds found in the literature as, for the first time, convergence to TDMA is defined in a *stochastic sense*. Our analytic results apply to the DESYNC algorithm and to pulse-coupled oscillator algorithms with inhibitory coupling. The experimental evaluation via iMote2 TinyOS nodes (based on the IEEE 802.15.4 standard) as well as via computer simulations demonstrates that, for the vast majority of settings, our stochastic model is within one standard deviation from the experimentally-observed convergence iterations. The proposed estimates are thus shown to characterize the desynchronization convergence iterations significantly better than existing conjectures or bounds. Therefore, they contribute towards the analytic understanding of how a desynchronization-based system is expected to evolve from random initial conditions to the desynchronized steady state.

Index Terms—wireless sensor networks, desynchronization, stochastic modeling, pulse coupled oscillators, TDMA.

I. INTRODUCTION

EFFICIENT usage of shared spectrum in distributed (ad-hoc) networking architectures is important for high data-rate communications [1]. This is particularly so for wireless sensor networks (WSNs), where wasting packet transmissions due to collisions in the medium access also means wasting battery resources [1]–[8]. Desynchronization

is a new WSN primitive leading to fair time-division multiple access (TDMA) scheduling that does not require the presence of a coordinating node [3], [4], [6], [7], [9]–[13]. All desynchronization approaches are based on the principle of *reactive listening*, where nodes periodically broadcast short packets (so-called “beacon” or “fire” messages [4], [6], [7], [9]–[12], [14]) and then update their next broadcast time based on the reception of fire messages from some of the remaining nodes. These methods make use of a convergence interval, where nodes adjust their firing times, and a *steady-state* (SState) period. In the SState period, nodes have converged into fair TDMA scheduling and fire messages are sent by each node in regular (periodic) intervals of T seconds, followed by data packets.

Historically, biology-inspired synchronization and desynchronization algorithms emerged from pioneering work in *pulsed-coupled oscillators* (PCOs) [15] and integrate-and-fire models [16]–[18]. Since the original formulation of desynchronization within the context of WSNs [3], [4], several authors extended properties of its basic reactive listening primitive in a number of ways. Extensions towards multihop or complex network topologies [19] have been proposed via: (i) including neighboring information in fire messages [20], (ii) low-complex graph theory methods [21], [22], and (iii) broadcast/reception of only a limited number of beacon messages to/from the immediate phase neighbors [5], [23]–[25]. The effects of node mobility in desynchronization were discussed in recent work [26]. Synchronization and desynchronization methods with limited listening or limited beacon broadcasts were also proposed recently for increased energy efficiency in WSN designs [19], [27]–[30]. Under the knowledge of the total number of nodes, it was shown that maintaining one node with fixed beaconing (i.e., an “anchored” node) [12] allows for faster convergence to TDMA. Other works focused on modifications to the basic desynchronization to allow for TDMA with: low-complexity scheduling [31], unequal slot sizes [3], [14], as well as scheduling under discrete resources (non-continuous time) [10]. Finally, in our recent work [32] we proposed a time-frequency extension of the desynchronization process in order to achieve increased bandwidth efficiency and allow for low-complex distributed coordination across the multiple channels supported by the IEEE 802.15.4 standard for WSN communications.

In all these works, the number of convergence itera-

*Corresponding author.

DB is with the Department of Electrical Engineering, Faculty of Engineering, Prince of Songkla University, Hat Yai, Songkla, 90112 Thailand; Tel. +66 74 287244, Fax +66 74 459395, Email: dujdow.b@psu.ac.th.

ND and YA are with the Electronic and Electrical Engineering Department, University College London, Roberts Building, Torrington Place, London, WC1E 7JE, U.K.; Tel. +44 20 7679 7303, Fax. +44 20 7388 9325 (both authors), Email: {n.deligiannis, i.andreopoulos}@ucl.ac.uk. This work was supported in part by the EPSRC, grant EP/K033166/1.

Copyright (c) 2014 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to permissions@ieee.org.

tions required until the steady state plays a crucial role in latency, energy and bandwidth efficiency of WSN deployments based on desynchronization. For example, the required convergence iterations were a key issue in the simulations and experiments of several desynchronization-based systems [3], [4], [10], [12], [20]. Beyond single-channel desynchronization, convergence iterations comprise a crucial parameter in the convergence delay of multichannel desynchronization [32], which is an important factor in the energy consumption of practical deployments [33]. Finally, deriving estimates for the convergence iterations forms a crucial step in the analytic understanding of how the system evolves from random initial conditions to the desynchronized steady state [12].

A. Related Work

It is well known that deriving *closed-form* estimates for the required convergence iterations to SState is hard [2], [4], [9], [12] due to the non-deterministic aspects of the desynchronization process, namely, the random initial condition of the phase of each node and the random perturbations in the firing order of nodes due to noise. Therefore, existing works focus on *order-of-convergence* [4] or *lower bounds of convergence iterations* of desynchronization, proven or conjectured via experimentation [3], [9], [10], [12]. Moreover, these works consider only the noise-free case.

While order-of-convergence estimates provide a coarse (asymptotic) characterization of the convergence, they do not predict the expected number of iterations required for desynchronization to converge to the SState. On the other hand, the existing *lower bounds* on the desynchronization convergence iterations are currently given without a characterization on their tightness to real-world experiments or simulations.

Instead, following a *stochastic approach* yields an analytic understanding of the behavior of the convergence of desynchronization. Particularly, it leads to analytic estimates for the desynchronization iterations that should be a close match to experiments and simulations. Recent work considered probabilistic approaches to analyze properties of synchronization [2], [24], [28]. However, no previous work proposes a framework for the analytic estimation of the expected convergence iterations required to achieve desynchronization. In addition, in all related work, the models are applicable to *synchronization*, and the required differences (i.e., different phase update, reachback response and pre-emptive message staggering [2] and limiting the node connectivity [24], [28]) do not permit a direct mapping of their experimentally-derived convergence estimates to desynchronization systems.

This gap in the analytic understanding of desynchronization is in fact explicitly recognized in the related literature [3], [10], [12], where it is stated that, although desynchronization algorithms are shown to work properly by various experiments and computer simulations, they still lack theoretical proofs for the expected iterations until convergence to SState.

B. Contribution

In this paper, we address this issue by embracing the non-deterministic aspects of desynchronization and proposing *stochastic* (instead of deterministic) estimates for the convergence iterations. In order for our estimates to have wide applicability, we focus on the two reactive listening primitives that form the basis of desynchronization algorithms with limited listening: (i) the DESYNC algorithm of Degeys *et al.* [4], [9]; (ii) PCOs with inhibitory coupling and limited listening proposed by Pagliari *et al.* [3]. In particular, Degeys *et al.* [4], [9] reduce the listening interval by considering only the temporally adjacent firing events of each node's firing and Pagliari *et al.* [3] limit the listening interval by introducing an appropriate PCO-dynamics function [1], [15]. This is of particular relevance to WSNs, because reductions in the listening interval correspond to substantial reductions in the energy consumption of wireless sensors [5], [28], [33].

If the total number of nodes is known, PCOs with inhibitory coupling and limited listening have been conjectured to converge to SState faster than the DESYNC algorithm [3]. Via the proposed stochastic modeling framework, we propose analytic estimates for the number of iterations until desynchronization is *expected to have converged* to SState within a predetermined threshold. We validate our results based on a real WSN deployment, as well as under a simulation environment, and demonstrate the superiority of the proposed stochastic estimates against the existing convergence bounds in the literature [3], [4], [10], [12].

C. Paper Organization

We first review the considered reactive listening primitives in Section II. The proposed stochastic estimates of the convergence iterations to SState are derived in Section III. Experimental results and comparisons are provided in Section IV. Section V presents results when using the proposed model within two WSN TDMA systems based on desynchronization, while Section VI provides concluding remarks.

II. DESYNCHRONIZATION PRIMITIVES

A. Notations and Symbolism

Italicized letters indicate scalars and boldface letters indicate vectors. For vectors \mathbf{a} and \mathbf{b} , the circular convolution [34] with period W is given by ($0 \leq n < W$)

$$(\mathbf{a} * \mathbf{b})_W [n] = \sum_{m=-\infty}^{\infty} \left(a[m] \sum_{k=-\infty}^{\infty} b[n - m - kW] \right)$$

Random variables (RVs) are represented by Greek uppercase letters, e.g., $\Phi \sim \mathcal{N}(\mu_\Phi, \sigma_\Phi)$ or $\Delta \sim \mathcal{U}(\mu_\Delta, \sigma_\Delta)$, with $\mathcal{N}(\cdot)$ and $\mathcal{U}(\cdot)$ reserved to indicate the normal and uniform probability density functions (PDFs), respectively, with mean μ_Φ (and μ_Δ) and standard deviation σ_Φ (and σ_Δ). The mathematical operators and key concepts used in the paper are summarized in Table I.

Table I
MATHEMATICAL OPERATORS AND KEY CONCEPTS.

Symbol	Definition
$\ \mathbf{v}\ $	vector norm-2
$\mathbf{v}[n]$	the n th element of vector \mathbf{v} , $n \geq 0$
$(\mathbf{a} * \mathbf{b})_W[n]$	the n th sample of circular convolution of period W
$\text{expr} \pmod{1}$	modulo-1 of expression $\text{expr} \in \mathbb{R}$
$a \leftarrow \text{expr}$	the value of variable a is updated via expression $\text{expr} \in \mathbb{R}$
$\lfloor u \rfloor$	the largest integer that is smaller or equal to u
$\lceil u \rceil$	the smallest integer that is larger or equal to u
$\text{Pr}[\text{expr}]$	probability of occurrence of expression $\text{expr} \in \mathbb{R}$
Key Concept	Explanation
W	total number of nodes in the desynchronization process
T	period of firing cycles (in seconds)
α	phase coupling constant of desynchronization
b_{thres}	steady-state convergence threshold of desynchronization
$\varphi_{\text{own}}^{(k)}$	phase variable of “own” node (i.e., the node under consideration) during its k th firing cycle
$\varphi_{-i}^{(k)}, \varphi_{+i}^{(k)}$	phase variables of the i th firing <i>prior to</i> or <i>after</i> “own” node’s firing in its k th firing cycle

B. Introduction to Desynchronization

Consider a WSN comprising W fully-meshed nodes, i.e., every WSN node can receive message broadcasts from all other nodes. Each node in the WSN is an “oscillator” that performs a task with a period of T seconds [4]. In the beginning, each node sets its internal timer to a random initial value between $[0, T)$. Upon the completion of its cycle, each node broadcasts a fire message and immediately resets its internal timer to zero. In the steady (desynchronized) state, each node fires every T seconds. For each node, its *firing cycle* comprises the time duration in-between two sequential fire message transmissions of its own. For each node, the percentage of the way through its k th firing cycle is denoted as the node’s *own firing phase* [2]–[4], [14], [20], [31], [32], $\varphi_{\text{own}}^{(k)} \in [0, 1)$. In order to achieve convergence to steady state, beyond its own phase, each node counts the percentage of time $t \in [0, T)$ from the moments it receives fire messages from other nodes and updates its own phase according to the desynchronization primitives detailed in the next two subsections. Each node listens for fire message broadcasts within a certain interval before (and possibly after) its own firing, which is termed as the *listening interval*.

Following the schema of Degeys *et al.* [4], the fire messages’ phase values can be imagined as beads moving clockwise on a ring with period $T = 1$ s (Fig. 1). When a node’s own firing phase reaches unity, i.e., top of Fig. 1, the node broadcasts a fire message and the node’s own firing phase is reset to zero. The figure illustrates a node’s own firing phase during its k th firing cycle and Fig. 1(a) shows the phase of the received fire messages from the two nodes

that fired immediately before and after it, denoted by $\varphi_{-1}^{(k)}$ and $\varphi_{+1}^{(k)}$, respectively¹. Thus, superscripts always indicate the *firing cycle* of “own” node in the WSN (i.e., of the node under consideration) and subscripts indicate the order relative to “own” node. The nodes corresponding to the previous and next firings of “own” node are called *phase neighbors* and the listening interval corresponding to the k th firing of “own” node is illustrated in Fig. 1. For all desynchronization algorithms under consideration [3], [4], [9], [12], [20], [23], [26], [32]:

- 1) the notion of phase neighbors indicates temporal adjacency of fire messages and is independent of the nodes’ physical location;
- 2) it is immaterial which physical sensor node is linked to which fire message, as desynchronization is solely dependent on the received fire message phase.

Therefore, the analysis of this paper is presented *from the viewpoint of any single node in the WSN* [20], [22], [25] and does not need to discern the specific firing order of all nodes in the network, which in fact may not be constant during the desynchronization convergence process.

C. DESYNC

In this approach, each node updates its firing phase *once* within each of its firing cycles, at the moment when the next fire message is received. As shown in Fig. 1(a), each node listens for the fire message preceding and following its own fire message broadcast. Hence, the duration of each node’s listening interval depends on the relative times in-between these two messages. It then performs its phase update during its k th firing cycle based on the phases of these two received messages, i.e., $\varphi_{-1}^{(k)}$ and $\varphi_{+1}^{(k)}$. Specifically, each node’s phase update moves its own firing phase towards the middle of the listening interval². The phase update of DESYNC during the k th firing cycle is expressed by [4], [9]

$$\varphi_{\text{own}}^{(k)} \leftarrow \left[(1 - \alpha) \varphi_{\text{own}}^{(k)} + \frac{\alpha}{2} (\varphi_{-1}^{(k)} + \varphi_{+1}^{(k)}) \pmod{1} \right], \quad (1)$$

where $\alpha \in (0, 1)$ denotes the *phase-coupling constant* that controls the speed of the phase adaptation. Previous work [4], [9] showed that the reactive listening primitive of (1) disperses all fire message broadcasts at intervals of $\frac{T}{W}$. Thus, after k_{SState} firing cycles, the DESYNC algorithm leads to fair TDMA scheduling, where all fire messages in the network are periodic and the phase update of (1) leads to convergence to SState, expressed by

$$\exists k_{\text{SState}} \in \mathbb{N}^* \text{ s.t. } \forall k \geq k_{\text{SState}} : \left| \varphi_{\text{own}}^{(k)} - \varphi_{-1}^{(k)} - \frac{1}{W} \right| \leq b_{\text{thres}}, \quad (2)$$

¹In this section, we ignore the phase measurement noise and assume each fire time can be determined precisely by all receiving nodes. This noise is however taken into account in the modeling framework.

²Since (1) is applied at the moment when the next firing is received, we have: $\varphi_{+1}^{(k)} = 0$, as seen in Fig. 1(a); however, we include $\varphi_{+1}^{(k)}$ in (1) to clarify that the operation of DESYNC depends on both the previous and next firing phase.

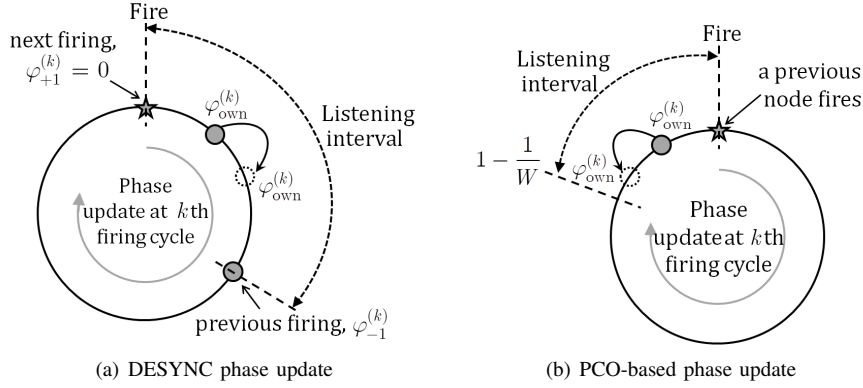


Figure 1. The phase update of “own” node (node under consideration) during its k th firing cycle when: (a) the next fire message is received in DESYNC, with the node’s listening interval defined as the time between the fire message preceding and following its own firing; (b) a fire message is received in PCO-based desynchronization while the node’s own phase is within its listening interval (i.e., when $1 - \frac{1}{W} < \varphi_{\text{own}}^{(k)} < 1$).

with b_{thres} the preset convergence threshold, typically [3], [4], [14], [20], [31], [32] $b_{\text{thres}} \in [0.001, 0.020]$. In a practical WSN deployment [4], [24], [25], [32], [33], each node would check that the condition of (2) holds for several consecutive firing cycles beyond $k = k_{\text{SState}}$ (e.g., five cycles) and would then declare that convergence has been achieved.

In steady state, each node transmits data packets for $T(\frac{1}{W} - b_{\text{thres}})$ seconds immediately following its fire message broadcast (which limits the maximum number of nodes supported under TDMA to less than $\lfloor \frac{1}{b_{\text{thres}}} \rfloor$). The only overhead stems from the fire message broadcasts, which are very short packets (just two bytes in our implementation).

Assuming negligible propagation delay and error-free detection of messages, it has been conjectured via simulations [3], [4] that convergence requires iterations of the order

$$k_{\text{DESYNC},[3][4]} \sim O\left(\frac{1}{\alpha} W^2 \ln \frac{1}{b_{\text{thres}}}\right). \quad (3)$$

Even though the order estimate (3) gives a coarse characterization for the convergence iterations, it cannot provide the expected number of iterations until convergence to SState is achieved. Moreover, real-world experimentation with TelosB and iMote2 motes [4], [32] under fixed α and b_{thres} values show that the measured number of iterations until convergence to SState is not proportional to W^2 .

D. PCO-based Inhibitory Coupling

PCO-based desynchronization with inhibitory coupling updates each node’s own firing phase according to the received fire messages within a certain interval of its firing cycle [3]. This is termed as the *listening interval* and its positioning within the k th firing cycle of a node is illustrated Fig. 1(b). As such: (i) the phase of the node’s own firing changes after a fire message from a previous phase-neighbor is received within the listening interval (i.e., unlike DESYNC, a varying number of phase updates may occur within *each* firing cycle); (ii) knowledge of the total number of nodes (W) is required [3]. Hence, a phase update during the k th firing cycle of PCO-based desynchronization

of a node [3] is performed at $\varphi_{\text{own}}^{(k)} \times T$ seconds after the node’s last fire message broadcast, $1 - \frac{1}{W} < \varphi_{\text{own}}^{(k)} < 1$:

$$\varphi_{\text{own}}^{(k)} \leftarrow \left[(1 - \alpha) \varphi_{\text{own}}^{(k)} + \alpha \left(1 - \frac{1}{W} \right) \pmod{1} \right], \quad (4)$$

where $\alpha \in (0, 1)$ the phase-coupling constant controlling the speed of the phase adaptation. All fire messages received outside the listening interval $(1 - \frac{1}{W}, 1)$ are simply ignored. After k_{SState} firing cycles, (4) has been shown to converge to dispersed fire message broadcasts received at intervals of $\frac{1}{W} \times T$ seconds [3], i.e., (2) holds under convergence to SState. Hence, like in the DESYNC case, once fair TDMA is achieved, the only overhead stems from the short fire message broadcasts.

Assuming negligible propagation delay, error-free detection of broadcast messages and that $1 - \frac{1}{W} > \alpha$, it has been shown [3] that the number of firing cycles for convergence is lower bounded by

$$k_{\text{PCO},[3]} \geq \left\lceil \frac{\ln b_{\text{thres}} - \ln \left[2 + \frac{2}{\alpha^W (1 - \alpha)} \right]}{\ln(1 - \alpha) + \ln W} \right\rceil. \quad (5)$$

Nevertheless, the tightness of (5) has neither been proven nor demonstrated via real-world experiments or simulation results.

III. PROPOSED STOCHASTIC MODELING OF DESYNC AND PCO-BASED DESYNCHRONIZATION WITH INHIBITORY COUPLING

Assumption 1 (Fully-meshed Topology): *We consider fully-meshed networks where each node can directly receive the fire message broadcasts from all other nodes.*

When performing the first phase update of (1) and (4), each node’s own firing phase, as well as the phases of all received fire messages from its phase neighbors, are modeled by independent random variables that are uniformly distributed in $[0, 1)$. This is formally stated in the following assumption.

Assumption 2 (Phase Model): *For every node under consideration (“own” node) and its phase neighbors, their*

firing phase during the first phase update is modeled by:
 $\forall i \in \{-1, \text{own}, +1\} : \Phi_i^{(1)} \sim \mathcal{P}_{\Phi_i^{(1)}}$, with³

$$\mathcal{P}_{\Phi_i^{(1)}} = \mathcal{U}(\mu_{\Phi_i^{(1)}}, \sigma_{\Phi_i^{(1)}}) \pmod{1}. \quad (6)$$

We define the mean times of successive phase updates to be equidistant, which, for the DESYNC update of (1), is expressed as

$$\mu_{\Phi_{-1}^{(1)}} - \mu_{\Phi_{\text{own}}^{(1)}} = \frac{1}{W}, \quad \mu_{\Phi_{\text{own}}^{(1)}} - \mu_{\Phi_{+1}^{(1)}} = \frac{1}{W} \quad (7)$$

and for the PCO update of (4) is stated by

$$\mu_{\Phi_{\text{own}}^{(1)}} = 1 - \frac{1}{W}. \quad (8)$$

In the beginning of the desynchronization process, all fire message broadcasts are completely uncoordinated (random), i.e., $\sigma_{\Phi_i^{(1)}} = \frac{1}{\sqrt{12}}$.

We remark that there is no loss of generality from the assumption of equidistant means of (7) and (8) as the modulo operator of (6) ensures that the PDFs wrap around one such that for every node and $\forall i \in \{-1, \text{own}, +1\} : \Phi_i^{(1)}$ are always uniformly-distributed between $[0, 1)$ irrespective of the assumed mean values. However, we opt for the use of (7) and (8) as this facilitates the mathematical exposition of the proposed estimates.

Our estimates of the convergence iterations for DESYNC and PCO-based desynchronization assume that each phase in (1) and (4) is contaminated by white noise due to the varying propagation, mobility and node processing delays of a WSN environment. This is captured in the following assumption.

Assumption 3 (Measurement Noise Model): All phase values in the update of (1) or (4) are contaminated by additive noise, modeled by an independent, zero-mean, uniformly-distributed, random variable, $\Delta \sim \mathcal{U}(0, \sigma_\Delta)$.

The assumption of uniform distribution for the measurement noise, as well as the value of the standard deviation σ_Δ will be derived experimentally, as they incorporate the effects of interference, wireless propagation and processing delays that can only be inferred via measurements from a real setup.

Due to the measurement noise and the interaction between fire message broadcasts, for each phase update of each node's firing phase, $\varphi_{\text{own}}^{(k)}$, during its k th firing cycle, the PDF of $\Phi_{\text{own}}^{(k)}$, $\mathcal{P}_{\Phi_{\text{own}}^{(k)}}$, changes after applying (1) or (4). Consequently, this changes the probability of convergence to SState, as follows:

$$\begin{aligned} \Pr \left[\left| \Phi_{\text{own}}^{(k)} - \mu_{\Phi_{\text{own}}^{(k)}} \right| \leq b_{\text{thres}} \right] &= \int_{-b_{\text{thres}}}^{b_{\text{thres}}} \mathcal{P}_{\Phi_{\text{own}}^{(k)}}(u - \mu_{\Phi_{\text{own}}^{(k)}}) du \\ &= \text{erf} \left(\frac{b_{\text{thres}}}{\sqrt{2} \sigma_{\Phi_{\text{own}}^{(k)}}} \right), \end{aligned} \quad (9)$$

³The use of the modulo operator in (6) is imposed because, by definition, we must ensure $\varphi_i \in [0, 1)$.

where $\text{erf}(u) = \frac{2}{\sqrt{\pi}} \int_0^u e^{-t^2} dt$ is the error function [35]. Notice that (9) holds under the assumption that $\mathcal{P}_{\Phi_{\text{own}}^{(k)}}$ converges to a normal distribution for both DESYNC and PCO-based desynchronization, which, as the next two subsections will show, turns out to be the case. We therefore use a stochastic criterion for convergence based on the confidence intervals of the normal distribution [35]. By defining the confidence coefficient

$$c_{\text{conf}} = \Pr \left[\left| \Phi_{\text{own}}^{(k)} - \mu_{\Phi_{\text{own}}^{(k)}} \right| \leq b_{\text{thres}} \right], \quad 0 < c_{\text{conf}} < 1, \quad (10)$$

and replacing in (9), we get

$$\sigma_{\Phi_{\text{own}}^{(k)}} = \frac{b_{\text{thres}}}{\sqrt{2} \times \text{erf}^{-1}(c_{\text{conf}})}, \quad (11)$$

with $\text{erf}^{-1}(u)$ the inverse error function that can be computed by its Maclaurin series

$$\text{erf}^{-1}(u) = \frac{1}{2} \sqrt{\pi} \left(u + \frac{\pi}{12} u^3 + \frac{7\pi^2}{480} u^5 + \dots \right). \quad (12)$$

Thus, (11) becomes the mechanism for defining the phase update iteration leading to SState. Specifically, we determine the firing cycle k_{SState} for which $\sigma_{\Phi_{\text{own}}^{(k_{\text{SState}})}}$ is closest to the right-hand side of (11). That is, we determine the firing cycle leading to convergence to SState with probability that closely matches c_{conf} , which is our (predetermined) confidence.

Definition 1 (Steady State): We define a desynchronization primitive as being in steady state with $c_{\text{conf}} \times 100\%$ confidence, at the k_{SState} th firing cycle, $0 < c_{\text{conf}} < 1$, where

$$k_{\text{SState}} = \arg \min_{\forall k \in \mathbb{N}} \left| \sigma_{\Phi_{\text{own}}^{(k)}} - \frac{b_{\text{thres}}}{\sqrt{2} \times \text{erf}^{-1}(c_{\text{conf}})} \right|, \quad (13)$$

with $\sigma_{\Phi_{\text{own}}^{(k)}}$ the standard deviation of a node's own firing phase PDF at the application of the updates of (1) or (4) during its k th firing cycle.

Since $\sigma_{\Phi_{\text{own}}^{(k)}}$ is affected by measurement noise, in order for the system to remain in the converged state indefinitely, the threshold for the convergence, b_{thres} , must be set according to the (estimated) σ_Δ . Conversely, we can treat the entire desynchronization process as a "black box" system and estimate σ_Δ by measuring the phase deviation from the mean obtained when performing the update of (1) or (4) during SState. This will be demonstrated in the experimental section.

A. Modeling of DESYNC Convergence

Proposition 1. Under Assumptions 1–3, the expected number of firing cycles for the DESYNC phase update of (1) to converge according to Definition 1 is

$$k_{\text{desync}} = \arg \min_{\forall k \in \mathbb{N}} \left| \sigma_{\text{desync}, k} - \frac{b_{\text{thres}}}{\sqrt{2} \times \text{erf}^{-1}(c_{\text{conf}})} \right|, \quad (14)$$

with

$$\sigma_{\text{desync},k} = \sqrt{\|\mathbf{v}_W^{(k)}\|^2 \sigma_{\Phi^{(1)}}^2 + \sum_{j=1}^k \|\mathbf{v}_W^{(j)}\|^2 \sigma_{\Delta}^2}, \quad (15)$$

$$\mathbf{v} = \begin{bmatrix} \frac{\alpha}{2} & 1 - \alpha & \frac{\alpha}{2} \end{bmatrix}, \quad (16)$$

and

$$\mathbf{v}_W^{(j)} = \underbrace{(\mathbf{v} * \dots * \mathbf{v})_W}_{j \text{ times}},$$

being the vector produced by j consecutive circular convolutions of period W .

Proof: Consider any single node in the WSN. We denote the initial phase random variables corresponding to the node under consideration by the $1 \times W$ vector

$$\Phi^{(1)} = [\dots \Phi_{-1}^{(1)} \Phi_{\text{own}}^{(1)} \Phi_{+1}^{(1)} \dots]. \quad (17)$$

The corresponding additive measurement noise sources [independent identically distributed (i.i.d.) random variables from Assumption 3] are denoted by the $1 \times W$ vector

$$\Delta^{(1)} = [\dots \Delta_{-1}^{(1)} \Delta_{\text{own}}^{(1)} \Delta_{+1}^{(1)} \dots]. \quad (18)$$

Evidently, the number of elements before and after $\Phi_{\text{own}}^{(1)}$ and $\Delta_{\text{own}}^{(1)}$ in (17) and (18) depends on how many fire message broadcasts (firings) precede or follow the node's own firing during its initial firing cycle. The first phase update of (1) is expressed stochastically as

$$\begin{aligned} \Phi_{\text{own}}^{(1)} \leftarrow & \left[(1 - \alpha) (\Phi_{\text{own}}^{(1)} + \Delta_{\text{own}}^{(1)}) \right. \\ & \left. + \frac{\alpha}{2} (\Phi_{-1}^{(1)} + \Delta_{-1}^{(1)} + \Phi_{+1}^{(1)} + \Delta_{+1}^{(1)}) \pmod{1} \right]. \end{aligned} \quad (19)$$

Notice that (19) imposes that the statistics of $\Phi_{-1}^{(1)}$ and $\Phi_{+1}^{(1)}$ correspond to the initial firing cycle (Assumption 2). This is because, during each phase update, we do not take into account nodes' phase updates that may have been carried out *during* the first firing cycle. This corresponds to the operational form of the DESYNC algorithm (i.e., ‘‘DESYNC stale’’ of [4], [20]). It is straightforward to derive from (19) that $\mu_{\Phi_{\text{own}}^{(1)}}$ remains unchanged after the first phase update, while the standard deviation is modified to:

$$\sigma_{\text{desync},1} = \|\mathbf{v}\| \sqrt{(\sigma_{\Phi^{(1)}}^2 + \sigma_{\Delta^{(1)}}^2)}. \quad (20)$$

Furthermore, by writing (19) for all the phase elements of $\Phi^{(1)}$ given in (17), we reach

$$\Phi^{(1)} \leftarrow \left[\left[\mathbf{v} * (\Phi^{(1)} + \Delta^{(1)}) \right]_W \pmod{1} \right]. \quad (21)$$

The circular convolution performs periodic extension of the phase and noise vectors of (17) and (18), which corresponds to the circular dependency between consecutive firing cycles⁴.

⁴For the special cases of $W \in \{2, 3, 4\}$ nodes, we set $W = 5$ in the circular convolution of (16) to avoid erroneous overlapping within $\mathbf{v}^{(j)}$ due to the short period of the circular convolution.

Generalizing (21) to the k th firing cycle, we reach

$$\begin{aligned} \Phi^{(k)} &= \left(\mathbf{v}_W^{(k)} * \Phi^{(1)} \right)_W \\ &+ \sum_{j=1}^k \left(\mathbf{v}_W^{(j)} * \Delta^{(j)} \right)_W \pmod{1}, \end{aligned} \quad (22)$$

where $\Delta^{(j)}$ is the i.i.d. measurement noise vector per iteration. Therefore, we obtain: $\mu_{\Phi_{\text{own}}^{(k)}} = \mu_{\Phi_{\text{own}}^{(1)}}$ and $\sigma_{\text{desync},k}$, shown in (15). It can now be shown that $\Phi_{\text{own}}^{(k)}$ becomes normally distributed after a few firing cycles (see Appendix A), i.e.,

$$\Phi_{\text{own}}^{(k)} \sim \mathcal{N} \left(\mu_{\Phi_{\text{own}}^{(1)}}, \sigma_{\text{desync},k} \right) \pmod{1}. \quad (23)$$

Hence, we reach (14) for convergence under Definition 1. ■

Proposition 1 shows how k_{desync} is affected by α as well as by the initial conditions and the noise assumptions expressed by $\sigma_{\Phi^{(1)}}$ and σ_{Δ} in Assumptions 2 and 3, respectively. Interestingly, according to (14), the number of nodes, W , does not appear to influence the convergence to the steady state. This is in contrast to the conjecture of Degesys *et al.* [3], [4] given by (3). Nevertheless, experimental results given in the next section will demonstrate that real-world WSNs, as well as simulation results, are in agreement with Proposition 1.

B. Modeling of PCO-based Convergence

Proposition 2. *Under Assumptions 1–3, the expected number of firing cycles for the PCO-based phase update of (4) to converge according to Definition 1 is*

$$\begin{aligned} k_{\text{PCO}} &= \arg \min_{\forall k \geq 2} \left[\sum_{l=2}^k \left[\text{erf} \left(\frac{\lfloor \frac{W}{2} \rfloor + 1}{W \sigma_{\text{PCO},l} \sqrt{2}} \right) \right. \right. \\ &\quad \left. \left. - \frac{1}{2} \text{erf} \left(\frac{1}{W \sigma_{\text{PCO},l} \sqrt{2}} \right) \right] + 1 - \frac{1}{W} - l_{\text{SSupd}} \right], \end{aligned} \quad (24)$$

with

$$l_{\text{SSupd}} = \arg \min_{\forall l \in \mathbb{N}} \left| \sigma_{\text{PCO},l} - \frac{b_{\text{thres}}}{\sqrt{2} \times \text{erf}^{-1}(c_{\text{conf}})} \right|, \quad (25)$$

and $\forall l \in \mathbb{N}$,

$$\sigma_{\text{PCO},l} = \sqrt{(1 - \alpha)^{2l} \sigma_{\Phi^{(1)}}^2 + \frac{(\alpha - 1)^2}{\alpha(\alpha - 2)} [(1 - \alpha)^{2l} - 1] \sigma_{\Delta}^2}. \quad (26)$$

Proof: We separate the proof into three parts, based on the temporal evolution of the convergence process. We present here the main part of the proof, which comprises the analysis of the first firing cycle and the derivation of the l th phase update of a node during its k th firing cycle in PCO, while the remaining details to complete the proof are given in Appendix B.

First firing cycle: Consider any single node in the WSN. The expected number of phase updates it will perform

within its first firing cycle is equal to the number of fire message broadcasts (firings) expected to be heard by the node within $\varphi_{\text{own}}^{(1)} \in (1 - \frac{1}{W}, 1)$, which is

$$\sum_{j=1}^{W-1} j \binom{W-1}{j} \left(\frac{1}{W}\right)^j \left(1 - \frac{1}{W}\right)^{W-1-j} = 1 - \frac{1}{W}. \quad (27)$$

This stems from the binomial theorem, since Assumption 2 mandates that the initial phase of each node is uniformly distributed within $[0, 1)$. Via (4), a phase update during the first firing cycle of a node can be expressed stochastically as

$$\Phi_{\text{own}}^{(1)} \leftarrow \left[(1 - \alpha) (\Phi_{\text{own}}^{(1)} + \Delta_{\text{own}}^{(1)}) + \alpha \left(1 - \frac{1}{W}\right) \pmod{1} \right], \quad (28)$$

with $\Delta_{\text{own}}^{(1)}$ the random variable modeling the measurement noise (Assumption 3) of the node's own phase. From (8) and (28) we obtain

$$\mu_{\Phi_{\text{own}}^{(1)}} = 1 - \frac{1}{W}, \quad (29)$$

i.e., the mean values of successive fire message updates remain equidistant after the first firing cycle of a node. The standard deviation of $\Phi_{\text{own}}^{(1)}$ after the update of (28) is

$$\sigma_{\text{PCO},1} = (1 - \alpha) \sqrt{\sigma_{\Phi^{(1)}}^2 + \sigma_{\Delta}^2}. \quad (30)$$

Generalizing (28) to the l th phase update of the k th firing cycle of a node, leads to

$$\begin{aligned} \Phi_{\text{own}}^{(k)} &= (1 - \alpha)^l \Phi_{\text{own}}^{(1)} + \sum_{j=1}^l (1 - \alpha)^j \Delta_{\text{own}}^{(l-j+1)} \\ &+ \alpha \left(1 - \frac{1}{W}\right) \sum_{j=0}^{l-1} (1 - \alpha)^j \pmod{1}, \end{aligned} \quad (31)$$

with $\Delta_{\text{own}}^{(l-j+1)}$ i.i.d. random variables, each stemming from Assumption 3. The mean of $\Phi_{\text{own}}^{(k)}$ is given by (29) and its standard deviation is given by (26). Similarly to Proposition 1, it can be shown that $\Phi_{\text{own}}^{(k)}$ becomes a normally-distributed random variable after a few phase updates (see Appendix A). We can thus reach convergence under Definition 1 for l given by (25). However, given that in PCO-based desynchronization the number of phase updates per firing cycle is *not* fixed, i.e., in general, $l \neq k$, in order to derive the *expected number of firing cycles* until a node converges to steady state, we need to derive the expected number of phase updates after each firing cycle. We can then match the number of phase updates expected to take place until convergence to the corresponding number of firing cycles. The details of this process and the proof of (24) are given in Appendix B. ■

Proposition 2 shows that k_{PCO} is affected by α , as well as by the initial conditions and the noise assumptions, expressed by $\sigma_{\Phi^{(1)}}$ and σ_{Δ} in Assumptions 2 and 3. The total number of nodes, W , is also influencing the number of iterations for convergence to the steady state. However, as it will be shown by the next section (experiments), this effect is negligible in practice. This is in contrast to the

lower bound derived by Pagliari *et al.* [3], given by (5). However, the experimental results of the next section will demonstrate that real-world WSNs, as well as simulation results, are in agreement with Proposition 2.

IV. EXPERIMENTAL VALIDATION

For our experiments, we used iMote2 Crossbow sensors with TinyOS1.x. All nodes use the IEEE 802.15.4 standard with the default 2.4GHz Chipcon CC2420 wireless transceiver. We followed the TinyOS standard message format but reduced it to two data bytes when sending fire messages. Similar to prior work [4], we set the backoff time to 1.2ms.

A. Standard Deviation of the Phase Measurement Noise

The test environment was a standard university laboratory room. Our approach for measuring σ_{Δ} was carried out as follows: (i) we implemented the DESYNC and PCO-based desynchronization in TinyOS nesC code as described in Section II; (ii) we set $\alpha = 0.95$ to ensure maximum coupling strength and $T = 1\text{s}$ (this period value was used in all our experiments) and (iii) we measured the oscillatory behavior of each node's phase after the WSN was left operating for a prolonged interval of time (during which nodes were occasionally moved within the test area) to ensure convergence to SState. The statistics of the oscillatory phase behavior, observed via this experiment, express the cumulative effects of interference, mobility and clock drift amongst nodes. This approach is easy to replicate under any real-world WSN setup involving varying levels of interference or node mobility [7], [12], [26].

For both algorithms, we found the standard deviation of the oscillating phase amplitude around the SState value of each node's phase to be $\sigma_{\Delta} = 0.34\text{ms}$ and the accumulated phase statistics over all nodes were confirmed as marginally white. This validates our noise assumption stated in Assumption 3. The derived value for σ_{Δ} was used during the experimental validation of Proposition 1 and Proposition 2. No other parameter tuning is needed for the proposed model.

B. Measurement and Simulation Setup

In both DESYNC and PCO-based desynchronization, once all nodes were activated to transmit and receive on a single channel, a special "mix message" was broadcast by one of the nodes (chosen randomly) in order to trigger all nodes to set their initial fire message phase to a random interval within $T = 1\text{s}$ from its reception. This creates the initial conditions of Assumption 2. The nodes will then desynchronize their transmission of fire messages and converge to fair TDMA scheduling. We present results under two convergence thresholds, i.e., $b_{\text{thres}} = 0.001$ and $b_{\text{thres}} = 0.020$, with coupling constants $\alpha \in \{0.05, \dots, 0.95\}$ and number of nodes $W \in \{4, 8, 16\}$. We use $c_{\text{conf}} = 0.9999$ to detect convergence under Definition 1 with near certainty. Finally, we have experimented with various settings

for the firing period and all such experiments led to very similar results for the convergence iterations to the SState. Thus, all reported experiments use firing period of $T = 1s$, which complies with previous work [3], [4], [10], [12], [20].

Under this setup, each node detects convergence to SState by checking if (2) is valid for its last ten firing cycles. After achieving SState and remaining in this state for 50 firing cycles, a node broadcasts another mix message, in order to repeat the process. Each node reported the number of its firing cycles until convergence was detected (minus nine cycles) via a special “report” message to a base station listening passively to all messages for monitoring purposes. This facilitated the automated collection of 50 such results per number of nodes, threshold and coupling constant.

In order to cross-validate our theoretical and experimental results with simulations, we used the Matlab code of Degesys *et al.* [4] for DESYNC and added to it Matlab code for PCO with inhibitory coupling. In order to simulate the noise conditions observed in our experimental setup, we deliberately apply zero-mean additive noise in each phase update with $\sigma_{\Delta} = 0.34ms$ and set each node to misfire with probability 0.4%. Despite the fact that the simulation cannot capture the complex behavior of the real system in full detail, it allows for numerous desynchronization processes to be simulated (300 Matlab runs per triplet $\{W, \alpha, b_{thres}\}$ for each algorithm).

C. DESYNC Results

The results for this desynchronization mechanism are reported in Fig. 2. All measurements around a value of α correspond to results with that value of α ; they have been plotted slightly separately solely for ease of illustration. For comparison purposes, we have also included the order-of-convergence conjecture of (3) [3] [4] in our results, by scaling the order estimate to fit within the range of the obtained experiments and simulations.

The results of Fig. 2 show that the WSN tends to converge to steady state faster when α decreases (until $\alpha = 0.25$), since the presence of measurement noise causes higher-amplitude perturbations for strong coupling, i.e., for high values of α . However, for very small values of α , the convergence iterations increase dramatically due to weakened coupling between phase-neighboring nodes. Moreover, by comparing the convergence results for low and high convergence threshold b_{thres} , one can observe that the use of small convergence threshold increases the required convergence iterations to SState.

The proposed model predicts these trends in the convergence iterations accurately. Specifically, the estimate of Proposition 1 is within one standard deviation of the experimental and simulation results for the vast majority of cases. The Pearson correlation coefficients for the proposed model curves against the mean experimental values (averaged over $W \in \{4, 8, 16\}$) were found to be 0.9893 and 0.9931 for $b_{thres} = 0.001$ and $b_{thres} = 0.020$, respectively. The corresponding Pearson correlation coefficients for the conjecture of (3) from [3], [4] were 0.8639 and 0.9464.

These results underline the superior estimation accuracy of our approach. Finally, the experimental results of Fig. 2 show no statistical dependence on W , which agrees with Proposition 1.

D. Results with PCO-based Desynchronization

The results are reported in Fig. 3 for both small and large convergence thresholds. Since (5) provided negative estimates for most values of α , we added an offset to the results of the bound to bring as many as possible to the non-negative region and present only the non-negative ones. Evidently, the bound of (5) does not match the observed behavior. We remark however that this is to be expected, as the bound of (5) is derived under the assumption that each firing is influenced only by the firing of one phase-neighboring node [3].

Fig. 3 shows that, in PCO, the convergence iterations decrease monotonically with α . The proposed model predicts this trend correctly and, remains within one standard deviation from the majority of the experimental and simulation results. The model results do not change⁵ for different values of W , which agrees with the overall experimentally-observed behavior of the system. Finally, by comparing the convergence results for low and high convergence threshold, we note that the use of small convergence threshold increases the required convergence iterations to SState. The proposed model predicts this behavior correctly and agrees with the experimental trends reported. The Pearson correlation coefficients for the model curves against the mean experimental values were 0.9739 for $b_{thres} = 0.001$ and 0.9989 for $b_{thres} = 0.020$.

E. Discussion

By cross referencing between Fig. 2 and Fig. 3 we can compare the convergence iterations of both algorithms under different settings. Under appropriate choice for the coupling coefficient, α , the required firing cycles for convergence with PCO-based desynchronization is comparable to those of DESYNC. In addition, we can make the following observations.

- *Accuracy of previous analytic work on estimation of convergence iterations:* Previous estimates or bounds must be scaled to fit the range of the experimental values, as they are either order-of-convergence estimates [i.e., (3)], or overly optimistic bounds [i.e., (5)]. Moreover, as illustrated in Fig. 2 and Fig. 3, when varying the coupling parameter α , previous estimates do not accurately match the behavior observed in the experimentally-obtained convergence iterations.
- *Impact of measurement noise:* Contrary to the proposed model, these previous estimates do not take into account the measurement noise conditions. Independently from this work, and following different analysis and modeling approaches, recent work in synchronization [29], [30] and desynchronization [10] has shown

⁵i.e. Proposition 2 leads to the same results for the convergence iterations under different values of W

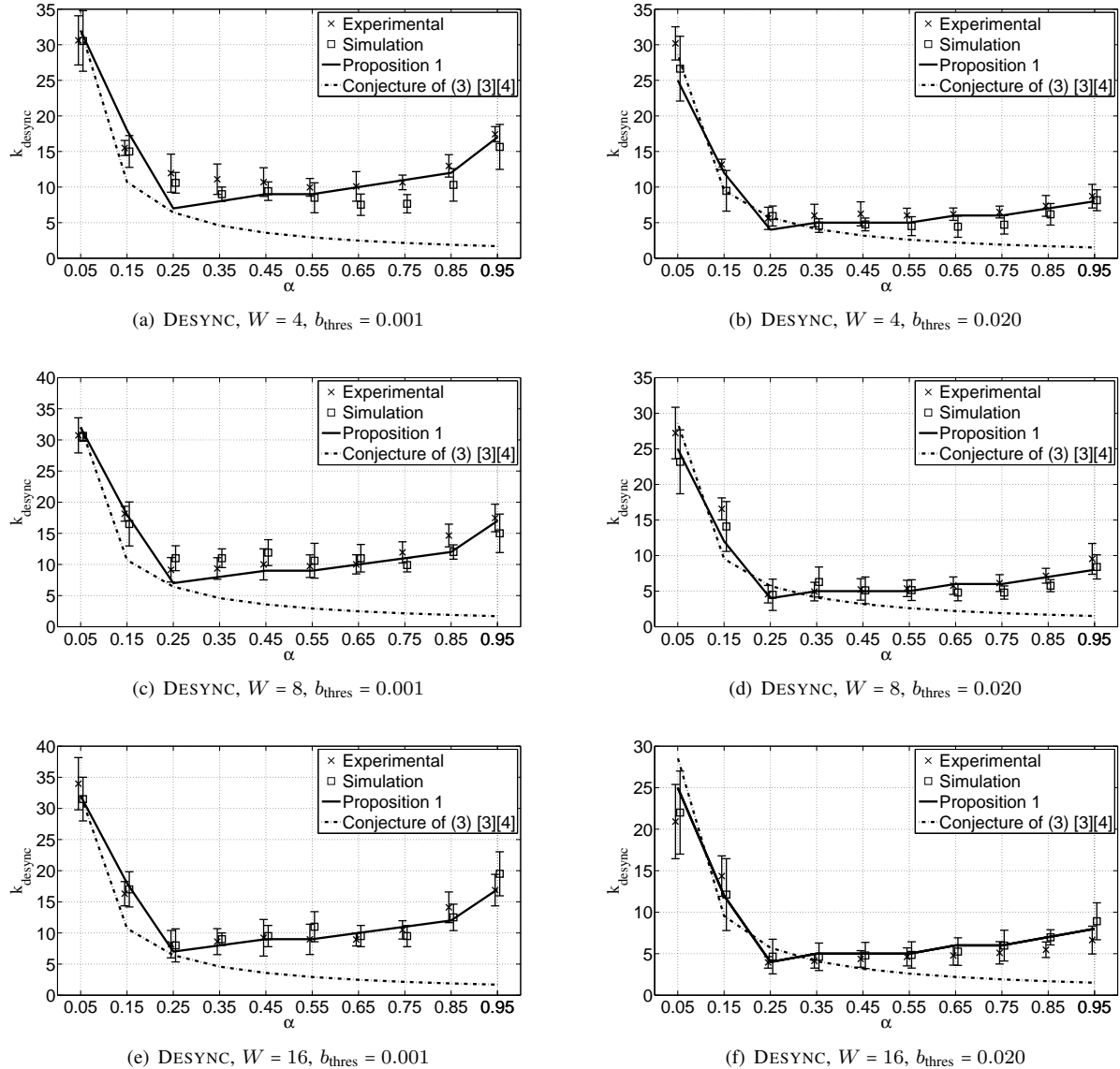


Figure 2. Required firing cycles for convergence to SState for the DESYNC algorithm for various values of α . The vertical error bars correspond to one standard deviation from the experimental (or simulation) mean values, which are indicated by marks.

that noise in the desynchronization phase update (e.g., by quantization) and drops (or collisions) of beacon messages can affect the convergence iterations and can lead to convergence iterations that deviate from the estimates obtained based on the ideal (noise-free) model assumed by earlier work.

- *Generalization to non-uniformly distributed initial firings:* In Propositions 1 and 2, every node's initial phase random variable was assumed to be i.i.d. uniform via Assumption 2. However, Propositions 1 and 2 hold for any i.i.d. random variable $\Phi^{(1)}$ that satisfies the three conditions for the generalized form of the central limit theorem to be applicable [35, pp. 219-220] (see Appendix A).
- *Utilized desynchronization algorithm and the network topology:* While we focused on the initial desynchronization proposals with limited listening, recent

work has extended these initial paradigms towards other frameworks, where nodes adjust their own firing phase based on larger listening intervals and/or via the usage of anchored (i.e., non-adjusting) nodes [10]–[12], [21], [23], [25], [26]. While we do not investigate the applicability of our modeling approach to all such frameworks, our analysis includes the coupling parameter α and adjusts according to the node firings received within the predetermined listening interval. It is also worth noting that Propositions 1 and 2 cover the important scenario of a fully-meshed (all-to-all) WSN. Desynchronization extensions to multihop scenarios have been investigated by several works [20]–[22]. Given that there is a large variation in the topologies to consider and that it has been shown that the steady-state difference between the phases of consecutive firings, as well as the average number of convergence

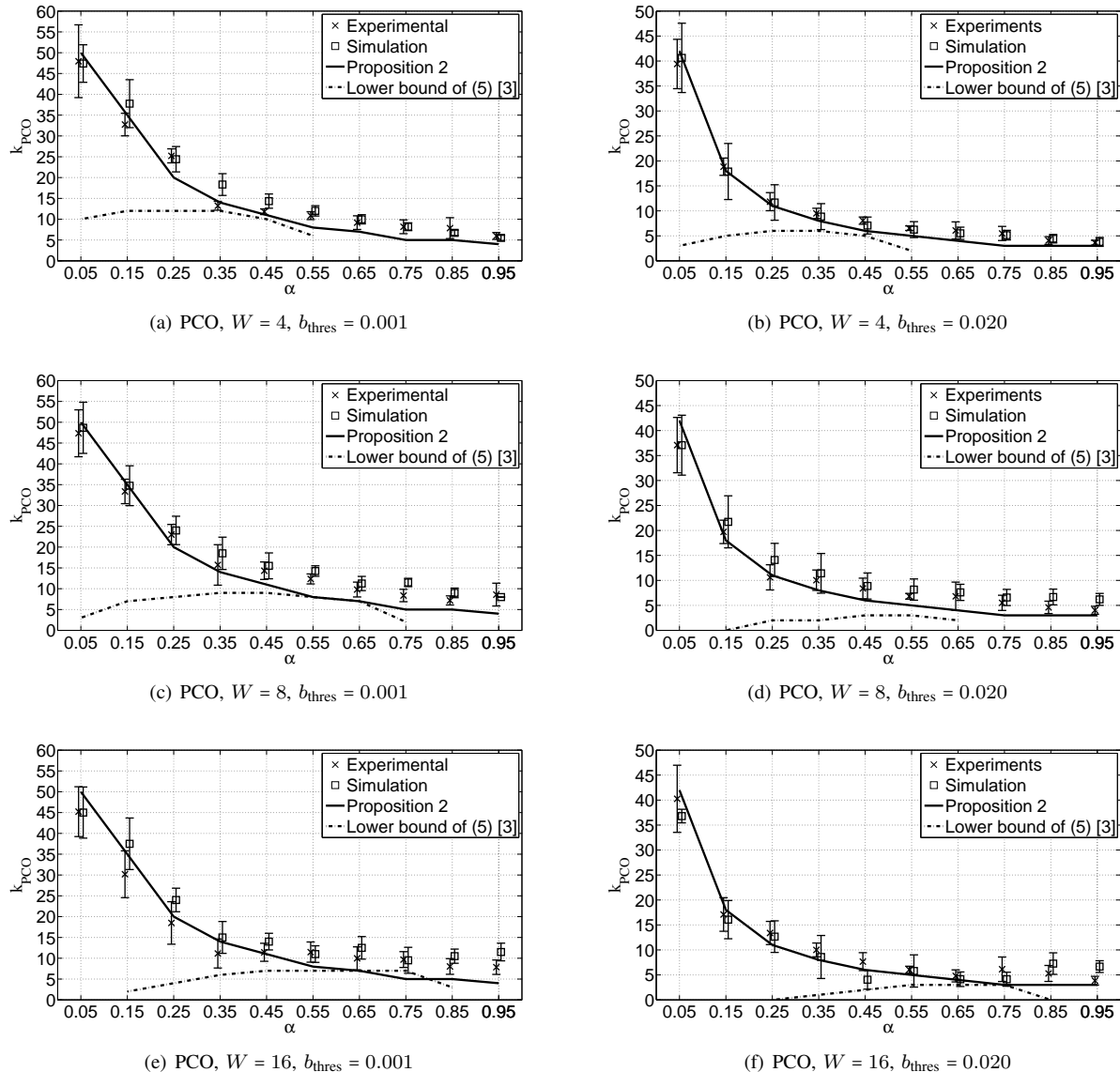


Figure 3. Required firing cycles for convergence to SState for the PCO-based algorithm for various values of α . The vertical error bars correspond to one standard deviation from the experimental (or simulation) mean values, which are indicated by marks.

iterations, varies depending on the network topology under consideration [21], [22], [25], [30] (due to the “selective” listening of beacons stemming solely from non-hidden nodes), we do not investigate the validity of our analysis to such cases. However, we remark that this could be attempted following the same approach as for Propositions 1 and 2, if the topology specification is known *a-priori*.

- *Consideration of phase updates performed during each node’s firing cycle:* As mentioned in the proof of Proposition 1, to calculate $\Phi_{\text{own}}^{(k)}$ and its moments for DESYNC and PCO-based convergence, our analysis does not take into account the variability in the neighboring nodes’ phase updates⁶. This is in agreement

⁶e.g., updates that may have been carried out during the k th firing cycle of the node under consideration, or variability due to misfiring or other non-idealities not captured by our noise assumption (Assumption 2)

with the way DESYNC is applied in practice and constitutes a simplification for PCO-based desynchronization [3], [4]. Given that our stochastic modeling framework is in good agreement with the average experimental and simulation results without requiring experimental tuning (besides knowledge of the standard deviation of the phase measurement noise), our approach forms an important step towards considering stochastic models for the convergence of desynchronization systems.

V. APPLICATION EXAMPLES

The proposed stochastic estimation framework can benefit desynchronization-based TDMA protocols in WSNs by analytically estimating the impact of the phase-coupling constant α , convergence threshold b_{thres} and firing cycle

period T on such deployments [3], [4], [32]. We present two such examples.

A. Maximizing the Bandwidth Per Node

In the first case, we consider a WSN that initially comprises W nodes, where nodes are expected to join or leave the network every T_{swap} seconds. This scenario usually occurs when mobile nodes periodically enter and exit the coverage area of the network, or some nodes switch on and off periodically to conserve energy. In fair TDMA scheduling, the bandwidth per node is $\frac{B_{\text{WSN}}}{W}$ bps, where B_{WSN} is the maximum application-layer bandwidth⁷ in IEEE 802.15.4. In practice, the fluctuating number of nodes in the WSN will result in bandwidth loss as, each time nodes join or leave, the system needs to converge to SState before transmission resumes with equal slot size [4], [32], [33]. Using the proposed framework, we can derive an estimate of the expected bandwidth per node under such conditions. Specifically, if the PCO or DESYNC firing-cycle period is T seconds and the node joining or exiting occurs (on average) every T_{swap} seconds, the expected bandwidth per node can be estimated as

$$B_{\text{swap}} = \left(1 - \frac{k_{\text{method}}T}{T_{\text{swap}}}\right) \frac{B_{\text{WSN}}}{W}, \text{ method} \in \{\text{desync, PCO}\}, \quad (32)$$

with k_{method} the expected firing cycles until convergence to SState, given by Proposition 1 or Proposition 2. The factor $\frac{k_{\text{method}}T}{T_{\text{swap}}}$ in (32) expresses the normalized loss of bandwidth per node due to convergence to TDMA under DESYNC and PCO every time nodes join or leave the network.

Using the experimental setup of Section IV, we present an example of the calculation of (32) for a WSN comprising $W = 10$ nodes with $T = 1$ s and $1 \sim 3$ nodes entering or exiting the network every T_{swap} seconds, with $T_{\text{swap}} \in [70, 130]$ to incorporate up to 30% variability around the mean value of 100s. The maximum application-layer bandwidth was measured to be $B_{\text{WSN}} = 86$ kbps. Table II and Table III present the results when using the value of α that was estimated to provide the minimum k_{method} under Propositions 1 and 2 against the result when using values for α suggested from previous work. It is evident that, for all cases, the proposed model provides the setting for α that minimizes the convergence iterations and leads to the maximum achievable bandwidth per node. This is despite the fact that the model assumes uniformly-distributed initial fire times, while in the application results the values of most nodes may be (approximately) equidistant at the moment $1 \sim 3$ nodes enter or exit the network. The impact on the achieved bandwidth per node is much more pronounced in the case of DESYNC, where previous work used $\alpha = 0.95$ instead of the best option, which is $\alpha = 0.25$. The results in Table II and Table III, show that selecting α through the proposed model brings gains of up to 13% in the

⁷Following the approach of Degeys [4], we estimate B_{WSN} by using a single transmitter and receiver to measure the achieved delivery rate at the application layer.

bandwidth per node compared with the standard settings used in existing works [3], [4].

B. Estimating the Required Firing Cycle Period

In the second application example, we focus on the case of a WSN using DESYNC or PCO and requiring convergence to steady state to be achieved within T_{SState} seconds on average. The desired value of T_{SState} depends on the application context, e.g., a predefined value T_{SState} in order to limit the delay and buffering requirements between the data acquisition and transmission when the WSN is activated [13], [32], [33].

Based on Proposition 1 and Proposition 2, and under given settings for b_{thres} and α , we can match the period of firing cycles (T) to the desired convergence time by: $k_{\text{method}}T = T_{\text{SState}}$. By solving the last equation for T we derive the firing cycle period that meets the convergence time expectation under k_{method} given by Proposition 1 and Proposition 2 for each setting of b_{thres} and α .

We have experimented with various parameters of DESYNC or PCO and the obtained theoretical and experimental results are reported in Table IV using the average obtained from multiple runs with $W \in \{4, 8, 16\}$. With the exception of two cases (DESYNC at: $\alpha = 0.25$, $b_{\text{thres}} = 0.001$ and PCO at: $\alpha = 0.95$, $b_{\text{thres}} = 0.001$), the theoretical prediction is within a 25% margin of the experimentally observed convergence time.

VI. CONCLUSIONS

A novel stochastic estimation framework for the convergence iterations to fair TDMA scheduling is proposed for the two desynchronization primitives with limited listening, namely, DESYNC and pulse-coupled oscillators with inhibitory coupling. Our stochastic estimates establish the expected firing cycles until each node's firing converges to the steady state with very high confidence. For both algorithms, our analytic expressions are validated based on simulations and experiments with a fully-meshed network of wireless sensors. The results show that our estimates are more accurate than previous order-of-convergence estimates and lower bounds. Our model incorporates the influence of system parameters (i.e., total number of nodes, coupling coefficient, convergence threshold and phase measurement noise) on the expected convergence iterations. Therefore, it can be used to estimate the best operational parameters (and the associated delay) to establish fair TDMA scheduling under several desynchronization-based WSN protocols. More broadly, the analysis of this paper contributes towards the analytic understanding of how a desynchronization system is expected to evolve from random initial conditions to the desynchronized steady state.

APPENDIX A

We show that $\Phi_{\text{own}}^{(k)}$ of (22) and (31), i.e., the RV modeling each node's own phase during its k th firing cycle under DESYNC and PCO (respectively), is normally distributed after a few phase updates.

Table II

DESYNC: AVERAGE BANDWIDTH PER NODE, B_{SWAP} (IN KBPS), UNDER DIFFERENT CONVERGENCE THRESHOLDS AND DIFFERENT VALUES FOR THE PHASE-COUPLING CONSTANT α .

DESYNC	$b_{\text{thres}} = 0.001$ (1ms)		$b_{\text{thres}} = 0.020$ (20ms)	
	Measurement	Theoretical via (32) and Prop. 1	Measurement	Theoretical via (32) and Prop. 1
$\alpha = 0.95$ [4]	6.55	7.14	7.72	7.91
$\alpha = 0.25$ (from Fig. 2)	7.41	8.00	8.17	8.26
Average bandwidth gain per node (%)	13.13	12.04	5.83	4.42

Table III

PCO: AVERAGE BANDWIDTH PER NODE, B_{SWAP} (IN KBPS), UNDER DIFFERENT CONVERGENCE THRESHOLDS AND DIFFERENT VALUES FOR THE PHASE-COUPLING CONSTANT α .

PCO	$b_{\text{thres}} = 0.001$ (1ms)		$b_{\text{thres}} = 0.020$ (20ms)	
	Measurement	Theoretical via (32) and Prop. 2	Measurement	Theoretical via (32) and Prop. 2
$\alpha = 0.75$ [3]	6.90	8.17	7.65	8.34
$\alpha = 0.95$ (from Fig. 3)	7.54	8.26	8.13	8.34
Average bandwidth gain per node (%)	9.28	1.10	6.27	0.00

Table IV

DESYNC AND PCO FOR DESIRED CONVERGENCE TIME OF 10S. THE CORRESPONDING FIRING CYCLE PERIOD, T , IS GIVEN IN PARENTHESES (IN SECONDS) IN THE "THEORETICAL" COLUMNS.

Method	DESYNC		PCO	
	Experimental T_{SState}	Theoretical T_{SState}	Experimental T_{SState}	Theoretical T_{SState}
$\alpha = 0.25, b_{\text{thres}} = 0.001$	14.01	10.00 (1.43)	7.80	10.00 (0.50)
$\alpha = 0.95, b_{\text{thres}} = 0.001$	12.15	10.00 (0.59)	18.75	10.00 (2.50)
$\alpha = 0.25, b_{\text{thres}} = 0.020$	11.75	10.00 (2.50)	10.83	10.00 (0.91)
$\alpha = 0.95, b_{\text{thres}} = 0.020$	10.38	10.00 (1.25)	12.65	10.00 (3.33)

Once (22) and (31) are reached, we can make the following observations:

- $\Phi_{\text{own}}^{(k)}$ is a linear mixture of independent random variables, i.e.: (i) i.i.d. noise and phase vectors Δ and $\Phi^{(1)}$ in DESYNC; (ii) $\Phi_{\text{own}}^{(1)}$ and $\forall l, j: \Delta_{\text{own}}^{(l-j+1)}$ in PCO;
- for DESYNC: $\forall k \in \mathbb{N}^+$, we can pick $\varepsilon = (1 - \alpha)^k \sigma_{\Phi^{(1)}}$ and, from (15), $\sigma_{\text{desync}}^{(k)} > \varepsilon$;
- for PCO: $\forall l \in \mathbb{N}^+ : \sigma_{\text{PCO},l} > (1 - \alpha)^l \sigma_{\Phi^{(1)}}$;
- all initial PDFs have finite support (they are all variants of the uniform distribution); hence, densities $P_{\Phi_{\text{own}}^{(k)}}$ will have finite support since they are linear mixtures of PDFs with finite support.

These observations satisfy the three conditions for the generalized form of the central limit theorem to be applicable [35, pp. 219-220], and thus $\Phi_{\text{own}}^{(k)}$ becomes a normally-distributed random variable after a few phase updates. Papoulis [35, pp. 219-220] suggests that the normal PDF is accurately approximated with just 5 linear combinations of PDFs satisfying the above criteria, meaning that five phase updates will suffice for the normal distribution to be an accurate approximation of the firing phase of each node.

APPENDIX B

We present the analysis that matches the expected number of phase updates until convergence to the expected number of firing cycles and concludes the proof of (24).

Subsequent firing cycles, effect of phase-neighboring

firings: A pictorial illustration of the PDF of the l th phase update of a node (performed during its k th firing cycle, with $k, l \geq 2$) is given in Fig. 4 in conjunction with its listening interval and the PDFs of the two previous firings and the next firing. Since all phase RVs are normally distributed after a few phase updates, it is straightforward to infer from Fig. 4 that the probability for the previous firing (represented by RV $\Phi_{-1}^{(k)}$) to occur within the listening interval is

$$\frac{1}{2} \operatorname{erf} \left(\frac{1}{W \sigma_{\text{PCO},l} \sqrt{2}} \right).$$

Moreover, the probability that $\Phi_{-2}^{(k)}$ will occur within the node's listening interval is

$$\frac{1}{2} \left[\operatorname{erf} \left(\frac{2}{W \sigma_{\text{PCO},l} \sqrt{2}} \right) - \operatorname{erf} \left(\frac{1}{W \sigma_{\text{PCO},l} \sqrt{2}} \right) \right].$$

This is also the probability that $\Phi_{+1}^{(k)}$ will occur within the node's listening interval.

Subsequent firing cycles and the effect of all firings within a window of W firing events:

We can now generalize the previous calculation to the probability of occurrence of the $\lfloor \frac{W}{2} \rfloor$ previous and next firings within the node's listening interval. Beyond the previous firing, for the j th firing after the node's own firing or the $(j+1)$ th firing before the node's own firing ($1 \leq j \leq \lfloor \frac{W}{2} \rfloor$), this probability is

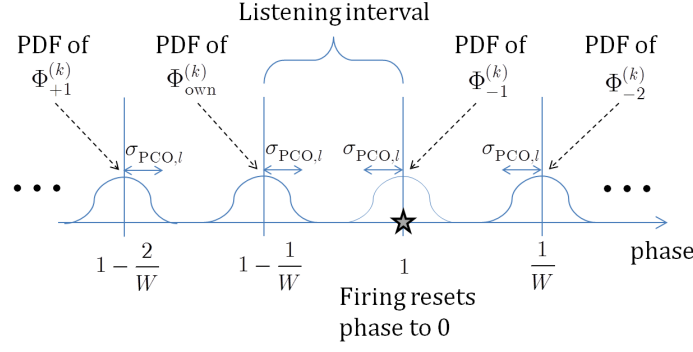


Figure 4. A pictorial illustration of the PDFs of the phase RVs $\{\dots, \Phi_{-2}^{(k)}, \Phi_{-1}^{(k)}, \Phi_{\text{own}}^{(k)}, \Phi_{+1}^{(k)}, \dots\}$ of the k th firing cycle of “own” node, during “own” node’s l th phase update, performed via (4).

$$\frac{1}{2} \left[\operatorname{erf} \left(\frac{j+1}{W\sigma_{\text{PCO},l}\sqrt{2}} \right) - \operatorname{erf} \left(\frac{j}{W\sigma_{\text{PCO},l}\sqrt{2}} \right) \right].$$

Hence, summing up the probabilities of firings occurring within the node’s listening interval for all $\lfloor \frac{W}{2} \rfloor$ firings before and after the current one, the expected number of phase updates is given by the expression in the summation term of (24), where we used the identity ($\forall b \in \mathbb{N}^*, \forall c \neq 0$):

$$\begin{aligned} \sum_{j=1}^b \left[\operatorname{erf} \left(\frac{j+1}{c} \right) - \operatorname{erf} \left(\frac{j}{c} \right) \right] + \frac{1}{2} \operatorname{erf} \left(\frac{1}{c} \right) \\ = \operatorname{erf} \left(\frac{b+1}{c} \right) - \frac{1}{2} \operatorname{erf} \left(\frac{1}{c} \right). \end{aligned} \quad (33)$$

The expected number of phase updates within k firing cycles of a node is

$$\sum_{l=2}^k \left[\operatorname{erf} \left(\frac{\lfloor \frac{W}{2} \rfloor + 1}{W\sigma_{\text{PCO},l}\sqrt{2}} \right) - \frac{1}{2} \operatorname{erf} \left(\frac{1}{W\sigma_{\text{PCO},l}\sqrt{2}} \right) \right] + 1 - \frac{1}{W}.$$

As a result, for l_{SSupd} phase updates leading to convergence under Definition 1 [shown by (25)], the corresponding number of firing cycles is given by (24).

REFERENCES

- [1] O. Simeone, U. Spagnolini, Y. Bar-Ness, and S. Strogatz, “Distributed synchronization in wireless networks,” *IEEE Signal Process. Mag.*, vol. 25, no. 5, pp. 81–97, Sep. 2008.
- [2] R. Leidenfrost and W. Elmenreich, “Firefly clock synchronization in an 802.15.4 wireless network,” *Eurasip J. Embed. Syst.*, vol. 2009, Article id. 18585, 17 pages, 2009.
- [3] R. Pagliari, Y.-W. Hong, and A. Scaglione, “Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling,” *IEEE J. on Select. Areas in Commun.*, vol. 28, no. 4, pp. 564–575, May 2010.
- [4] J. Degeysys, I. Rose, A. Patel, and R. Nagpal, “Desync: Self-organizing desynchronization and tdma on wireless sensor networks,” *Proc. 6th Int. Symp. on IPSN*, pp. 11–20, Apr. 2007.
- [5] A. Mutazono, M. Sugano, and M. Murata, “Energy efficient self-organizing control for wireless sensor networks inspired by calling behavior of frogs,” *Computer Communications*, vol. 35, no. 6, pp. 661–669, 2012.
- [6] H. Yamamoto, N. Wakamiya, and M. Murata, “An inter-networking mechanism using stepwise synchronization for wireless sensor networks,” in *Bio-Inspired Models of Network, Information, and Computing Systems*. Springer, 2012, pp. 276–287.
- [7] T. Nakano, “Biologically inspired network systems: a review and future prospects,” *IEEE Trans. on Syst., Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, no. 5, pp. 630–643, Sep. 2011.
- [8] J. Klinglmayr and C. Bettstetter, “Self-organizing synchronization with inhibitory-couples oscillators: convergence and robustness,” *ACM Transactions on Autonomous and Adaptive Systems*, vol. 7, no. 3, Sep. 2012.
- [9] A. Patel, J. Degeysys, and R. Nagpal, “Desynchronization: The theory of self-organizing algorithms for round-robin scheduling,” *Proc. IEEE Internat. Conf. on Self-Adaptive and Self-Organizing Syst. (SASO’07)*, pp. 87–96, Jul. 2007.
- [10] S. Ashkiani and A. Scaglione, “Discrete dithered desynchronization,” *arXiv preprint arXiv:1210.2122*, 2012.
- [11] S. Choochaisri, K. Apicharttrisor, K. Korprasertthaworn, P. Taechalertpaisarn, and C. Intanagonwiwat, “Desynchronization with an artificial force field for wireless networks,” *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 2, pp. 7–15, 2012.
- [12] C. Lien, S. Chang, C. Chang, and D. Lee, “Anchored desynchronization,” in *Proc. IEEE INFOCOM’12*, 2012, pp. 2966–2970.
- [13] I. Bojic, V. Podobnik, I. Ljubi, G. Jezic, and M. Kusek, “A self-optimizing mobile network: Auto-tuning the network with firefly-synchronized agents,” *Information Sciences, Elsevier*, vol. 182, no. 1, pp. 77–92, 2012.
- [14] R. Pagliari and A. Scaglione, “Scalable network synchronization with pulse-coupled oscillators,” *IEEE Trans. on Mobile Computing*, vol. 10, no. 3, pp. 392–405, Mar. 2011.
- [15] R. E. Mirollo and S. H. Strogatz, “Synchronization of pulse-coupled biological oscillators,” *SIAM Journal on Applied Mathematics*, vol. 50, no. 6, pp. 1645–1662, 1990.
- [16] P. C. Bressloff and S. Coombes, “Desynchronization, mode locking, and bursting in strongly coupled integrate-and-fire oscillators,” *Physical Review Letters*, vol. 81, no. 10, p. 2168, 1998.
- [17] S. R. Campbell, D. L. Wang, and C. Jayaprakash, “Synchrony and desynchrony in integrate-and-fire oscillators,” *Neural computation*, vol. 11, no. 7, pp. 1595–1619, 1999.
- [18] Y. Kuramoto, *Chemical oscillations, waves, and turbulence*. Courier Dover Publications, 2003.
- [19] A. Arenas, A. Diaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou, “Synchronization in complex networks,” *Physics Reports*, vol. 469, no. 3, pp. 93–153, 2008.
- [20] J. Degeysys and R. Nagpal, “Towards desynchronization of multi-hop topologies,” in *Proc. IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO’08)*, 2008, pp. 129–138.
- [21] A. Motkin, T. Roughgarden, P. Skraba, and L. Guibas, “Lightweight coloring and desynchronization for networks,” in *Proc. IEEE INFOCOM’09*, 2009, pp. 2383–2391.
- [22] H. Kang and J. L. Wong, “A localized multi-hop desynchronization algorithm for wireless sensor networks,” in *Proc. IEEE INFOCOM 2009*. IEEE, 2009, pp. 2906–2910.
- [23] D. De Guglielmo, G. Anastasi, and M. Conti, “A localized desynchronization algorithm for periodic data reporting in ieee 802.15.4 wsns,” in *IEEE Symposium on Computers and Communications (ISCC’12)*, 2012, pp. 605–610.
- [24] Y. Wang, F. Nunez, and F. Doyle, “Statistical analysis of the pulse-coupled synchronization strategy for wireless sensor networks,”

- IEEE Trans. on Signal Processing*, vol. 61, no. 21, pp. 5193–5204, 2013.
- [25] C. Muhlberger, “Analyzing a self-organizing multi-hop protocol: Ease of simulations and need for real-world tests,” in *Proc. 9th IEEE Internat. Conf. Wireless Comm. and Mob. Comp., IWCMC, 2013*. IEEE, 2013, pp. 1029–1034.
- [26] T. Settawatcharawanit, S. Choochaisri, C. Intanagonwivat, and K. Rojviboonchai, “V-desync: Desynchronization for beacon broadcasting on vehicular networks,” in *IEEE Vehicular Technology Conference (VTC Spring)*, 2012, pp. 1–5.
- [27] A. Cornejo and F. Kuhn, “Deploying wireless networks with beeps,” *Distributed Computing*, vol. 182, no. 1, pp. 148–162, 2010.
- [28] Y. Wang, F. Nunez, and F. Doyle, “Energy-efficient pulse-coupled synchronization strategy design for wireless sensor networks through reduced idle listening,” *IEEE Trans. on Signal Processing*, vol. 60, no. 10, pp. 5293–5306, 2012.
- [29] J. Klinglmayr, C. Kirst, C. Bettstetter, and M. Timme, “Guaranteeing global synchronization in networks with stochastic interactions,” *New Journal of Physics*, vol. 14, no. 7, p. 073031, 2012.
- [30] J. Nishimura and E. J. Friedman, “Probabilistic convergence guarantees for type-ii pulse-coupled oscillators,” *Physical Review E*, vol. 86, no. 2, p. 025201, 2012.
- [31] Y.-W. Hong, A. Scaglione, and R. Pagliari, “Pulse coupled oscillators’ primitive for low complexity scheduling,” in *IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP’09)*, 2009, pp. 2753–2756.
- [32] D. Buranapanichkit and Y. Andreopoulos, “Distributed time-frequency division multiple access protocol for wireless sensor networks,” *IEEE Wireless Comm. Letters*, vol. 1, no. 5, pp. 440–443, Oct. 2012.
- [33] H. Besbes, G. Smart, D. Buranapanichkit, C. Kloukinas, and Y. Andreopoulos, “Analytic conditions for energy neutrality in uniformly-formed wireless sensor networks,” *IEEE Trans. on Wireless Communications*, pp. 1–16, “to appear”.
- [34] G. Strang and T. Nguyen, *Wavelets and filter banks*. Cambridge University Press, 1996.
- [35] A. Papoulis, *Probability and Statistics*. Prentice Hall, 1989.