Fast Desynchronization Algorithms for Decentralized Medium Access Control based on Iterative Linear Equation Solvers

Daniel Silvestre, João Hespanha and Carlos Silvestre

***Abstract*—We tackle the problem of having multiple transmitters cooperating to be desynchronized using a distributed algorithm. Although this problem can also be found in surveillance, it has the most impact in achieving a fair access to a wireless shared communication medium at the Medium Access Control (MAC) layer in the context of Wireless Sensor Networks (WSNs). In this paper, we first theoretically investigate the convergence rate of various optimization algorithms, giving closed-form expressions for the parameters achieving the best worst-case convergence rate. We then show that a recently proposed time-varying parameters Nesterov algorithm applied to this problem has worse performance assuming one can determine the number of sensors in the network. In order to remove such an assumption, the problem is seen as the solution of a linear equation corresponding to the first optimality condition. Both theoretically and in simulation, we show that using the Gauss-Seidel method improves the speed of convergence, although its performance degrades for large network sizes. In simulations, it is shown the behavior for various number of wireless devices, emphasizing how the algorithms actually perform in comparison with their worst-case theoretical rates for different network sizes.**

***Index Terms*—Distributed control; Communication networks; Optimization algorithms.**

I. Introduction

Desynchronization among different agents in a network plays a role in various tasks including data aggregation, duty cycling and cooperative communications. In the context of Wireless Sensor Networks (WSNs), a key aspect to achieve a fair Time Division Multiple Access (TDMA) scheduling is the definition of distributed algorithms that perform desynchronization at the Medium Access Control (MAC) layer. The problem lies in how to devise a distributed algorithm that can evenly spread the communicating time slots among the nodes [1], [2], [3], [4], [5], [6]. This can be seen as the dual of the consensus problem (see for example [7], [8]) and used in formation algorithms such as [9], [10].

Although there are centralized solutions to the desynchronization problem that rely on a coordination channel, a central node or a global clock (for example making use of GPS) [2], in this paper attention is focused on decentralized solutions. In the literature, it is common to accept algorithms where nodes hop between channels of the physical layer so as to avoid channels with excessive interference. The approach implemented in the Time-Synchronized Channel Hoping (TSCH) [2] protocol has been established as state-of-the-art in the IEEE 802.15.4e-2012 standard [11].

In the literature, various authors [1], [2], [5], [12], [13], [14], [15], [16], [17], [18], [19], [20] have proposed distributed desynchronization algorithms for WSN MAC-layer coordination. The main idea connecting these proposals is in modeling as Pulse-Coupled Oscillators (PCOs) the actions of biological agents such as fireflies, where a timing mechanism with a periodic pulsing is adjusted based on the timings of pulses sensed from a subset of the remaining nodes.

The seminal work by Mirollo and Strogatz [21] inspired the distributed desynchronization algorithms using the PCOs model, with others making advances in generalizing the model to other relevant WSN characteristics such as: limited listening [1], [22], [23] enabling power savings in wireless transceivers; algorithms suitable for multi-hop networks and hidden nodes [1], [14], [20]; scalability to a large number of nodes [5], [18]; and, fast convergence to steady-state [15], [16], [17], [23].

In [24], the authors established worst-case convergence rates and made an advancement with respect to the known lower bounds [17], [22] and order-of-convergence estimates [1], [5], [22]. The main contribution is in showing that the PCO-based desynchronization is equivalent to a gradient descent after a minor modification to a suitable quadratic function. It is then introduced a fast desynchronization based on the Nesterov algorithm. In this paper, we show that viewing the problem as the solution of a linear equation enables a considerable increase in the convergence rate. We also characterize both the convergence rate of the novel approach and the ones using gradient-descent-like methods. The Nesterov method has also been reformulated in [25][26] to accommodate faults.

In the literature, distributed solvers for linear equations will typically store an estimate of ϕ per agent (see an example in a

D. Silvestre is with School of Science and Technology from the NOVA University of Lisbon (FCT/UNL), 2829-516 Caparica, Portugal, with COPELABS from the Lusófona University, and also with the Institute for Systems and Robotics, Instituto Superior Técnico, University of Lisbon. dsilvestre@isr.tecnico.ulisboa.pt

João P. Hespanha is with the Dept. of Electrical and Computer Eng., University of California, Santa Barbara, CA 93106-9560, USA. This material is based upon work supported by the U.S. Office of NavalResearch under the MURI grant No. N00014-16-1-2710. hespanha@ece.ucsb.edu

C. Silvestre is with the Department of Electrical and Computer Engineering of the Faculty of Science and Technology of the University of Macau, Macau, China, on leave from Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal, csilvestre@umac.mo

This work was partially supported by the project MYRG2018-00198-FST of the University of Macau; by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) through Institute for Systems and Robotics (ISR), under Laboratory for Robotics and Engineering Systems (LARSyS) project UIDB/50009/2020, through project PCIF/MPG/0156/2019 FirePuma and through COPELABS, University Lusófona project UIDB/04111/2020.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3130888, IEEE Transactions on Automatic Control

recent publication in [27]). In this problem, that would require additional storage and knowledge of size of the state space. Opting by traditional methods enables a constant memory occupation in comparison with the linear growth of recent decentralized algorithms.

Therefore, the main idea of this paper can be summarized as applying the Gauss-Seidel algorithm to the first-order optimality equation, inspired by the approach used in [28] for the PageRank problem, which promoted a considerable speed-up. The convergence proof of the Gauss-Seidel method has been provided in [29]. The main contributions of this paper can be summarized as:

- We provide closed-form expressions for all parameters that achieve optimal worst-case convergence rate for first-order optimization algorithms, provided the number of transmitters is known.
- We show that the proposal in [24] of using a time-varying parameters version of the Nesterov is outperformed by selecting optimal fixed parameters.
- Proof of convergence and the respective rate is given for the Gauss-Seidel algorithm applied to the desynchronization problem.
- In simulation, the actual performance of the algorithms is analyzed with the Gauss-Seidel algorithm being the best option although scaling poorly for larger networks. The Heavy Ball method (the best worst-case convergence rate) is shown to first increase the error and therefore not being suitable for implementation.

The remainder of the paper is organized as follows. We introduce the desynchronization problem in Section II and the optimal first-order optimization algorithms outperforming the current state-of-the-art in Section III. Section IV describes the solution using the Gauss-Seidel iterative algorithm to solve linear equations with initial simulation results in Section V. Concluding remarks and directions for future work are offered in Section VI.

Notation : The transpose and the spectral radius of a matrix A are denoted by $A^\intercal$ and $\rho(A)$, respectively. For vectors a_i, $(a_1,\dots,a_n) := [a_1^\intercal \ \dots a_n^\intercal]^\intercal$. We let $1_n := [1\dots1]^\intercal$ and $0_n := [0\dots0]^\intercal$ indicate n-dimension vector of ones and zeros, and I_n denotes the identity matrix of dimension n. Dimensions are omitted when no confusion arises. The vector e_i denotes the canonical vector whose components equal zero, except component i that equals one. The symbol $\times$ denotes the cartesian product. The Euclidean norm for vector x is represented as $\|x\|_2 := \sqrt{x^\intercal x}$. We also define $\lambda_i(A)$ as the ith largest eigenvalue of the matrix A.

II. Desynchronization Problem

The desynchronization problem assumes agents turn on periodically in order to conserve battery power. As a consequence of scheduling the uptime to listen to only two neighbors, the interaction networks forms a ring. The main activities of each transmitter is to periodically broadcasts a *fire message* or a *pulse* and listen to the medium for the messages of their neighbors. Each node $i \in \{1,\cdots,n\}$ has a *phase* variable $\theta_i(t)$, as given in [1], [22].

$$\theta_i(t) = \frac{t}{T} + \phi_i(t) \mod 1,$$

where $\phi_i \in [0,1]$ is the so-called phase offset of node i and mod notation stands for the modulo arithmetic. Every node i broadcasts a pulse when its phase reaches the unity (i.e., every T time units) and then resets it to zero. Every time a node receives a pulse, it will adjust its offset ϕ according to an update equation. The work in [24] showed that this problem can be cast as the minimization of a quadratic function, which is given in the next proposition for completeness.

Proposition 1 (Desynchronization as an optimization [24]): Let $\phi^{(k)}$ denote the phases of all nodes at updating cycle k. The state of desynchronization corresponds to the solution of the following optimization problem:

$$\underset{\phi}{\text{minimize}} \quad g(\phi) := \frac{1}{2}\|D\phi - v1_n + \mathsf{e}_n\|_2^2 \tag{1}$$

where $v = 1/n$, 1_n is the vector of ones, $\mathsf{e}_n = (0,0,\cdots,0,1)^\intercal$, and

$$D = \begin{bmatrix} -1 & 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & & \ddots & & \vdots \\ 0 & \cdots & 0 & 0 & -1 & 1 \\ 1 & \cdots & 0 & 0 & 0 & -1 \end{bmatrix}.$$

In [24], it is proposed a parameter-varying version of the Nesterov's method as a fast convergent algorithm. The objective of this paper is to present results showing that considering other types of solutions to the optimization problem in Proposition 1 achieves a higher performance.

III. Optimal Fixed-parameter Nesterov-based Desynchronization Algorithm

As described in the previous section, [24] introduced an optimization formulation to the problem and proposed the use of a time-varying parameter version of the Nesterov method as a fast algorithm to desynchronize the transmitters, called Fast-Desync. Fast-Desync has the advantage of not requiring knowledge of the number of transmitters in the network at the expenses of a sub-optimal choice of parameters. In this section, we present other optimization algorithms and compute closed-form expressions for the parameters and worst-case convergence rates.

In [30], it is shown that the Gradient descent, Heavy-ball and Nesterov methods defined as:

$$\text{Gradient}: \ x^{(k+1)} = x^{(k)} - \beta\nabla g(x^{(k)})$$

$$\text{Heavy-ball}: \ \begin{aligned} x^{(k+1)} &= x^{(k)} - \beta\nabla g(x^{(k)}) \\ &\quad + \gamma(x^{(k)} - x^{(k-1)}) \end{aligned}$$

$$\text{Nesterov}: \ \begin{aligned} x^{(k+1)} &= \xi^{(k)} - \beta\nabla g(\xi^{(k)}) \\ \xi^{(k)} &= (1+\gamma)x^{(k)} - \gamma x^{(k-1)} \end{aligned}$$

can be analyzed as dynamical systems to compute optimal parameters β and γ. This reformulation allows to use standard techniques from linear systems theory to compute the convergence rate in the worst-case trajectory, i.e., the positive constant $\lambda < 1$ such that $\|x^{(k)} - x^\star\| \leq c\lambda^k \|x^{(0)} - x^\star\|$, for some constant $c > 0$, where $x^\star$ corresponds to the steady state value for the system with initial conditions $x^{(0)}$.

However, the results presented in [30] apply to strongly convex functions, which, since g is quadratic, means that matrix Q must satisfy $mI_n \preceq Q \preceq LI_n$, which is equivalent to say that the eigenvalues of Q lie in the interval $[m, L]$ for $m > 0$. In the next lemma, the results are generalized to the case when function g is only convex but the eigenvectors associated with the zero eigenvalues are part of the minima. The result translates that the expressions in [30] for the parameters of the optimization algorithms achieving the best worst-case convergence rate for strongly convex functions are valid, provided that we consider the minimum non-zero eigenvalue as a replacement for the minimum eigenvalue of Q (which would be zero).

Lemma 2: Consider a convex quadratic function $g(x) = x^\intercal Qx + c^\intercal x$ with Hessian matrix $0 \preceq Q \preceq LI_n$ and a subspace $\mathcal{S}$ containing any vector s resulting from a linear combination of eigenvectors of Q associated with zero eigenvalues. If any vector $s \in \mathcal{S}$ is a minimizer of the function, i.e.,

$$\forall s \in \mathcal{S} : g(s) = g^\star,$$

where $g^\star$ is the global minimum of function g, then, optimal parameters achieving the best worst-case convergence rate for linear first-order optimization algorithms depend solely on m and L, where m is the minimum non-zero eigenvalue of Q.

Proof. Given that any linear first-order algorithm can be described by a transition matrix T

$$T = A + BQC,$$

where A, B and C represent the operation of the particular algorithm, its convergence rate depends on the spectra of T. From the statement of the lemma, all vectors in the null space of Q are global minima of function g. Thus, the error analysis needs only to consider initial conditions that do not start in $\mathcal{S}$. Moreover, from the results in [30], the spectra of T is given by the eigenvalues of $A_1 + B_1\lambda_i(Q)C_1$, where A_1, B_1 and C_1 correspond to the matrices A, B, C applied to a function g with domain in $\mathbb{R}$. Given any eigenvalue $\lambda_i(Q)$, we only need to consider $\lambda_i(Q) > 0$ since the initial conditions aligned with the eigenvectors of $\lambda_i(Q) = 0$ correspond to minima of function g, and the conclusion follows. ■

Using Lemma 2, and assuming a known number of transmitters in the network, allows to compute optimal parameters for the three optimization algorithms, which is given in the next theorem. The expressions are given in terms of n which can be hardcoded in the transmitters software.

Theorem 3: Consider the DESYNC problem in (1) with n transmitting sources. Then,

GRADIENT descent with parameter:

$$\beta = \begin{cases} \frac{1}{3-\cos\left(\frac{2\pi}{n}\right)}, & \text{if } n \text{ is even} \\ \frac{1}{2-\cos\left(\frac{2\pi}{n}\right)-\cos\left(\frac{(n-1)\pi}{n}\right)}, & \text{if } n \text{ is odd} \end{cases} \tag{2}$$

achieves worst-case convergence rate ρ_G:

$$\rho_G = \begin{cases} \frac{1+\cos\left(\frac{2\pi}{n}\right)}{3-\cos\left(\frac{2\pi}{n}\right)}, & \text{if } n \text{ is even} \\ \frac{\cos\left(\frac{2\pi}{n}\right)-\cos\left(\frac{(n-1)\pi}{n}\right)}{2-\cos\left(\frac{2\pi}{n}\right)-\cos\left(\frac{(n-1)\pi}{n}\right)}, & \text{if } n \text{ is odd} \end{cases} \tag{3}$$

HEAVY-BALL with parameters:

$$\beta = \begin{cases} \frac{1}{\left(1+\sin\left(\frac{\pi}{n}\right)\right)^2}, & \text{if } n \text{ is even} \\ \frac{1}{\left(\sin\left(\frac{\pi}{n}\right)+\sin\left(\frac{(n-1)\pi}{2n}\right)\right)^2}, & \text{if } n \text{ is odd} \end{cases} \tag{4}$$

and

$$\gamma = \begin{cases} \left(\frac{1-\sin\left(\frac{\pi}{n}\right)}{1+\sin\left(\frac{\pi}{n}\right)}\right)^2, & \text{if } n \text{ is even} \\ \left(\frac{\sin\left(\frac{(n-1)\pi}{2n}\right)-\sin\left(\frac{\pi}{n}\right)}{\sin\left(\frac{(n-1)\pi}{2n}\right)+\sin\left(\frac{\pi}{n}\right)}\right)^2, & \text{if } n \text{ is odd} \end{cases} \tag{5}$$

achieves worst-case convergence rate ρ_H:

$$\rho_H = \begin{cases} \frac{1-\sin\left(\frac{\pi}{n}\right)}{1+\sin\left(\frac{\pi}{n}\right)}, & \text{if } n \text{ is even} \\ \frac{\sin\left(\frac{(n-1)\pi}{2n}\right)-\sin\left(\frac{\pi}{n}\right)}{\sin\left(\frac{(n-1)\pi}{2n}\right)+\sin\left(\frac{\pi}{n}\right)}, & \text{if } n \text{ is odd} \end{cases} \tag{6}$$

NESTEROV with parameters:

$$\beta = \begin{cases} \frac{2}{7-\cos\left(\frac{2\pi}{n}\right)}, & \text{if } n \text{ is even} \\ \frac{2}{4-3\cos\left(\frac{(n-1)\pi}{n}\right)-\cos\left(\frac{2\pi}{n}\right)}, & \text{if } n \text{ is odd} \end{cases} \tag{7}$$

and

$$\gamma = \frac{1 - 2\sqrt{\beta}\sin\left(\frac{\pi}{n}\right)}{1 + 2\sqrt{\beta}\sin\left(\frac{\pi}{n}\right)} \tag{8}$$

achieves worst-case convergence rate ρ_N:

$$\rho_H = 1 - 2\sqrt{\beta}\sin\left(\frac{\pi}{n}\right) \tag{9}$$

Proof. We start by noting that function g has an infinite number of global minima corresponding to one solution rotated in the circle, meaning that g is not strongly convex. Therefore, taking any $x^\star$ and adding a scaled vector from the null space of Q results in a global minimum, i.e.,

$$\forall s \in \mathbb{R} : g(x^\star) = g(x^\star + s\mathbb{1}_n)$$

Using Lemma 2, the algorithms convergence rates depend on $m = \lambda_2(Q)$ and $L = \lambda_n(Q)$. Given the particular structure of Q, both m and L have closed-form expressions:

$$m = 2 - 2\cos\left(\frac{2\pi}{n}\right) \tag{10}$$

whereas

$$L = \begin{cases} 4, & \text{if } n \text{ is even} \\ 2 - 2\cos\left(\frac{(n-1)\pi}{n}\right), & \text{if } n \text{ is odd} \end{cases} \tag{11}$$

Using the optimal parameter value $\beta = \frac{4}{m+L}$ (see [30] or [29]) makes the convergence rate $\rho_G = \frac{\kappa-1}{\kappa+1}$ for $\kappa = L/m$. Replacing for the values of m and L from (10) and (11), we obtain the expressions in (2) and (3).

The HEAVY-BALL parameters can be found when the eigenvalues of T_2 and T_n have equal magnitude, resulting in

$\beta = \frac{4}{(\sqrt{L}+\sqrt{m})^2}$ and $\gamma = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^2$. Replacing the values of m and L and using the fact that $1 - \cos(x) = 2\sin^2(x/2)$ yields the values for the parameters in (4) and (5), and the worst convergence rate is given by $\rho_H = \sqrt{\gamma}$ which yields (6).

The last algorithm corresponding to NESTEROV and solving such that three of the eigenvalues of T_2 and T_n have equal magnitude results in $\beta = \frac{4}{3L+m}$ and $\gamma = \frac{\sqrt{3\kappa+1}-2}{\sqrt{3\kappa+1}+2}$. Replacing the values of m and L and using the fact that $1-\cos(x) = 2\sin^2(x/2)$, after some algebraic manipulations, yields the values for the parameters in (7) and (8), and using the expressions in [29], the worst convergence rate is given by $\rho_N = 1 - \frac{2}{\sqrt{3\kappa+1}}$ which yields (9). ∎

The main consequence of Theorem 3 is that if the number of transmitters to be desynchronized is known, it is possible to achieve optimal worst-case convergence rates. In a sense, the FAST-DESYNC algorithm in [24] using parameters $0 \leq \beta \leq \frac{1}{\max \lambda_i(Q)}$ and $\gamma^{(k)} = \frac{k-1}{k+2}$ is convergent because it uses the suboptimal maximum eigenvalue of Q of 4 for the even case (in the odd case this is a reasonable approximation only as $n \gg 1$) and disregards the minimum eigenvalue. In the next theorem, we find a novel explicit formula for the convergence rate of this time-varying parameter version and show that the convergence rate is governed by $1 - 1/\kappa$ in comparison with $1-1/\sqrt{\kappa}$ when selecting the optimal fixed parameters as seen in Theorem 3. Thus, the current proposal of using the fixed-parameter Nesterov method for cases of known number of nodes n outperforms the current state-of-the-art in [24].

Theorem 4: The Nesterov method [24] with $\beta = \frac{1}{\max \lambda_i(Q)} = \frac{1}{4}$ and $\gamma^{(k)} = \frac{k-1}{k+2}$ has a worst-case convergence rate, at time instant k, $\lambda_{FD}^{(k)}$ given by:

$$\lambda_{FD}^{(k)} = \begin{cases} 1, & \text{if } k = 0 \\ \varphi, & \text{if } k = 1 \\ \left|\varphi\left((1+\gamma^{(k-1)})\lambda_{FD}^{(k-1)} - \gamma^{(k-1)}\lambda_{FD}^{(k-2)}\right)\right|, & \text{if } k \geq 2 \end{cases}$$

for $\varphi = 1 - \frac{1}{\kappa}$. Moreover, $\lambda_{FD}^{(k)}$ is $\mathcal{O}(1/\kappa)$ whereas λ_N is $\mathcal{O}(1/\sqrt{\kappa})$.

Proof. The FAST-DESYNC [24] algorithm can be modeled through the Linear Time-Varying (LTV) model for dynamical systems using the following matrices:

$$A^{(k)} = \begin{bmatrix} (1+\gamma^{(k)})I_n & -\gamma^{(k)}I_n \\ I_n & 0_n \end{bmatrix}, B = \begin{bmatrix} -\beta I_n \\ 0_n \end{bmatrix},$$
$$C^{(k)} = \begin{bmatrix} (1+\gamma^{(k)})I_n & -\gamma^{(k)}I_n \end{bmatrix}.$$

The transition matrix corresponding to the evolution of $x^{(k+1)} - x^\star$ is given by:

$$T^{(k)} = A^{(k)} + BQC^{(k)}.$$

Writing the error equation results in the relationship:

$$x^{(k+1)} - x^\star = \mathcal{T}^{(k+1)}(x^{(0)} - x^\star)$$

with matrix $\mathcal{T}^{(k+1)}$ being the transition matrix from the initial conditions to the current time instant, i.e.,

$$\mathcal{T}^{(k)} = T^{(k)} \cdots T^{(1)}.$$

All matrices $T^{(k)}$ admit the same eigenvalue decomposition using orthogonal matrix U, which allows writing $\mathcal{T}^{(k)}$ as

$$\begin{bmatrix} U & 0_n \\ 0_n & U \end{bmatrix} (A^{(k)} + B\Lambda C^{(k)}) \cdots (A^{(1)} + B\Lambda C^{(1)}) \begin{bmatrix} U & 0_n \\ 0_n & U \end{bmatrix}^\intercal,$$

since $U^\intercal U = I$. Since the spectrum of $\mathcal{T}^{(k)}$ is equivalent to that of $(A^{(k)} + B\Lambda C^{(k)}) \cdots (A^{(1)} + B\Lambda C^{(1)})$, we can study the spectrum of $(A_1^{(k)} + B_1\lambda_i C_1^{(k)}) \cdots (A_1^{(1)} + B_1\lambda_i C_1^{(1)})$ for all eigenvalues $\lambda_i(Q)$, and where matrices with subscript equal to one correspond to setting n to one.

As a consequence of the previous transformation, the convergence rate at each time instant k is given by the product of all matrices from 1 to k in the form:

$$T_1^{(k)} = \begin{bmatrix} (1+\gamma^{(k)})(1-\beta\lambda_i(Q)) & -\gamma^{(k)}(1-\beta\lambda_i(Q)) \\ 1 & 0 \end{bmatrix}$$

and, since $\beta = 1/4$, we get the worst-case value for $1 - \beta\lambda_i(Q) = 1 - \frac{1}{\kappa} = \varphi$. In addition, since $\gamma^{(1)} = 0$ (by definition of the algorithm as there is no momentum term at the first iteration), we get

$$T_1^{(1)} = \begin{bmatrix} \varphi & 0 \\ 1 & 0 \end{bmatrix}$$

which means that any matrix $\mathcal{T}_1^{(k)}$ will have the second column equal to zeros. As a consequence, the eigenvalues are always going to be a zero and the first entry of the matrix since it is a lower triangular. Therefore, the convergence rate for each time instant k evolves according to the sequence:

$$\lambda_{FD}^{(k)} = \varphi\left(\lambda_{FD}^{(k-1)} + \gamma^{(k-1)}(\lambda_{FD}^{(k-1)} - \lambda_{FD}^{(k-2)})\right).$$

Since the minimum of the eigenvalues is achieved after n iterations, i.e., when variable λ_{FD} goes from 1 to below zero, then we can propose the lower bound for the rate that corresponds to:

$$\lambda_{FD}^{(k)} = \varphi\left(\lambda_{FD}^{(k-1)} - \gamma^{(k-1)}\frac{1}{n}\right)$$

which has non-recursive definition given by

$$\lambda_{FD}^{(k)} = \varphi^k(1 - \frac{k}{n})$$

thus, reaching the conclusion since the optimal fixed parameters achieves convergence rate of $1 - \frac{2}{\sqrt{3\kappa+1}}$. ∎

Remark 5: The result at Theorem 4 hints that the convergence rate is slower as the size of the network increases. However, using the expression for the exact convergence rate, we have depicted in Fig. 1 the comparison between the rates for three sizes of networks. In each case, the rate achieved with the optimal fixed parameter is always faster than that using the time-varying version.

IV. DESYNCHRONIZATION USING GAUSS-SEIDEL ITERATIONS

In the previous section, we have shown that the current state-of-the-art underperforms in comparison with setting an optimal fixed-parameter for the Nesterov method with the expressions being given in Theorem 3. However, in some scenarios, it will be infeasible to have all transmitters know the number of nodes

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3130888, IEEE Transactions on Automatic Control

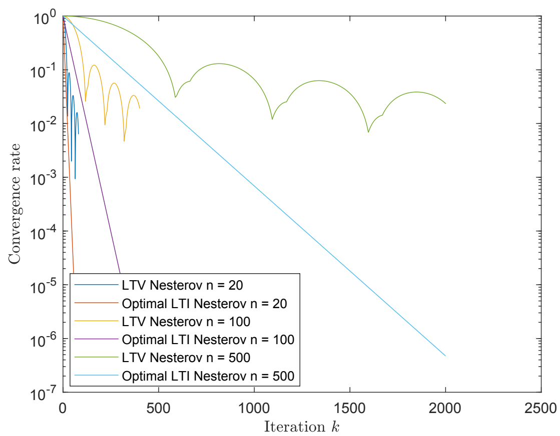

Fig. 1. Logarithmic evolution of the convergence rate for the time-varying Nesterov (LTVnesterov) and optimal fixed parameter Nesterov for networks sizes of 20, 100 and 500.

in the entire network. In this section, we present results when viewing the problem in (1) as the solution of a linear equation. We first present the Gauss-Seidel algorithm for completeness and then apply it to $\nabla g(\phi) = 0$ in order to obtain a faster update rule without the need to set up parameters.

For a general system $Ax = b$, with $A = L + D + U$ decomposed in lower, diagonal and upper matrices, the Gauss-Seidel method has the following update rule:

$$x(k+1) = (L+D)^{-1}(b - Ux(k)) \tag{12}$$

which, by taking advantage of the triangular form of $L+D$, can be sequentially updated for each i using forward substitution, leading to the desynchronization algorithm:

$$\begin{aligned} \phi_1^{(k+1)} &= \frac{1}{2}\left(1 - \phi_2^{(k)} - \phi_n^{(k)}\right) \\ \phi_i^{(k+1)} &= \frac{1}{2}\left(-\phi_{i-1}^{(k+1)} - \phi_{i+1}^{(k)}\right), 2 \le i \le n-1 \\ \phi_n^{(k)} &= \frac{1}{2}\left(-1 - \phi_1^{(k+1)} - \phi_{n-1}^{(k+1)}\right) \end{aligned} \tag{13}$$

which requires communication with the immediate neighbors akin the original problem and exploits the inherent sequential behavior of the DESYNC algorithm. In this setup, nodes use the most updated values for the phases.

The next theorem provides the exponential rate of convergence of the iteration in (13).

Theorem 6 (Convergence Rate of Gauss-Seidel): The iterative method (13) asymptotically converges to a desynchronization state with exponential convergence rate λ_{GS}, i.e.,

$$\phi^{(k+1)} - \phi^\star \le \lambda_{GS}^{k+1}(\phi^{(0)} - \phi^\star) \tag{14}$$

where

$$\lambda_{GS} = |\lambda_2(T_{GS})|$$

and

$$T_{GS} = \sum_{j=0}^{n-1} \left(\frac{1}{2}\right)^{j+1} E^j E^\intercal, \tag{15}$$

$$E = \begin{bmatrix} 0_{n-1}^\intercal & 0 \\ I_{n-1} & 0_{n-1} \end{bmatrix} + e_n e_1^\intercal \tag{16}$$

with $|\lambda_n(T_{GS})| \le |\lambda_{n-1}(T_{GS})| \le \cdots \le |\lambda_1(T_{GS})|$.

Proof. The inequality in (14) comes directly from seeing the Gauss-Seidel algorithm as a linear time-invariant system where $T_{GS} := -(D+L)^{-1}U$ is the transition matrix for a general linear equality $Ax = b$ as in (12).

The first step in the proof consists of writing the matrix T_{GS} for the DESYNC problem. Given the partition $A = L+D+U$, the matrix T_{GS} has the following expression:

$$\begin{aligned} T_{GS} &= (2I_n - E)^{-1}E^\intercal \\ &= \frac{1}{2}(I_n - \frac{1}{2}E)^{-1}E^\intercal \end{aligned} \tag{17}$$

with the strictly lower triangular matrix E being defined as in (16).

We remark that

$$(I_n + N)^{-1} = I_n + \sum_{k=1}^{n-1}(-1)^k N^k$$

for a general strictly lower triangular matrix N. Using the above equality and after some algebraic manipulations, (17) simplifies to (15).

The second step is to show stability by proving that the spectral radius of T_{GS} is within the unit circle, i.e., $\rho(T_{GS}) \le 1$. Matrix T_{GS} is row stochastic since its elements are trivially nonnegative and

$$\begin{aligned} T_{GS}1_n &= \left(\sum_{j=0}^{n-1}\left(\frac{1}{2}\right)^{j+1} E^j\right)(1_n + D^\intercal e_n) \\ &= \frac{1}{2}(1_n + D^\intercal e_n) + \frac{1}{2^2}\begin{bmatrix} 0 \\ 2 \\ 1_{n-3} \\ 3 \end{bmatrix} + \frac{1}{2^3}\begin{bmatrix} 0_2 \\ 2 \\ 1_{n-3} \end{bmatrix} \\ &\quad + \frac{1}{2^4}\begin{bmatrix} 0_3 \\ 2 \\ 1_{n-4} \end{bmatrix} + \cdots + \frac{1}{2^n}\begin{bmatrix} 0_{n-1} \\ 2 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ \frac{1}{2} + \frac{2}{2^2} \\ \sum_{j=1}^{2}\frac{1}{2^j} + \frac{2}{2^3} \\ \vdots \\ \sum_{j=1}^{n-2}\frac{1}{2^j} + \frac{2}{2^{n-1}} \\ \frac{3}{2^2} + \sum_{j=3}^{n-1}\frac{1}{2^j} + \frac{2}{2^n} \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ \frac{1}{2} + \frac{1}{2} \\ 1 - \frac{1}{2^2} + \frac{1}{2^2} \\ \vdots \\ 1 - \frac{1}{2^{n-2}} + \frac{1}{2^{n-2}} \\ \frac{3}{2^2} - \frac{1}{2^{n-1}} + \frac{1}{2^2} + \frac{1}{2^{n-1}} \end{bmatrix} \\ &= 1_n. \end{aligned}$$

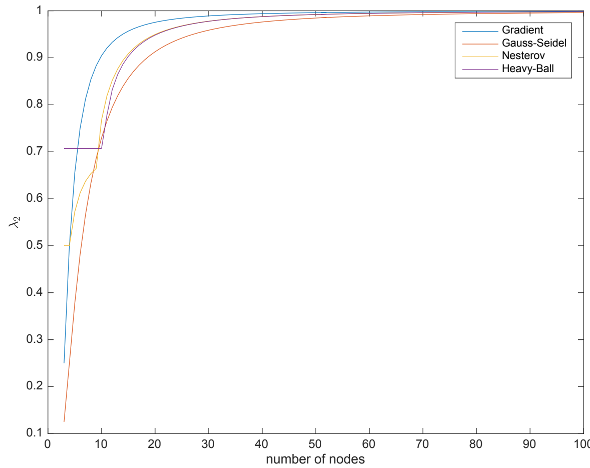

Fig. 2. The convergence rate λ_2 for the different algorithms depending on the number of nodes n.

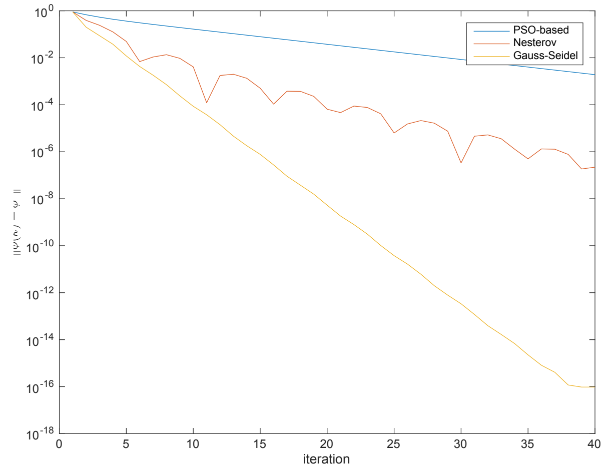

Fig. 3. Logarithmic evolution of the error norm for the PCO-based, Nesterov and Gauss-Seidel algorithms.

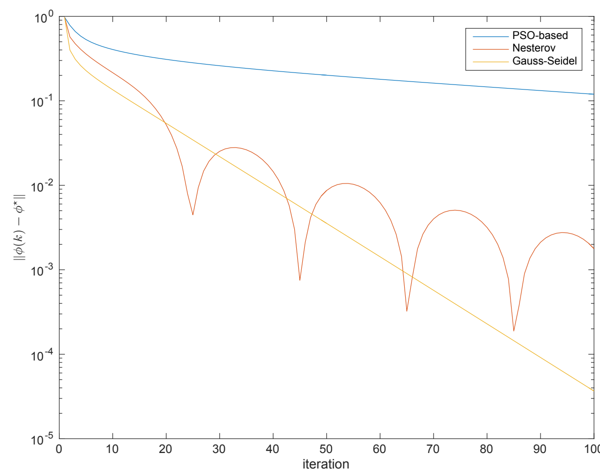

Fig. 4. Logarithmic evolution of the error norm for the PCO-based, Nesterov and Gauss-Seidel algorithms for the 20 node network.

By noticing that the first row is equal to two times the nth minus the $n-1$th rows, the following equality is true

$$UT_{GS} = \begin{bmatrix} 0 & \mathbb{0}_{n-1}^\intercal \\ \mathbb{0}_{n-1} & T_{GS}^{sub} \end{bmatrix}$$

where

$$U = \left[\begin{array}{c|ccc} 1 & \mathbb{0}_{n-3}^\intercal & 1 & -2 \\ \hline \mathbb{0}_{n-1} & & I_{n-1} & \end{array}\right]$$

and the matrix T_{GS}^{sub} is a submatrix of T_{GS} obtained by removing the first row and column. Since the matrix U implements elementary row operations, the multiplication has no effect on the spectra of T_{GS}, meaning that $\lambda_i(T_{GS}) = \lambda_i(UT_{GS}), \forall 1 \leq i \leq n$. In particular, from the format of UT_{GS}, it follows that

$$\{\lambda_i(T_{GS}), 1 \leq i \leq n\} = \{\lambda_i(T_{GS}^{sub}), 1 \leq i \leq n-1\} \cup \{0\}$$

Similarly, T_{GS}^{sub} remains row stochastic and its support graph is strongly connected since the last row and columns are full (meaning that the correspondent node would have edges to and from all the remaining nodes in the graph). As a consequence, $\lambda_1(T_{GS}^{sub}) = 1$ and $|\lambda_2(T_{GS}^{sub})| < 1$ by the Perron-Frobenius Theorem and the conclusion about the convergence rate also follows. ■

V. Simulation results

In this section, simulations are presented using the toolbox in [31] in order to illustrate whether the theoretical rates represent an advantage in practical sense. All simulations have considered an initial starting phase state $\phi^{(0)} = \mathbb{1}_n/n$ corresponding to all nodes sharing the same phase and being completely synchronized.

Figure 2 compares the evolution of the convergence rates for Gradient with $\beta = \frac{1}{4}$ (which is equivalent to the PCO-based Desync when $\beta = \frac{\alpha}{2}$ as demonstrated in [24]), Gauss-Seidel, Nesterov and Heavy-Ball for the fixed parameters $\beta = \frac{1}{4}, \gamma = \frac{1}{2}$. This hints at the fact that indeed considering both optimization methods and iterative algorithms for solving linear equations yields improvements in performance in comparison with the PCO model. As expected, as the number of nodes increases, so does the convergence rate, which is approaching the unity as n grows to infinity.

An important remark is that the version of the Nesterov method proposed in [24] has time-varying parameters (in particular $\gamma = \frac{k-1}{k+2}$) that might contribute to increase the speed of convergence. In order to compare the method proposed in [24], a simulation of a $n = 5$ node network was conducted and the logarithm of the error norm is presented is Fig. 3.

Figure 3 shows that the Gauss-Seidel iteration achieves a faster convergence at a fixed rate in comparison with the algorithm in [24]. Both methods present a clear advantage when compared to the PCO-based method with parameter $\alpha = 0.2$. Additional simulations were conducted to assess the potential advantage of the Nesterov method with a time-varying parameter. A network of $n = 20$ nodes was also simulated and the results are depicted in Fig. 4.

The main observation from the evolution of the error in Figs. 3 and 4 is that as n increases, the behavior of the error norm changes. For small networks, the Gauss-Seidel method

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3130888, IEEE Transactions on Automatic Control

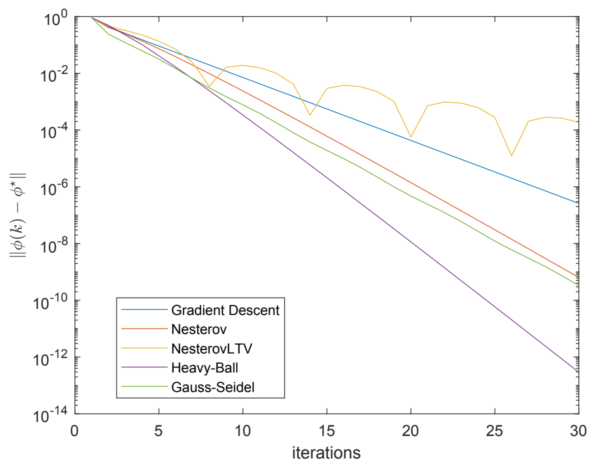

Fig. 5. Logarithmic evolution of the error norm for the PCO-based (Gradient Descent), Nesterov, LTV Nesterov, Heavy-Ball and Gauss-Seidel algorithms for a 6 node network.

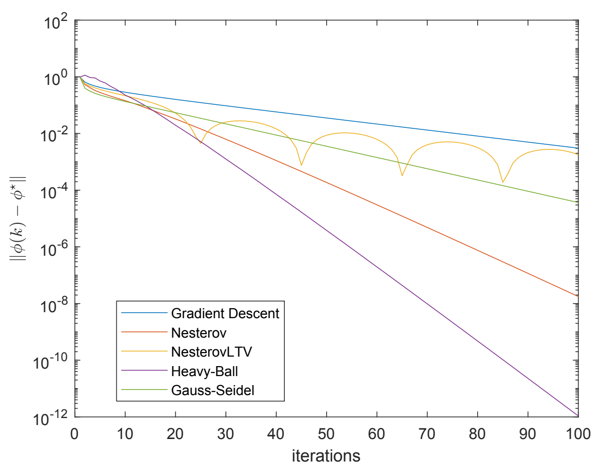

Fig. 6. Logarithmic evolution of the error norm for the PCO-based (Gradient Descent), Nesterov, LTV Nesterov, Heavy-Ball and Gauss-Seidel algorithms for a 20 node network.

outperforms the Nesterov algorithm. When increasing n, the error decreases faster using the Gauss-Seidel up to a small tolerance and then the Nesterov method becomes faster. The observed oscillations tend to fade for larger n.

The simulations presented so far only considered algorithms for which there is no knowledge of the number of transmitters n. In the remainder of the simulations, both the Gauss-Seidel and the LTV version of Nesterov from [24] are compared against the Nesterov and Heavy-Ball algorithms selecting optimal parameters from Theorem 3. The initial state is set to $1_n/n$ and we report the same error function as previously.

In Fig. 5 it is depicted the error evolution for a small size network of 6 nodes. In small networks, simulations show that the LTV Nesterov has the worst performance while, as expected, the Heavy-Ball algorithm has the best performance given that it is the fastest for quadratic functions of all methods. As proven in Theorem 4, the convergence of LTV Nesterov is not monotonous. The Gauss-Seidel outperforms the optimal Nesterov method but with a similar behavior.

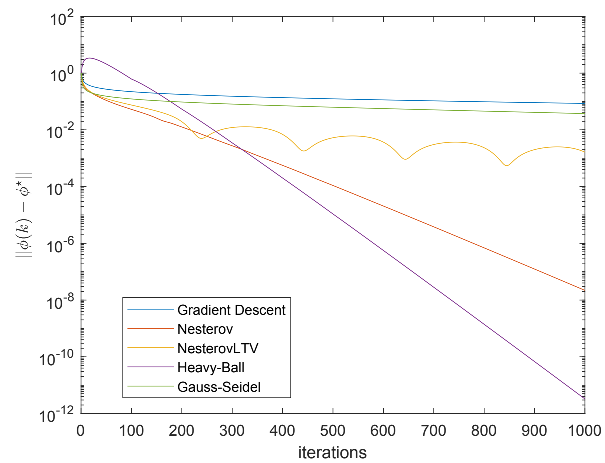

Fig. 7. Logarithmic evolution of the error norm for the PCO-based (Gradient Descent), Nesterov, LTV Nesterov, Heavy-Ball and Gauss-Seidel algorithms for a 200 node network.

In order to test for medium-sized networks, a similar simulation was conducted for a 20 node network. The Gradient with optimal parameter is the slowest and the oscillatory behavior of the LTV Nesterov is heightened and still underperfoms in comparison with the Gauss-Seidel. For this case, both the Nesterov and the Heavy-Ball methods achieve a better convergence since it scales with $1/\sqrt{\kappa}$. Another curious fact that starts to emerge is the bad performance of the Heavy-Ball in the beginning of the simulation.

In order to illustrate the behavior of the algorithms for large networks, a 200 transmitter network is also simulated with the error evolution being presented in Fig. 7. For such cases, the behavior of the Gauss-Seidel approaches the optimal Gradient Descent albeit faster but both underperform in comparison with the Nesterov and Heavy-Ball. The LTV Nesterov has a performance in between these two classes and still maintains its oscillatory behavior. The initial increase in the error for the Heavy-Ball is noticeable which might discourages its application for large networks and error tolerances around 10^{-3} or 10^{-4} for which the Nesterov method produces equivalent results without the initial increase in the error.

VI. Discussion and Future Work

In this paper, the desynchronization problem was addressed in the format of an optimization problem. In this formulation, it is possible to apply different distributed optimization algorithms and also, given the quadratic objective function, to use an iterative algorithm to solve linear equations to find the solution of zero gradient.

The analysis of the convergence rate is carried out by writing the optimization algorithms as Linear Time-Invariant (LTI) systems and analyzing the correspondent transition matrix. The Gauss-Seidel is shown to be convergent for the desynchronization problem and its convergence rate as the second largest eigenvalue in magnitude of a row stochastic matrix.

When no information regarding the number of transmitters is known, the Gauss-Seidel algorithm performs better than the

PCO and the LTV Nesterov but its performances degrades for large networks. In contrast, when the number of nodes in the network can be determined, it is shown exact closed-form expressions for the parameters of both the Gradient descent, Nesterov and Heavy-Ball that considerable improve performance in comparison with the PCO-based, LTV Nesterov and Gauss-Seidel. For large networks, the Heavy-Ball might not be desirable since its error increases in an initial phase and takes approximately the same number of iterations to get errors around 10^{-3} than the Nesterov.

In future work, the results presented herein point towards the need to investigate other iterative solvers for linear equations that may bring additional gains in performance for medium or large-sized networks. Other optimization algorithms both linear and nonlinear can also be implemented and optimal parameters may be derived following similar reasoning used in this paper. These questions are of utmost importance since both the optimization options and the iterative solvers for linear equations present faster convergence rates than the PCO-based algorithm that constitutes the IEEE standard. A last direction of future work is to use the methods proposed herein to address the multi-channel case of the desynchronization problem.

REFERENCES

[1] J. Degesys and R. Nagpal, "Towards desynchronization of multi-hop topologies," in *Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, Oct 2008, pp. 129–138.

[2] A. Tinka, T. Watteyne, and K. Pister, *A Decentralized Scheduling Algorithm for Time Synchronized Channel Hopping*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 201–216.

[3] T. Watteyne, X. Vilajosana, B. Kerkez, F. Chraim, K. Weekly, Q. Wang, S. Glaser, and K. Pister, "OpenWSN: a standards-based low-power wireless development environment," *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 5, pp. 480–493, 2012.

[4] X. Vilajosana, Q. Wang, F. Chraim, T. Watteyne, T. Chang, and K. S. J. Pister, "A Realistic Energy Consumption Model for TSCH Networks," *IEEE Sensors Journal*, vol. 14, no. 2, pp. 482–489, Feb 2014.

[5] R. Pagliari and A. Scaglione, "Scalable network synchronization with pulse-coupled oscillators," *IEEE Transactions on Mobile Computing*, vol. 10, no. 3, pp. 392–405, March 2011.

[6] D. Buranapanichkit and Y. Andreopoulos, "Distributed time-frequency division multiple access protocol for wireless sensor networks," *IEEE Wireless Communications Letters*, vol. 1, no. 5, pp. 440–443, October 2012.

[7] D. Antunes, D. Silvestre, and C. Silvestre, "Average consensus and gossip algorithms in networks with stochastic asymmetric communications," in *50th IEEE Conference on Decision and Control and European Control Conference*, Dec 2011, pp. 2088–2093.

[8] D. Silvestre, J. P. Hespanha, and C. Silvestre, "Broadcast and gossip stochastic average consensus algorithms in directed topologies," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 2, pp. 474–486, June 2019.

[9] R. Ribeiro, D. Silvestre, and C. Silvestre, "Decentralized control for multi-agent missions based on flocking rules," in *CONTROLO 2020*, J. A. Gonçalves, M. Braz-César, and J. P. Coelho, Eds. Cham: Springer International Publishing, 2021, pp. 445–454.

[10] R. Ribeiro, D. Silvestre, and C. Silvestre, "A rendezvous algorithm for multi-agent systems in disconnected network topologies," in *2020 28th Mediterranean Conference on Control and Automation (MED)*, Sep. 2020, pp. 592–597.

[12] O. Simeone, U. Spagnolini, Y. Bar-Ness, and S. H. Strogatz, "Distributed synchronization in wireless networks," *IEEE Signal Processing Magazine*, vol. 25, no. 5, pp. 81–97, September 2008.

[11] IEEE, "IEEE Standard for Local and metropolitan area networks– Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer," *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–225, April 2012.

[13] A. Patel, J. Degesys, and R. Nagpal, "Desynchronization: The theory of self-organizing algorithms for round-robin scheduling," in *First International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2007)*, July 2007, pp. 87–96.

[14] A. Motskin, T. Roughgarden, P. Skraba, and L. Guibas, "Lightweight coloring and desynchronization for networks," in *IEEE INFOCOM 2009*, April 2009, pp. 2383–2391.

[15] C. M. Lien, S. H. Chang, C. S. Chang, and D. S. Lee, "Anchored desynchronization," in *IEEE INFOCOM 2012*, March 2012, pp. 2966–2970.

[16] R. Leidenfrost and W. Elmenreich, "Firefly clock synchronization in an 802.15.4 wireless network," *EURASIP Journal on Embedded Systems*, vol. 2009, no. 1, p. 186406, Jul 2009.

[17] J. Klinglmayr and C. Bettstetter, "Self-organizing synchronization with inhibitory-coupled oscillators: Convergence and robustness," *ACM Trans. Auton. Adapt. Syst.*, vol. 7, no. 3, pp. 30:1–30:23, Oct. 2012.

[18] Y.-W. Hong and A. Scaglione, "A scalable synchronization protocol for large scale sensor networks and its applications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 5, pp. 1085–1099, May 2005.

[19] S. Choochaisri, K. Apicharttrisorn, K. Korprasertthaworn, P. Taechalertpaisarn, and C. Intanagonwiwat, "Desynchronization with an artificial force field for wireless networks," *SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 2, pp. 7–15, Mar. 2012.

[20] 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*, vol. 182, no. 1, pp. 77 – 92, 2012, nature-Inspired Collective Intelligence in Theory and Practice.

[21] 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.

[22] R. Pagliari, Y. W. P. Hong, and A. Scaglione, "Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 564–575, May 2010.

[23] Y. Wang, F. Nunez, and F. J. Doyle, "Energy-efficient pulse-coupled synchronization strategy design for wireless sensor networks through reduced idle listening," *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5293–5306, Oct 2012.

[24] N. Deligiannis, J. F. C. Mota, G. Smart, and Y. Andreopoulos, "Fast desynchronization for decentralized multichannel medium access control," *IEEE Transactions on Communications*, vol. 63, no. 9, pp. 3336–3349, Sept 2015.

[25] D. Silvestre, J. H. Hespanha, and C. Silvestre, "Resilient desynchronization for decentralized medium access control," in *to appear in 2020 IEEE Conference on Decision and Control (CDC)*, December 2020.

[26] D. Silvestre, J. P. Hespanha, and C. Silvestre, "Resilient desynchronization for decentralized medium access control," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 803–808, July 2021.

[27] S. Mou, J. Liu, and A. S. Morse, "A distributed algorithm for solving a linear algebraic equation," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2863–2878, Nov 2015.

[28] D. Silvestre, J. Hespanha, and C. Silvestre, "A pagerank algorithm based on asynchronous gauss-seidel iterations," in *2018 Annual American Control Conference (ACC)*, June 2018, pp. 484–489.

[29] D. Silvestre, J. Hespanha, and C. Silvestre, "Desynchronization for decentralized medium access control based on gauss-seidel iterations," in *2019 American Control Conference (ACC)*, July 2019, pp. 4049–4054.

[30] L. Lessard, B. Recht, and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, 2016.

[31] D. Silvestre, "Optool — an optimization toolbox for iterative algorithms," *SoftwareX*, vol. 11, p. 100371, 2020.