

# Desynchronization for Decentralized Medium Access Control based on Gauss-Seidel Iterations

Daniel Silvestre, João Hespanha and Carlos Silvestre

**Abstract**—We address the Desynchronization problem of achieving an equally spaced transmission schedule in a cooperative fashion. This problem arises in a shared medium communication and is of importance to achieve a fair multiple access schedule at the Medium Access Control (MAC) layer in the context of Wireless Sensor Networks (WSNs). In this paper, we investigate the convergence rate of different optimization algorithms and the potential benefits of addressing the problem as a solution of a set of linear equations. Initial results suggest that the Gauss-Seidel method can yield a faster convergence than previously proposed methods that employ a version of the Nesterov’s method. Our approach also poses an interesting path for future research given the benefits of using other more advanced methods to solve systems of linear equations. Through simulations, we provide evidence to support future research on optimizing the parameter selection and also on categorizing the conditions under which one solution might be better in detriment of another.

**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].

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 Hopping (TSCH) [2] protocol has been established as state-of-the-art in the IEEE 802.15.4e-2012 standard [6].

A large body of research [1], [2], [4], [7], [8], [9], [10], [11], [12], [13] exists regarding distributed desynchronization algorithms that can enable a decentralized WSN MAC-layer coordination. These algorithms are inspired by biological agents modeled as Pulse-Coupled Oscillators (PCOs) and each node acts as a timing mechanism with a periodic pulsing that is adjusted based on the timings of pulses received from other nodes.

Distributed desynchronization algorithms using the PCOs model are based on the seminal work by Mirollo and Strogatz [14] to model the PCO dynamics and often present different characteristics that are of relevance to WSNs: limited listening [1], [15], [16] that enables power saving in wireless transceivers; algorithms capable of dealing with multi-hop networks and hidden nodes [1], [8], [13]; scalability to a large number of nodes [4], [12]; and, fast convergence to steady-state [9], [10], [11], [16].

Prior to the work in [17], the convergence speed of the desynchronization algorithm was not theoretically studied and was provided by means of simulations or empirical measures. The state-of-the-art at that time included lower bounds [11], [15] and order-of-convergence estimates [1], [4], [15]. In [17], worst-case convergence rates are established and the algorithm was applied to the case of multichannel. In addition, the PCO-based desynchronization is shown to be identical to a gradient descent algorithm with a minor modification. Building on that result, a fast-converging algorithm is provided using the Nesterov method due to its worst-case convergence rate. However, that bound represents the worst performance over all differentiable functions  $g$  and not necessarily a good fit for the problem at hand. Here we exploit the basis framework developed in [17] to investigate the use of iterative algorithms for solving linear equations as an alternative approach to address the desynchronization problem. In the process, we are able to characterize both the convergence rate of the novel approach and the ones using gradient-descent-like methods.

The idea presented in this paper is to use a standard algorithm for distributed solution of linear equations to gain in performance when compared to the NESTEROV method. The approach discussed herein is to introduce the Gauss-Seidel algorithm and apply it to solving the linear equation  $\nabla g(\phi) = 0$ , inspired by the approach used in [18] for the

D. Silvestre is with the Department of Electrical and Computer Engineering of the Faculty of Science and Technology of the University of Macau, Macau, China, and with the Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal, [dsilvestre@isr.ist.utl.pt](mailto:dsilvestre@isr.ist.utl.pt)

João P. Hespanha is with the Dept. of Electrical and Computer Eng., University of California, Santa Barbara, CA 93106-9560, USA. J. Hespanha was supported by the the National Science Foundation under Grants No. EPCN-1608880 and CNS-1329650. [hespanha@ece.ucsb.edu](mailto: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](mailto:csilvestre@umac.mo)

This work was partially supported by the project MYRG2016-00097-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 UID/EEA/50009/2019.

PageRank problem.

The main contributions of this paper are:

- We provide convergence rates for different types of gradient-descent-based methods by leveraging its analysis as dynamical systems.
- We study the convergence properties of the Gauss-Seidel iteration when applied to the linear equation defining the optimal solution to desynchronization.

*Notation* : The transpose and the spectral radius of a matrix  $A$  are denoted by  $A^\top$  and  $\rho(A)$ , respectively. We let  $\mathbf{1}_n := [1 \dots 1]^\top$  and  $\mathbf{0}_n := [0 \dots 0]^\top$  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  $\mathbf{e}_i$  denotes the canonical vector whose components equal zero, except component  $i$  that equals one. The notation  $\text{diag}(v)$  is a diagonal matrix with its diagonal equal to the vector  $v$ . The Euclidean norm for vector  $x$  is represented as  $\|x\|_2 := \sqrt{x^\top x}$ .

## II. REVIEW OF PCO-BASED DESYNCHRONIZATION

A PCO-based desynchronization algorithm assumes a ring network where each agent broadcasts periodically a *fire message* or a *pulse*. Such dynamics is modeled by a *phase* variable  $\theta_i(t)$  for each node  $i \in \{1, \dots, n\}$  which is defined [1], [15]

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

where  $\phi_i \in [0, 1]$  is the so called phase offset of node  $i$  and mod represents the modulo arithmetics. The idea behind (1) is to consider the phase going from zero to one along a circle. Every node  $i$  broadcasts a pulse when its phase reaches the unity (i.e., every  $T$  time units) and then resets it to zero. When the nodes listen to other nodes pulses, they adjust their  $\phi$  variable according to an update equation based on the PCO dynamics [14]. The algorithm DESYNC [1] assumes an ordered initial state  $0 \leq \theta_1(0) \leq \dots \leq \theta_n(0)$  which is kept assuming perfect beacon reception. The phase is updated based on the node phase neighbors  $\theta_{i-1}$  and  $\theta_{i+1}$  after receiving the beacon from node  $i-1$

$$\theta'_i(t_{i-1}) = (1-\alpha)\theta_i(t_{i-1}) + \alpha \frac{\theta_{i-1}(t_{i-1}) + \theta_{i+1}(t_{i-1})}{2} \quad (2)$$

where  $t_{i-1}$  is the time instant at which fire message from node  $i-1$  was received by node  $i$  and assuming we consider the nodes to be placed on a circle such that node 1 and  $n$  are neighbors. The *jump-phase parameter*  $\alpha \in (0, 1)$  translates how much node  $i$  changes its phase in response to the phase of its neighbors.

The algorithm DESYNC implementing the update equation (2) has the following interesting features: each node updates its phase exactly once during each updating cycle; there is no need for any of the nodes to know  $n$ ; requires limited listening as only two neighbors are used. The update in (2) has also been shown to converge to the state of desynchrony at time  $\bar{t}$ , meaning that for, all  $t \geq \bar{t}$ , the time between consecutive beacons becomes  $T/n$  apart from a small  $\epsilon$  error.

The work in [17] introduced a slight modification to the DESYNC algorithm that is summarized in the following assumption, which uses the superscript  $k$  to denote the update cycle of the variable and drops the time dependency.

*Assumption 1 (update variation [17]):* In DESYNC, node  $n$  updates its phase at update cycle  $k$  using  $\theta_{n-1}^{(k-1)}$  instead of  $\theta_{n-1}^{(k)}$ .

Assumption 1 allows to combine (1) and (2) into

$$\begin{aligned} \phi_1^{(k)} &= (1-\alpha)\phi_1^{(k-1)} + \frac{\alpha}{2} \left( \phi_2^{(k-1)} + \phi_n^{(k-1)} - 1 \right) \\ \phi_i^{(k)} &= (1-\alpha)\phi_i^{(k-1)} + \frac{\alpha}{2} \left( \phi_{i-1}^{(k-1)} + \phi_{i+1}^{(k-1)} \right), 2 \leq i \leq n-1 \\ \phi_n^{(k)} &= (1-\alpha)\phi_n^{(k-1)} + \frac{\alpha}{2} \left( \phi_{n-1}^{(k-1)} + \phi_1^{(k-1)} + 1 \right) \end{aligned} \quad (3)$$

where  $\phi^{(k)} = (\phi_1^{(k)}, \phi_2^{(k)}, \dots, \phi_n^{(k)}) \in \mathbb{R}^n$  is a vector containing all the phases and  $d = (1, 0, \dots, 0, -1) \in \mathbb{R}^n$ . The format in (3) is similar to a consensus system ([19], [20]) with an input bias. That fact enables the use of standard tools to analyze those systems and provide convergence rate results. It can be further converted into an optimization in the result that follows.

*Proposition 2 (Desynchronization as optimization [17]):* Let  $\phi^{(k)}$  denote the phases of all nodes at updating cycle  $k$ . If Assumption 1 holds, then DESYNC (3) can be viewed as the steepest descent method applied to

$$\underset{\phi}{\text{minimize}} \quad g(\phi) := \frac{1}{2} \|D\phi - v\mathbf{1}_n + \mathbf{e}_n\|_2^2 \quad (4)$$

where  $v = 1/n$ ,  $\mathbf{1}_n$  is the vector of ones,  $\mathbf{e}_n = (0, 0, \dots, 0, 1)$ , and

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

Specifically, the updates in (3) can be written as

$$\phi^{(k)} = \phi^{(k-1)} - \frac{\alpha}{2} \nabla g(\phi^{(k-1)}).$$

In [17], the authors propose using the Nesterov's fast gradient algorithm as a means to get a fast convergent algorithm. The worst-case convergence rate for Nesterov's method to produce a solution  $\bar{\phi}$  that satisfies  $g(\bar{\phi}) - g(\phi^*) \leq \epsilon$ , where  $\phi^*$  minimizes  $g$ , is  $O(1/\sqrt{\epsilon})$  whereas for the steepest descent method can only achieve a convergence rate  $O(1/\epsilon)$ . Since there are multiple solutions to the desynchronization problem,  $x_*$  can be taken as the state separated by  $1/n$  with the first entry equal to  $x_1^{(k)}$  when computing the error at time  $k$ . We omit the dependence of time of  $x^*$  since for analysis purposes it suffices to know that any phase vector with entries distanced by  $1/n$  is a minimizer of  $g$  or a fixed-point of any of the optimization algorithms. The objective of this paper is to present results showing that considering other types of solutions to the optimization problem in Proposition 2 achieves a higher performance.

### III. DESYNCHRONIZATION AS AN OPTIMIZATION PROBLEM

In the previous section a brief review of the PCO-based DESYNC algorithm was presented and how it has been shown that under Assumption 1 it is equivalent to the steepest descent algorithm. In this section, using the optimization problem format for the DESYNC, other algorithms are investigated and their correspondent theoretical convergence rates are given.

We introduce the compact notation

$$\left[ \begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] \quad (5)$$

to represent the dynamical system

$$\begin{aligned} x^{(k+1)} &= Ax^{(k)} + Bu^{(k)} \\ y^{(k)} &= Cx^{(k)} + Du^{(k)} \end{aligned}$$

with state vector  $x^{(k)} \in \mathbb{R}^m$ , output vector  $y^{(k)} \in \mathbb{R}^n$  and input vector  $u^{(k)} \in \mathbb{R}^n$ .

In [21], it is shown how different optimization algorithms can be casted as dynamical systems in the form of (5) that we summarize in the next proposition.

*Proposition 3 (Optimization algorithms as linear systems):* Given the problem

$$\underset{x}{\text{minimize}} \quad g(x)$$

for a differentiable function  $g$ , 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 expressed in the format (5) as:

$$\text{GRADIENT : } \left[ \begin{array}{c|c} I_n & -\beta I_n \\ \hline I_n & 0_n \end{array} \right] \quad (6a)$$

$$\text{HEAVY-BALL : } \left[ \begin{array}{cc|c} (1 + \gamma)I_n & -\gamma I_n & -\beta I_n \\ I_n & 0_n & 0_n \\ \hline I_n & 0_n & 0_n \end{array} \right] \quad (6b)$$

$$\text{NESTEROV : } \left[ \begin{array}{cc|c} (1 + \gamma)I_n & -\gamma I_n & -\beta I_n \\ I_n & 0_n & 0_n \\ \hline (1 + \gamma)I_n & -\gamma I_n & 0_n \end{array} \right] \quad (6c)$$

where  $u^{(k)} = \nabla g(y^{(k)})$ .

Expressing an optimization algorithm as a dynamical system facilitates establishing its convergence rate. The definition used for the rate of convergence is the positive constant  $\lambda < 1$  such that  $x^{(k)} - x^* \leq \lambda^k(x^{(0)} - x^*)$ . This result is given in the next theorem.

*Theorem 4 (Convergence Rate and Stability):* The optimization methods of GRADIENT descent, HEAVY-BALL and NESTEROV applied to the desynchronization problem in (4) converge if and only if:

GRADIENT :  $\alpha \in (0, 1)$  with rate

$$\lambda_G = \max\left\{ \left| 1 - \frac{\alpha}{2} \min \lambda_i \right|, \left| 1 - \frac{\alpha}{2} \max \lambda_i \right| \right\}$$

HEAVY-BALL :  $\lambda_H < 1$

$$\lambda_H = \begin{cases} \frac{1}{2} |1 + \gamma - \beta \lambda_i| + \frac{1}{2} \sqrt{\Delta}, & \text{if } \Delta \geq 0 \\ \sqrt{\gamma}, & \text{otherwise.} \end{cases}$$

$$\text{where } \Delta = (1 + \gamma)^2(1 - \beta \lambda_i)^2 - 4\gamma$$

NESTEROV :  $\lambda_N < 1$

$$\lambda_N = \begin{cases} \frac{1}{2} |(1 + \gamma)(1 - \beta \lambda_i)| + \frac{1}{2} \sqrt{\Delta}, & \text{if } \Delta \geq 0 \\ \sqrt{\gamma(1 - \beta \lambda_i)}, & \text{otherwise.} \end{cases}$$

$$\text{where } \Delta = (1 + \gamma)^2(1 - \beta \lambda_i)^2 - 4\gamma(1 - \beta \lambda_i)$$

with convergence rates  $\lambda_G$ ,  $\lambda_H$ , and  $\lambda_N$ , respectively,

*Proof:* Proposition 3 allow us to write each of the optimization algorithms into the format in (5). Since problem (4) is quadratic, its gradient is linear and given by

$$\nabla g(\phi) = D^T D \phi + D^T e_n.$$

Replacing  $u^{(k)} = \nabla g(y^{(k)})$  and using the fact that  $x^* = Ax^*$  and  $y^* = Cx^*$ , one can write the error system as

$$x^{(k+1)} - x^* = (A + BQC)(x^{(k)} - x^*)$$

where  $Q = D^T D = \nabla^2 g(\phi)$ . Thus, one can study the convergence properties of all three algorithms by studying the eigenvalues of  $T := A + BQC$ , following standard linear systems theory.

The analysis of the spectra of  $T$  can be simplified due to the following relationship

$$T = \begin{bmatrix} U & 0_n \\ 0_n & U \end{bmatrix} (A + B\Lambda C) \begin{bmatrix} U & 0_n \\ 0_n & U \end{bmatrix}^T$$

where we used the eigenvalue decomposition  $Q = U\Lambda U^T$ , where  $U$  is orthogonal and  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ . Thus, by similarity, the eigenvalues of  $T$  are the eigenvalues of all matrices

$$T_i = A_1 + B_1 \lambda_i C_1, \quad i = 1, \dots, n$$

where  $A_1$ ,  $B_1$  and  $C_1$  correspond to the matrices  $A$ ,  $B$  and  $C$  in (6a), (6b) and (6c) with  $n = 1$ , respectively. The implication being that to prove stability and to determine the convergence rate, it suffices to determine the eigenvalues of the  $T_i$ .

GRADIENT: For the case of the gradient method,  $T_i = 1 - \beta \lambda_i$ . Because  $Q$  is a circulant matrix, its eigenvalues are given by  $\lambda_i = 2 - 2 \cos(\frac{2\pi(i-1)}{n})$  and clearly  $\lambda_1 = 0$  which makes  $T_1 = 1$ . Solving the inequality  $|1 - \alpha(1 - \cos(\frac{2\pi(i-1)}{n}))| < 1$  (since  $\beta = \alpha/2$ ) gives that  $\alpha \in (0, 1)$ .

The convergence rate is given by the eigenvalue of  $T_i$  with the largest magnitude, i.e.,

$$\begin{aligned}\lambda_G &= \max \left| 1 - \frac{\alpha}{2} \lambda_i \right| \\ &= \max \left\{ \left| 1 - \frac{\alpha}{2} \min \lambda_i \right|, \left| 1 - \frac{\alpha}{2} \max \lambda_i \right| \right\}\end{aligned}$$

HEAVY-BALL: For this method, the matrices  $T_i$  are the following

$$T_i = \begin{bmatrix} 1 + \gamma - \beta \lambda_i & -\gamma \\ 1 & 0 \end{bmatrix}$$

and the algorithm converges provided that all eigenvalues except the one associated with  $\lambda_1 = 0$  are smaller than one in magnitude, or equivalently, that the following convergence rate can be made smaller than one

$$\begin{aligned}\lambda_H &= \max\{|\nu_1|, |\nu_2|\} \\ &= \begin{cases} \frac{1}{2}|1 + \gamma - \beta \lambda_i| + \frac{1}{2}\sqrt{\Delta}, & \text{if } \Delta \geq 0 \\ \sqrt{\gamma}, & \text{otherwise.} \end{cases}\end{aligned}$$

where  $\Delta = (1 + \gamma - \beta \lambda_i)^2 - 4\gamma$ . The above comes directly from the characteristic polynomial of  $T_i$

$$\nu^2 - (1 + \gamma - \beta \lambda_i)\nu + \gamma = 0.$$

A suboptimal choice is to select  $\beta = \frac{1}{\max \lambda_i}$  and solve for  $\gamma$ .

NESTEROV: For this method, the matrices  $T_i$  are the following

$$T_i = \begin{bmatrix} (1 + \gamma)(1 - \beta \lambda_i) & -\gamma(1 - \beta \lambda_i) \\ 1 & 0 \end{bmatrix}$$

and similarly to the HEAVY-BALL method,

$$\begin{aligned}\lambda_N &= \max\{|\nu_1|, |\nu_2|\} \\ &= \begin{cases} \frac{1}{2}|(1 + \gamma)(1 - \beta \lambda_i)| + \frac{1}{2}\sqrt{\Delta}, & \text{if } \Delta \geq 0 \\ \sqrt{\gamma(1 - \beta \lambda_i)}, & \text{otherwise.} \end{cases}\end{aligned}$$

where  $\Delta = (1 + \gamma)^2(1 - \beta \lambda_i)^2 - 4\gamma(1 - \beta \lambda_i)$ . The above comes directly from the characteristic polynomial of  $T_i$

$$\nu^2 - (1 + \gamma)(1 - \beta \lambda_i)\nu + \gamma(1 - \beta \lambda_i) = 0.$$

The previous theorem used a dynamical systems approach to establish convergence results similar to those in [17]. However, this approach enabled us to explicitly compute the rate of convergence for different optimization algorithms depending solely on the eigenvalues of  $Q$  (that are known in closed-form) and the chosen parameters  $\beta$  and  $\gamma$ .

#### IV. DESYNCHRONIZATION USING GAUSS-SEIDEL ITERATIONS

In this section, we present preliminary results regarding the solution to the problem (4) but using the approach of iterative algorithms for systems of linear equations. We describe the Gauss-Seidel algorithm and apply it to solving the linear equation  $\nabla g(\phi) = 0$ .

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:

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

which, by taking advantage of the triangular form of  $L + D$ , and the formulation of the desynchronization problem becomes

$$\begin{aligned}\phi_1^{(k+1)} &= \frac{1}{2} \left( 1 - \phi_2^{(k)} - \phi_n^{(k)} \right) \\ \phi_i^{(k+1)} &= \frac{1}{2} \left( -\phi_{i-1}^{(k+1)} - \phi_{i+1}^{(k)} \right), 2 \leq i \leq n-1 \\ \phi_n^{(k)} &= \frac{1}{2} \left( -1 - \phi_1^{(k+1)} - \phi_{n-1}^{(k+1)} \right)\end{aligned}\quad (7)$$

which requires communication with the immediate neighbors akin the original problem and exploits the inherent sequential behavior of the DESYNC algorithm to have nodes using the most updated values for the phases. The next theorem asserts the exponential rate of convergence for (7).

*Theorem 5 (Convergence Rate of Gauss-Seidel):* The iterative method (7) asymptotically converges to a desynchronization state with exponential convergence rate  $\lambda_{GS}$ , i.e.,

$$\phi^{(k+1)} - \phi^* \leq \lambda_{GS}^{k+1}(\phi^{(0)} - \phi^*) \quad (8)$$

where

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

and

$$T_{GS} = \sum_{j=0}^{n-1} \left( \frac{1}{2} \right)^{j+1} E^j E^\top, \quad (9)$$

$$E = \begin{bmatrix} 0_{n-1}^\top & 0 \\ I_{n-1} & 0_{n-1} \end{bmatrix} + e_n e_1^\top \quad (10)$$

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

*Proof:* The inequality in (8) comes directly from seeing the Gauss-Seidel algorithm as a linear time-invariant system with transition matrix  $T_{GS} := -(D + L)^{-1}U$ .

The first step in the proof is noting that matrix  $T_{GS}$  for the DESYNC problem has the form:

$$\begin{aligned}T_{GS} &= (2I_n - E)^{-1} E^\top \\ &= \frac{1}{2} (I_n - \frac{1}{2} E)^{-1} E^\top\end{aligned}\quad (11)$$

with the strictly lower triangular matrix  $E$  defined in (10). 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, (11) simplifies to (9).

The second step is to show stability by proving  $\rho(T_{GS}) \leq 1$ . Matrix  $T_{GS}$  is row stochastic since its elements are

trivially nonnegative and

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

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

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

where

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

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, it 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 follows. ■

## V. SIMULATION RESULTS

The main objective of this section is to compare the use of the Nesterov optimization against the Gauss-Seidel iteration, GS using the results in Theorems 4 and 5. In all simulations, the initial state  $\phi^{(0)} = \mathbf{1}_n/n$  meaning that all nodes start at the same phase.

Figure 1 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 [17]), GAUSS-SEIDEL, NESTEROV and HEAVY-BALL for the fixed parameters  $\beta = \frac{1}{4}, \gamma = \frac{1}{2}$ . The main observation is that the optimization methods improve performance and so does the use of an iterative algorithm for solving linear equations. As expected all algorithms have a convergence rate that approaches the unity as the number of nodes in the network grows.

An important remark is that the version of the Nesterov method proposed in [17] has time-varying parameters (in particular  $\gamma = \frac{k-1}{k+2}$ ) that might contribute to increase the

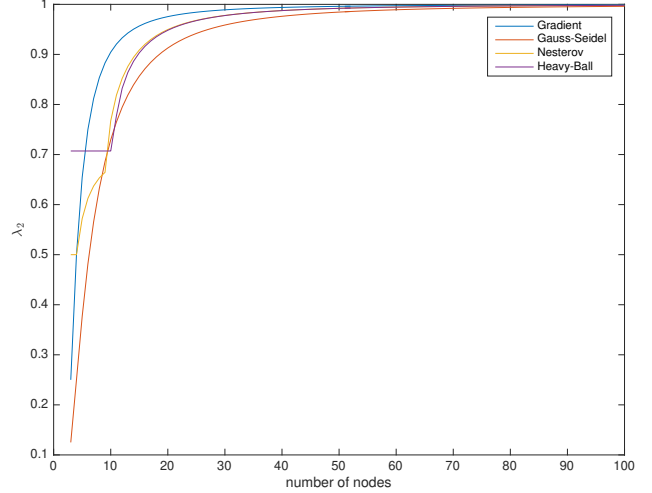


Fig. 1. The convergence rate  $\lambda_2$  for the different algorithms depending on the number of nodes  $n$ .

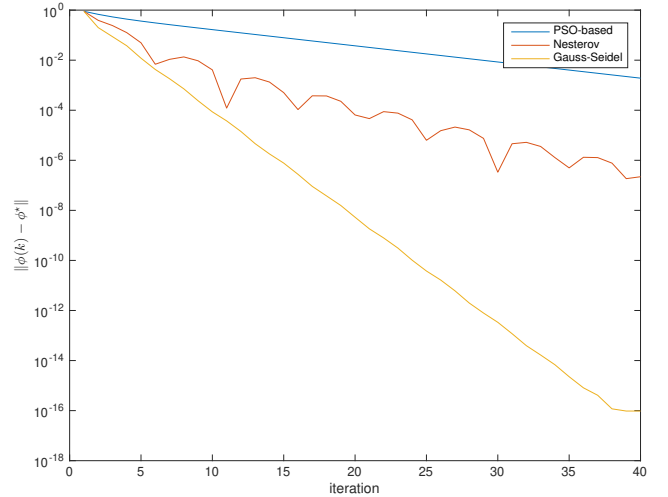


Fig. 2. Logarithmic evolution of the error norm for the PCO-based, Nesterov and Gauss-Seidel algorithms.

speed of convergence. In order to compare the method proposed in [17], a simulation of a  $n = 5$  node network was conducted and the logarithm of the error norm is presented in Figure 2.

Figure 2 shows that the Gauss-Seidel iteration achieves a faster convergence at a fixed rate in comparison with the algorithm in [17]. 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 Figure 3.

The main observation from the evolution of the error in Figures 2 and 3 is that as  $n$  increases, the behavior of the error norm changes. For small networks, the Gauss-Seidel method outperforms the Nesterov algorithm. When

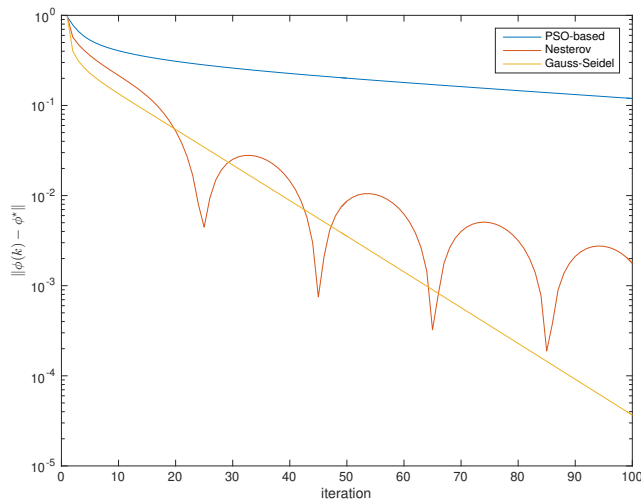


Fig. 3. Logarithmic evolution of the error norm for the PCO-based, Nesterov and Gauss-Seidel algorithms for the 20 node network.

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  $n$  as observed in simulations for larger networks. This trend suggest that we are still missing a complete description of the convergence properties of the optimization algorithms and iterative solutions for linear equations.

## 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.

The theoretical results and simulations presented in this paper suggest that further research is needed to optimize the convergence rate of all the methods and to possibly categorize under which assumptions it is beneficial to use one in detriment of another. Moreover, the contribution of using iterative solvers for linear equations suggests that one should consider other algorithms that may bring additional gains in performance. 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 state-of-the-art. 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. Degeys 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] 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.
- [4] 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.
- [5] 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.
- [6] 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 802.15.4-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–225, April 2012.
- [7] 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.
- [8] A. Motskin, T. Roughgarden, P. Skraba, and L. Guibas, "Lightweight coloring and desynchronization for networks," in *IEEE INFOCOM 2009*, April 2009, pp. 2383–2391.
- [9] C. M. Lien, S. H. Chang, C. S. Chang, and D. S. Lee, "Anchored desynchronization," in *2012 Proceedings IEEE INFOCOM*, March 2012, pp. 2966–2970.
- [10] 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.
- [11] 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.
- [12] 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.
- [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*, vol. 182, no. 1, pp. 77–92, 2012, nature-Inspired Collective Intelligence in Theory and Practice.
- [14] 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.
- [15] 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.
- [16] 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.
- [17] 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.
- [18] D. Silvestre, J. Hespanha, and C. Silvestre, "A pagerank algorithm based on asynchronous gauss-seidel iterations," in *Annual American Control Conference (ACC)*, June 2018, pp. 484–489.
- [19] 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.
- [20] 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*, pp. 1–1, 2018.
- [21] 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.