EDOARDO D'ELIA

ANALYSIS OF CLOCKS SYNCHRONIZATION ALGORITHMS IN WIRELESS SENSOR NETWORKS

TESI DI LAUREA MAGISTRALE

Relatore: Chiar.mo Prof. Sandro Zampieri
Co-relatore: Chiar.mo Dr. Ruggero Carli

Università degli Studi di Padova
Facoltà di Ingegneria
Dipartimento di Ignegneria dell'Informazione
Marzo 2011

L'uomo vede meglio le cose quando ne indaga accuratamente l'origine, che è la parte più importante di esse. Quando ne conosce la spiegazione, esse appaiono più belle.

— Gilbert K. Chesterton

Dedicato a tutti i miei fantastici Amici

ABSTRACT

In this work we analyze a novel distributed clock synchronization protocol. The algorithm compensates for both initial offsets and differences in internal clock speeds and is based on a Proportional-Integral (PI) controller that treats the different clock speeds as unknown constant disturbances and the different clock offsets as different initial conditions for the system dynamics.

The clocks are assumed to exchange information through either a symmetric-gossip or an asymmetric-gossip communication protocols.

Convergence of the algorithm is proved and analyzed with respect to the controller parameter. An intensive simulation study is provided to compare our algorithm with other distributed strategies presented in literature.

SOMMARIO

In questo lavoro analizziamo un nuovo protocollo distribuito per la sincronizzazione di orologi. L'algoritmo compensa sia gli offset iniziali che le velocità interne degli orologi e si basa su un controllore Proporzionale-Integrativo (PI) che tratta le diverse velocità come incogniti disturbi costanti e i differenti offset come distinte condizioni iniziali.

Assumeremo che gli orologi possano scambiarsi informazioni attraverso protocolli di comunicazione di tipo gossip simmetrico o asimmetrico.

Verrà fornita una prova della convergenza dell'algoritmo ed analizzata con riferimento ad i parametri di controllo. Inoltre, sarà presentato un intensivo studio simulativo per confrontare il nostro algoritmo con altre strategie distribuite presenti in letteratura.

Philosophy is the basis for Logic,
which is exactly 33.3% of Math.
(And because you are just dying to know:
the other 66.6% is evenly divided
between Set Theory and Number Theory,
the remaining per mil is magic).

— Anonymous Coward from **science.slashdot.org**

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my assistant supervisor Dr. Ruggero Carli. In these months Ruggero introduced me into the wonderful world of research and he taught me the relevance to proceed step by step and always in a rigorous fashion. I am also thankful to him for his constant willingness and for his help with my raw English.

Second I would express my gratitude to my supervisor Prof. Sandro Zampieri who gave me the opportunity to face up to an interesting problem.

I also wish express my gratitude to my fellow students who made these years at the University much more easier.

Of course, I cannot forget my friends for all the great moments spent together: thanks a million guys!

Worthy of mention are my aunt Semira and my grandmother "Mimina" for having always encouraged me.

Last but not least my thought goes to my parents and my younger brother Massimo for belive in me with no ifs, ands, or buts.

Padova, marzo 2011 E. D.

CONTENTS

LIST OF FIGURES

ACRONYMS

WSN Wireless Sensor Network

Consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants.

ATS Average Time-Sync

A protocol for synchronizing a wireless sensor network using an averaging consensus approach. This algorithm is based on a class of popular distributed algorithms known as consensus, agreement, gossip or rendezvous whose main idea is averaging local information.

DTSP Distributed Time-Sync Protocol

A fully distributed and asynchronous algorithm that achieve time synchronization in a multi-hop wireless sensor network and exploit some global constraints to improve the performance.

FTSP Flooding Time Synchronization Protocol

A time synchronization protocol that uses low communication bandwidth and it is robust against node and link failures. The algorithm achieves its robustness by utilizing periodic flooding of synchronization messages, and implicit dynamic topology update.

A decentralized synchronicity algorithm based on a mathematical model that describes how fireflies and neurons spontaneously synchronize.

TDMA Time Division Multiple Access

Is a channel access method for shared medium networks. It allows several users to share the same frequency channel by dividing the signal into different time slots. The users transmit in rapid succession, one after the other, each using his own time slot.

TPSN Timing-sync synchronization Protocol for Sensor Networks

A synchronization algorithm that works in two steps. In the first step, a hierarchical structure is established in the network and then a pair wise synchronization is performed along the edges of this structure to establish a global timescale throughout the network. Eventually all nodes in the network synchronize their clocks to a reference node.

NOTATION

$\mathbb{N}$	the natural number set
$\mathbb{R}$	the real number field
$\mathbb{C}$	the complex number field
$\mathbb{C}^{m\times n}$	the set of $m \times n$ matrices whose elements are in $\mathbb{C}$
$\mathbb{1}$	the (coloumn) vector whose entries are all equals to 1
I_n	identity matrix of order n
$\mathrm{diag}(x)$	diagonal matrix having the components of x as diagonal elements or, equivalently, $\mathrm{diag}(x_1, \ldots, x_n)$
$\mathrm{dg}(A)$	diagonal matrix with the same diagonal elements of A
$\mathrm{col}_i(A)$	the i-th column of A
$\mathrm{row}_i(A)$	the i-th row of A
A^T	transpose of A
$\bar{A}$	conjugate of A
A^*	conjugate transpose of A
$A \otimes B$	direct (Kronecker) product of A and B
$A \odot B$	Hadamard (element by element) product of A and B
$A^\dagger$	Moore-Penrose generalized inverse of A
$r(A)$	rank of A
$\|A\|$	Euclidean norm of A
$\|A\|_\infty$	supremum norm of A
$\|A\|_F$	Frobenius norm of A

If A is square,

$\det(A)$	determinant of A
$\mathrm{tr}(A)$	trace of A
$\Lambda(A)$	the set of eigenvalues of A
A^{-1}	inverse of A
$\rho(A)$	spectral radius of A

1 INTRODUCTION

The recent technological advances in Wireless Sensor Networks (WSNs) and the decreasing in cost and size of electronic devices have promoted the appearance of large inexpensive interconnected systems, each with computational and sensing capabilities. These complex networks of agents are used in a large number of applications covering a wide range of fields, such as, surveillance, targeting systems, controls, power scheduling and Time Division Multiple Access (TDMA) communications, monitoring areas, intrusion detection, vehicle tracking and mapping. One key problem in many of these applications is clock synchronization. Indeed, very often, it is essential that the agents act in a coordinated and synchronized fashion requiring global clock synchronization, that is, all the agents of the network need to refer to a common notion of time.

A wide variety of clock synchronization protocols have been proposed recently in the literature. Depending upon the architectures adopted, these protocols can be divided into three categories: tree-structure-based, cluster-structure-based, and fully-distributed.

Tree-structure-based protocols such as the Timing-sync synchronization Protocol for Sensor Networks (TPSN) by Ganeriwal *et al.* [2003] and the Flooding Time Synchronization Protocol (FTSP) by Maróti *et al.* [2004] consist in electing a reference node and creating a spanning tree rooted at this reference node, where each children synchronizes itself with respect to its parent. In cluster-structure-based protocols such as in Elson *et al.* [2002], the network is divided into distinct clusters, each with an elected cluster-head. All nodes within the same cluster synchronize themselves with the corresponding cluster-head, and each cluster-head synchronizes itself with an another cluster-head. Although these two strategies have been experimentally tested showing remarkable performance, they suffer from robustness and scalability issues. For instance, if a node dies or a new node is added, then it is necessary to rebuild the tree or the clusters, at the price of additional implementation overhead and possibly long periods in which the network or part of it is poorly synchronized.

Fully distributed algorithms for clocks synchronization have appeared in Werner-Allen *et al.* [2005], Simeone and Spagno-

lini [2007], Solis *et al.* [2006] and Schenato and Fiorentin [2009]. Werner-Allen *et al.* [2005] introduced a protocol inspired by the fireflies integrate-and-fire synchronization mechanism, able to compensate for different clock offsets but not for different clock skews. On the opposite, the algorithm proposed by Simeone and Spagnolini [2007] adopting a P-controller, compensates for the clock skews but not for the offsets. Distributed protocols that can compensate for both clock skews and offsets have been proposed by Solis *et al.* [2006] with the Distributed Time-Sync Protocol (DTSP) and by Schenato and Fiorentin [2009] with the Average Time-Sync (ATS). The first one is based on the cascade of two distributed least-squared algorithms, while the second one is based on the cascade of two first order consensus algorithms. Of note is the fact that both these strategies are highly non-linear and do not lead to a simple characterization of the effects of noise on the steady-state performance.

Differently, Carli *et al.* [2008] and Carli and Zampieri [2010] proposed a synchronization algorithm that can be formally analyzed not only in the noiseless scenario in terms of rate of convergence, but also in a noisy setting in terms of the steady-state synchronization error. This algorithm compensates for both initial offsets and differences in internal clock speeds and is based on a Proportional-Integral (PI) controller that treats the different clock speeds as unknown constant disturbances and the different clock offsets as different initial conditions for the system dynamics. Both convergence guarantees as well optimal design using standard optimization tools when the underlying communication graph is known, can be provided as shown by Carli *et al.* [2011]. It is important to remark that the time-synchronization algorithm proposed by Carli and Zampieri [2010] requires each node to perform all the operations related to the k-th iteration of the algorithm, including transmitting messages, receiving messages and updating estimates, within a short time window. This pseudo-synchronous implementation might be very sensitive to packet losses, node and link failure.

In this dissertation we developed and analyzed a far more practical version of the PI synchronization algorithm, assuming that clocks can exchange information through either a symmetric-gossip or an asymmetric-gossip communication protocols. This applies very well to real sensor networks, and drastically reduces the network requirements in terms of reliability, bandwidth, and synchronization. Theoretical results are provided when the underlying communication topology is given by the complete graph, while more general families of graphs are considered by means of simulations.

The presentation of the work is structured as follows:

IN THE SECOND CHAPTER we propose a mathematical model of each local clock, then we introduce the reader to the synchronization problem we want to solve, namely the fact that the time is an unknown variable, which has to be estimated.

IN THE THIRD CHAPTER we propose a Proportional-Integral (PI) consensus controller based on the *gossip communication* model. We first describe a symmetric implementation of the algorithm, where at each iteration two neighboring units establish a communication link and exchange their time stamp to each others. Then we report the asymmetric version of the protocol, where the communication link is directional, reducing the network requirements in terms of reliability and bandwidth.

IN THE FOURTH CHAPTER we provide a numerical comparison of our protocol with ATS, another asynchronous and fully distributed algorithm, focusing our attention to performance and accuracy.

IN THE LAST CHAPTER we briefly summarize the work done, the results achieved, and the possible directions of investigation.

2 THE SYNCHRONIZATION PROBLEM

We start by proposing a model of each local clock which is simple enough but which captures the main difficulty of the problem we want to solve, namely the fact that the time is an unknown variable, which has to be estimated.

2.1 MATHEMATICAL MODELING OF A CLOCK

Assume that each unit has a clock, which is an oscillator capable to produce an event at time $t(k)$, $k = 0, 1, 2, \ldots$ The clock has to use these clicks in order to estimate the time. The following cumulative function well describes the time evolution of the tick counter that can be implemented in the clock

$$s(t) = \int_{-\infty}^{t} f(\sigma) d\sigma,$$

where

$$f(t) = \sum_{k=0}^{\infty} \delta(t - t(k)). \tag{2.1}$$

In this way the counter output is a step shaped function. In case the clock period is small, in order to simplify the analysis, it is convenient to approximate step shaped function $s(t)$ with a continuous one by approximating the delta shaped function $f(t)$ with a regularized version. One way is to take

$$f(t) := \frac{1}{t(k+1) - t(k)} \quad \text{for all} \quad t \in [t(k), t(k+1)[\tag{2.2}$$

The graphs of the stepwise and of the regularized version of $s(t)$ is shown in Figure 1.

Notice that $f(t)$ can be interpreted as the oscillator frequency at time t. Typically an estimate $\hat{f}$ of $f(t)$ is available and it is known that $f(t) \in [f_{min}\ f_{max}]$. From the counter one can build a time estimate $\hat{t}(t)$ by letting

$$\hat{t}(t) = \hat{t}(t_0) + \hat{\Delta}(t)[s(t) - s(t_0)] = \hat{t}(t_0) + \hat{\Delta}(t) \int_{t_0}^{t} f(\sigma)\, d\sigma \tag{2.3}$$

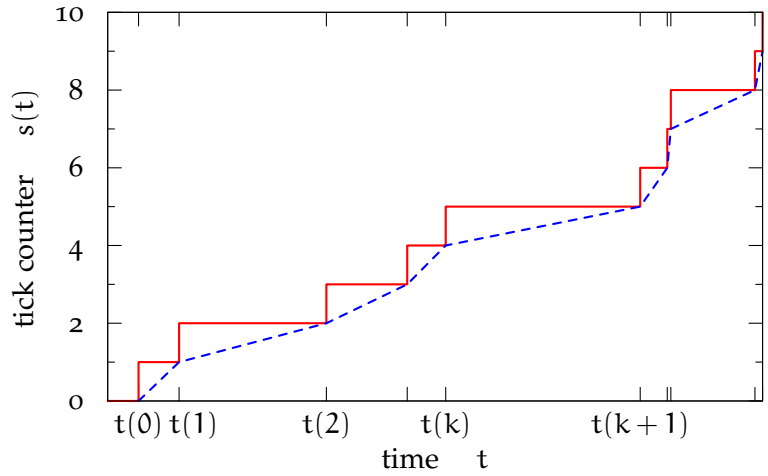

Figure 1: The graphs of of $s(t)$ resulting from the impulse shaped version of $f(t)$ given in (2.1) (continuous line) and from the regularized version of $f(t)$ given in (2.2) (dashed line).

where $\hat{\Delta}(t)$ is an estimate of the oscillation period $1/f(t)$. It is reasonable to initialize $\hat{\Delta}(t)$ to $1/\hat{f}$.

Both $\hat{t}(t)$ and $\hat{\Delta}(t)$ can be modified when the unit obtains information allowing it to improve its time and oscillator frequency estimates. Assume that these corrections are applied at time instants $T_{up}(h)$, where $h = 0, 1, \ldots$, called updating time instants. In this case we have [1]

$$\begin{cases} \hat{t}(T_{up}^{+}(h)) = \hat{t}(T_{up}^{-}(h)) + u'(h) \\ \hat{\Delta}(T_{up}^{+}(h)) = \hat{\Delta}(T_{up}^{-}(h)) + u''(h) \end{cases}$$

where u' and u'' denote the control inputs applied to $\hat{t}$ and $\hat{\Delta}$, respectively.

For $t \in [T_{up}^{+}(h)\ T_{up}^{-}(h+1)]$ the estimates evolve according to

$$\begin{cases} \hat{t}(t) = \hat{t}(T_{up}^{+}(h)) + \hat{\Delta}(T_{up}^{+}(h))\left(s(t) - s(T_{up}(h))\right) \\ \hat{\Delta}(t) = \hat{\Delta}(T_{up}^{+}(h)) \end{cases} \tag{2.4}$$

or according to

$$\begin{cases} \hat{t}(t) = \hat{t}(T_{up}^{+}(h)) + \hat{\Delta}(T_{up}^{-}(h))\left(s(t) - s(T_{up}(h))\right) \\ \hat{\Delta}(t) = \hat{\Delta}(T_{up}^{+}(h)) \end{cases} \tag{2.5}$$

Intuitively the updating rule (2.4) should perform better than the updating rule (2.5). However, for the sake of simplicity — as we will see later — in this dissertation we assume that the units adopt the updating rule (2.5).

1 Given the time t, with the symbols t^{+} and t^{-} we mean, respectively, the time instant just after t and time instant just before t.

2.2 CLOCK SYNCHRONIZATION

Assume now that we have a network composed by N clocks. For $i \in \{1, \dots, N\}$, let $f_i(t)$ be the evolution of the oscillator frequency of the clock i. Moreover, for $i \in \{1, \dots, N\}$, let $x_i(t) = [x_i'(t)\ x_i''(t)]^* = [\hat{t}_i(t)\ \hat{\Delta}_i(t)]^*$ denote the local state of the clock i. The objective is to synchronize the variables $x_i'(t)$, $i \in \{1, \dots, N\}$, namely, to find a law which allows the clocks to obtain the same time estimate.

Assume that the clocks can exchange their local state according to a graph $\mathcal{G} = (V, \mathcal{E})$, where $V = \{1, \dots, N\}$ and where $(i, j) \in \mathcal{E}$ whenever the clock i can send its state x_i to the clock j.

Specifically, each clock i, $i \in \{1, \dots, N\}$, transmits its state $x_i(t)$ at some time instants $T_{tx,i}(h)$, $h = 0, 1, \dots$, and can use any information it receives from the neighboring nodes to perform a control at the time instants $T_{up,i}(h)$, $h = 0, 1, \dots$

More precisely

$$x_i(T_{up,i}^+(h)) = x_i(T_{up,i}^-(h)) + u_i(h), \tag{2.6}$$

where $u_i(h) = [u_i'(h)\ u_i''(h)]^*$ is the control action applied at time $T_{up,i}(h)$. Moreover for $t \in [T_{up,i}^+(h)\ T_{up,i}^-(h+1)]$ we assume that the state x_i is updated according to (2.5).

For simplicity in this dissertation we assume the following properties.

Assumption 2.2.1. The oscillator frequencies f_i are constant but unknown, i.e., for $i \in \{1, \dots, N\}$, $f_i(t) = \bar{f}_i$ for all $t > 0$.

Assumption 2.2.2. The transmission delays are negligible, namely, if clock i perform its h-th transmission to clock j, then clock j receives the information $x_i(T_{tx,i}(h))$ exactly at time $T_{tx,i}(h)$ [2].

From Assumption 2.2.1 and from (2.3), it follows that (2.5), for the i-th clock, can be equivalently rewritten as

$$\begin{cases} x_i'(t) = x_i'(T_{up}^+(h)) + x_i''(T_{up}^-(h))\, \bar{f}_i\, (t - T_{up}(h)) \\ x_i''(t) = x_i''(T_{up}^+(h)) \end{cases} \tag{2.7}$$

The objective is to find a control strategy yielding the clock synchronization, namely such that there exist constants $a \in \mathbb{R}_{>0}$ and $b \in \mathbb{R}$ such that synchronization errors

$$e_i(t) := x_i'(t) - (at + b), \quad i = \{1, \dots, N\} \tag{2.8}$$

converge to zero or remain small.

2 In general, the information $x_i(T_{tx,i}(h))$ is received by clock $j \in \mathcal{N}_i$ at a delayed time $T_{rx,i,j}(h) \geqslant T_{tx,i}(h)$ where $T_{rx,i,j}(h) = T_{tx,i}(h) + \gamma_{i,j}(h)$ being $\gamma_{i,j}(h)$ a nonnegative real number representing the deliver delay between i and j.

3 THE PI PROTOCOL

In this chapter, following the work of Carli *et al.* [2008], we propose and analyze a Proportional-Integral (PI) consensus controller based on more practical communication schemes. Indeed, as emphasize by Carli and Zampieri [2010], a relevant limitation in their protocol is that the nodes data transmission and algorithm updating steps cannot occur exactly at the same time. To avoid this problem the authors introduced a pseudo-synchronous implementation which require each node to perform all the computing burden in a short time window. However, this strategy might be very sensitive to packet drop, node and link failure. To reduce the communication requirements we consider a different protocol, more precisely the *gossip communication* model, i. e. a model in which only a pair of clocks can exchange their information at any time. We first describe a symmetric implementation of the algorithm, where at each iteration two neighboring units establish a communication link and exchange their time stamp to each others. Then we report the asymmetric version of the protocol, where the communication link is directional, reducing the network requirements in terms of reliability and bandwidth.

3.1 SYMMETRIC GOSSIP

In this section we assume the clocks exchange information with each other through a *symmetric gossip* communication protocol over an undirected connected graph $\mathcal{G} = (V, \mathcal{E})$.

Informally, we assume that, for $i \in \{1, \dots, N\}$, clock i wakes up at the sample times of a Poisson process of a certain intensity $\lambda > 0$ and establishes a bidirectional link with one of its neighbors randomly selected.

Specifically let $\{t_{i,k}, k \in \mathbb{N}\}$ denote the time instants in which clock i wakes up. Then at time $t_{i,k}$,

- clock i randomly chooses with probability $1/d_i$ a clock j within the set of its neighbors $\mathcal{N}_i = \{j \in V : (i,j) \in \mathcal{E}, j \neq i\}$, where d_i denote the cardinality of $\mathcal{N}_i$;

- clock i sends to clock j only the information related to the first component of its state, i. e. $x'_i(t_{i,k})$;
- clock j *instantaneously* responds back to node i sending the information related to the first component of its state, i. e. $x'_j(t_{i,k})$.

Notice that for some $h, h' \in \mathbb{N}$ we have that

$$t_{i,k} = T_{tx,i}(h) = T_{tx,j}(h') = T_{up,i}(h) = T_{tx,j}(h').$$

Based on the information received clock i and clock j applies to their current states x_i and x_j the following correction

$$u_i = \begin{bmatrix} u'_i \\ u''_i \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 \\ \alpha \end{bmatrix} \left(x'_j(t_{i,k}) - x'_i(t_{i,k}) \right),$$
$$u_j = \begin{bmatrix} u'_j \\ u''_j \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 \\ \alpha \end{bmatrix} \left(x'_i(t_{i,k}) - x'_j(t_{i,k}) \right),$$

where α is a control parameter such that $\alpha > 0$. From (2.6), it follows

$$x'_i(t^+_{i,k}) = \frac{1}{2} \left(x'_i(t_{i,k}) + x'_j(t_{i,k}) \right), \tag{3.1a}$$
$$x'_j(t^+_{i,k}) = \frac{1}{2} \left(x'_i(t_{i,k}) + x'_j(t_{i,k}) \right) \tag{3.1b}$$

and

$$x''_i(t^+_{i,k}) = x''_i(t_{i,k}) + \frac{\alpha}{2} \left(x'_j(t_{i,k}) - x'_i(t_{i,k}) \right), \tag{3.2a}$$
$$x''_j(t^+_{i,k}) = x''_j(t_{i,k}) + \frac{\alpha}{2} \left(x'_i(t_{i,k}) - x'_j(t_{i,k}) \right). \tag{3.2b}$$

Remark 3.1.1. The control above introduced can be seen as a PI controller where $u'_i = x'_j(t_k) - x'_i(t_k)$ and $u''_i = \alpha(x'_j(t_k) - x'_i(t_k))$ represent, respectively, the proportional and the integral action on node i.

Now let $\{ t_k, k \in \mathbb{N} \}$ be the set of all the transmitting/updating time instants of the clocks' network, i. e.

$$\{ t_k, k \in \mathbb{N} \} = \bigcup_{i=1}^{N} \{ t_{i,k}, k \in \mathbb{N} \}.$$

We assume the N Poisson processes generating the transmission in which the clocks wake up are independent from one another, the updating time instants $\{ t_k, k \in \mathbb{N} \}$ can be seen as the sample times of a Poisson process of intensity $N\lambda$.

Next we provide a convenient vector-form description of the evolution of the clocks' network. To do so, we need some auxiliary definitions. First, let us introduce the following vectors

$$x' := [x'_1 \ \dots \ x'_N]^* \in \mathbb{R}^N,$$
$$x'' := [x''_1 \ \dots \ x''_N]^* \in \mathbb{R}^N,$$
$$x = [x'_1 \ \dots \ x'_N \ x''_1 \ \dots \ x''_N]^* \in \mathbb{R}^{2N}.$$

Second, for $i, j \in \{1, \dots, N\}$ let the matrix $E_{i\leftrightarrow j} \in \mathbb{R}^{N\times N}$ be defined as

$$E_{i\leftrightarrow j} := (e_i - e_j)(e_i - e_j)^*,$$

and let the matrix $D \in \mathbb{R}^{N\times N}$ be defined as

$$D = \mathrm{diag}\{\bar{f}_1, \dots, \bar{f}_N\}.$$

Finally let $\delta t_k := t_{k+1} - t_k$. Then, combining (2.7) with (3.1) and (3.2), we can write

$$x(t^-_{k+1}) = \begin{bmatrix} I - \frac{1}{2}E_{i\leftrightarrow j} & \delta t_k D \\ -\frac{\alpha}{2}E_{i\leftrightarrow j} & I \end{bmatrix} x(t^-_k).$$

To simplify the notation we define $x(k) := x(t^-_k)$, hence the above system becomes

$$x(k+1) = \begin{bmatrix} I - \frac{1}{2}E(k) & \delta t_k D \\ -\frac{\alpha}{2}E(k) & I \end{bmatrix} x(k), \tag{3.3}$$

where $E(k) = E_{i\leftrightarrow j}$ if, during the k-th iteration communication between node i and node j take place.

We extensively simulated algorithm in (3.3), for several communication graphs $\mathcal{G}$ and for different values of D, λ and α. Next we summarize some of the numerical evidences we extrapolated from the simulations we run:

1. Typically, for $\alpha > \lambda$, the algorithm does not reach the synchronization, independently from the values of $\bar{f}_i$.

2. For $\alpha \leqslant \lambda$ the synchronization is achieved depending on how spread the values $\{\bar{f}_i, i = 1, \dots, N\}$ are. More precisely suppose that, for $i \in \{1, \dots, N\}$, $\bar{f}_i \in [1 - \epsilon \ 1 + \epsilon]$ where $0 \leqslant \epsilon < 1$. Then the smaller the value of α is, the greater the value of ϵ is while still reaching the synchronization. In particular, if $\alpha \approx \lambda$ then the synchronization is attained only if $\epsilon << 1$, while if $\alpha \approx 0$ then ϵ can be also very close to 1.

3. When the synchronization is achieved, then

$$\lim_{k\to\infty} x''(k) = \beta\mathbb{1},$$

where β is in general close to $\frac{1}{N}\sum_{i=1}^{N} \bar{f}_i$. This value β represents the oscillator frequency of the "virtual clock" to which all the clocks synchronize.

Providing a theoretical analysis of the above evidences is quite challenging in general. In the next section we restrict to complete and circulant graphs. In these cases we will be able to provide some theoretical insights on the convergence properties of algorithm (3.3).

3.1.1 Mean-Square Analysis

To our aims it is convenient to consider the synchronization error

$$y(k) = \Omega x'(k) = \left(I - \frac{1}{N}\mathbb{1}\mathbb{1}^*\right) x'(k)$$

and the new variable

$$z(k) = \Omega D x''(k) = \left(I - \frac{1}{N}\mathbb{1}\mathbb{1}^*\right) D x''(k).$$

Since for any $i, j \in \{1, \dots, N\}$, $i \neq j$, $E_{i\leftrightarrow j}\Omega = E_{i\leftrightarrow j}\Omega = E_{i\leftrightarrow j}$, system (3.3) can be rewritten as

$$\begin{bmatrix} y(k+1) \\ z(k+1) \end{bmatrix} = \begin{bmatrix} I - \frac{1}{2}E(k) & \delta t_k \\ -\frac{\alpha}{2}\Omega D E(k) & I \end{bmatrix} \begin{bmatrix} y(k) \\ z(k) \end{bmatrix} \tag{3.4}$$

Clearly x' reaches the asymptotic synchronization if and only if $\lim_{k\to\infty} y(k) = 0$.

Now observe that, according to the data transmission and communication model described above, system (3.4) evolves as a random process such that

- for any $k \in \mathbb{N}$, the matrix $E(k)$ is randomly selected within the set

$$\mathcal{S} = \left\{ E_{i\leftrightarrow j} : (i,j) \in \mathcal{E} \right\}$$

and

$$\mathbb{P}[E(k) = E_{i\leftrightarrow j}] = \frac{1}{N}\left(\frac{1}{d_i} + \frac{1}{d_j}\right);$$

- the selection of the matrix $E(k)$ is independent from the selection of $E(k')$, $k' \neq k$;

- $\{\delta t_k : k \in \mathbb{N}\}$ are the interarrival times of a Poisson process of intensity $N\lambda$.

It is useful to introduce the following $N \times N$ matrix as the matrix having $W^{(i,j)}$ as element in the i^{th} row and j^{th} column and in the j^{th} row and i^{th} column, namely

$$W_{ij} = W_{ji} := W^{(i,j)}. \tag{3.5}$$

Note that, since $i \ni \mathcal{N}_i$, we have that $W_{ii} = 0$.

The goal is to perform a mean-square analysis of (3.4). To do so, we introduce the matrix

$$\Sigma(k) := \mathbb{E}\left[\begin{bmatrix} y(k) \\ z(k) \end{bmatrix} \begin{bmatrix} y^*(k) & z^*(k) \end{bmatrix}\right] = \begin{bmatrix} \Sigma_{yy}(k) & \Sigma_{yz}(k) \\ \Sigma_{yz}^*(k) & \Sigma_{zz}(k) \end{bmatrix},$$

where

$$\begin{aligned} \Sigma_{yy}(k) &:= \mathbb{E}[y(k)y^*(k)], \\ \Sigma_{yz}(k) &:= \mathbb{E}[y(k)z^*(k)], \\ \Sigma_{zz}(k) &:= \mathbb{E}[z(k)z^*(k)]. \end{aligned}$$

The objective is to study the evolution of

$$\Sigma(k+1) = \mathbb{E}[\Gamma(k)\Sigma(k)\Gamma^*(k)] \tag{3.6}$$

where

$$\Gamma(k) := \begin{bmatrix} I - \frac{1}{2}E(k) & \delta t_k \\ -\frac{\alpha}{2}\Omega D E(k) & I \end{bmatrix}.$$

In what follows, we perform our analysis by assuming that D is a small perturbation of the matrix I. Accordingly, we will design the parameter α only for $D = I$. From the fact that the eigenvalues of the expectation operator in (3.6) depend continuously on the matrix D, it will follow that this choice of α yields the stability also for a small enough perturbation of D.

Assuming that $D = I$, from (3.6) we obtain the following recursive equations[1]

$$\begin{aligned}\Sigma^+_{yy} = \Sigma_{yy} &- \frac{1}{2}\left\{\mathbb{E}[E_{i\leftrightarrow j}]\Sigma_{yy} + \Sigma_{yy}\mathbb{E}[E_{i\leftrightarrow j}]\right\} \\ &+ \frac{1}{4}\mathbb{E}[E_{i\leftrightarrow j}\Sigma_{yy}E_{i\leftrightarrow j}] + \frac{1}{N\lambda}\left\{\Sigma_{yz} + \Sigma^*_{yz}\right\} \\ &- \frac{1}{2N\lambda}\left\{\mathbb{E}[E_{i\leftrightarrow j}]\Sigma_{yz} + \Sigma^*_{yz}\mathbb{E}[E_{i\leftrightarrow j}]\right\} + \frac{2}{N^2\lambda^2}\Sigma_{zz},\end{aligned} \tag{3.7a}$$

$$\begin{aligned}\Sigma^+_{yz} = &-\frac{\alpha}{2}\Sigma_{yy}\mathbb{E}[E_{i\leftrightarrow j}] + \frac{\alpha}{4}\mathbb{E}[E_{i\leftrightarrow j}\Sigma_{yy}E_{i\leftrightarrow j}] + \Sigma_{yz} \\ &- \frac{1}{2}\mathbb{E}[E_{i\leftrightarrow j}]\Sigma_{yz} - \frac{\alpha}{2N\lambda}\Sigma^*_{yz}\mathbb{E}[E_{i\leftrightarrow j}] + \frac{1}{N\lambda}\Sigma_{zz}, \text{ and}\end{aligned} \tag{3.7b}$$

$$\begin{aligned}\Sigma^+_{zz} = &\frac{\alpha^2}{4}\mathbb{E}[E_{i\leftrightarrow j}\Sigma_{yy}E_{i\leftrightarrow j}] \\ &- \frac{\alpha}{2}\left\{\mathbb{E}[E_{i\leftrightarrow j}]\Sigma_{yz} + \Sigma^*_{yz}\mathbb{E}[E_{i\leftrightarrow j}]\right\} + \Sigma_{zz},\end{aligned} \tag{3.7c}$$

where we used the fact that

$$\mathbb{E}[\delta t_k] = \frac{1}{N\lambda} \quad \text{and} \quad \mathbb{E}[\delta t_k{}^2] = \frac{2}{N^2\lambda^2}.$$

The covariance matrix Σ then updates according to a linear transformation

$$\Sigma(k+1) = F[\Sigma(k)] \tag{3.8}$$

defined by the recursive equations that we just computed, and whose initial conditions can be obtained once we state the following assumption on $x'(0)$ and $x''(0)$.

Assumption 3.1.1. The initial condition $x'(0)$ is a random vector such that $\mathbb{E}[x'(0)] = 0$, $\mathbb{E}[x'(0)x'(0)^*] = \sigma_x^2 I$. The vector $x''(0)$ is such that $x''(0) = \mathbb{1}$.

It then follows that

$$\Sigma(0) = \begin{bmatrix}\sigma_x^2\Omega & 0 \\ 0 & 0\end{bmatrix}. \tag{3.9}$$

The analysis of the previous recursive equations is a challenging problem when $\mathcal{G}$ is an arbitrary communication graph. In the next section we provide a detailed theoretical analysis for the complete graph. Later we consider the more general family of

1 For the sake of notational simplicity, time dependence has been omitted here; the notation Σ^+_{ij} stand for $\Sigma_{ij}(k+1)$.

circulant graphs, then we also yield an analysis for the random geometric graphs by meaning of numerical simulations. Before that let us introduce now a useful technical Lemma from Carli [2008].

Lemma 3.1.1. *Let $\mathcal{G} = (V, \mathcal{E})$ be a given graph without any self-loops and let W be the matrix defined as in* (3.5)*. Let Q any symmetric matrix. Then*

$$\mathbb{E}[E_{i\leftrightarrow j}] = \operatorname{diag}(W\mathbb{1}) - W \tag{3.10}$$

and

$$\begin{aligned}\mathbb{E}[E_{i\leftrightarrow j}QE_{i\leftrightarrow j}] = \operatorname{diag}(W\mathbb{1})\operatorname{dg}(Q) + 2W\odot Q \\ + \operatorname{diag}(W\operatorname{dg}(Q)\mathbb{1}) - \{\operatorname{dg}(Q)W + W\operatorname{dg}(Q)\} \\ - 2\operatorname{dg}(A(W\odot Q)),\end{aligned} \tag{3.11}$$

where the matrix A represent the adjacency matrix of the graph $\mathcal{G}$, namely

$$A_{ij} = A_{ji} = \begin{cases} 1, & \text{if} \quad (i,j) \in \mathcal{E} \\ 0, & \text{otherwise.}\end{cases}$$

PROOF We have that

$$\begin{aligned}\mathbb{E}[E_{i\leftrightarrow j}] &= \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_i - e_j)(e_i - e_j)^* \\ &= \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_i^* + e_je_j^*) \\ &\quad - \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_j^* + e_je_i^*) = \operatorname{diag}(W\mathbb{1}) - W.\end{aligned}$$

Then we have that

$$\begin{aligned}\mathbb{E}[E_{i\leftrightarrow j}QE_{i\leftrightarrow j}] = \\ = \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_i - e_j)(e_i - e_j)^*Q(e_i - e_j)(e_i - e_j)^* \\ = \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_i^* + e_je_j^*)Q(e_ie_i^* + e_je_j^*) \\ - \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_j^* + e_je_i^*)Q(e_ie_i^* + e_je_j^*) \\ - \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_i^* + e_je_j^*)Q(e_ie_j^* + e_je_i^*) \\ + \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_j^* + e_je_i^*)Q(e_ie_j^* + e_je_i^*).\end{aligned}$$

Consider now the first term of the last equality. It follows that

$$\begin{aligned}\sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_i^*+e_je_j^*)Q(e_ie_i^*+e_je_j^*) &= \\ &= \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_i^*Qe_ie_i^*+e_je_j^*Qe_je_j^*) \\ &\quad+\sum_{(i,j)\in\mathcal{E}} W_{ij}(e_je_j^*Qe_ie_i^*+e_ie_i^*Qe_je_j^*) = \\ &= \sum_{(i,j)\in\mathcal{E}} (W_{ij}Q_{ii}e_ie_i^*+W_{ij}Q_{jj}e_je_j^*) \\ &\quad+\sum_{(i,j)\in\mathcal{E}} (W_{ji}Q_{ji}e_je_i^*+W_{ij}Q_{ij}e_ie_j^*) = \\ &= \operatorname{diag}(W\mathbb{1})\operatorname{dg}(Q)+W\odot Q.\end{aligned}$$

In a similar way it can be shown that

$$\begin{aligned}\sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_j^*+e_je_i^*)Q(e_ie_i^*+e_je_j^*) &\\ \quad+\sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_i^*+e_je_j^*)Q(e_ie_j^*+e_je_i^*) &= \\ = \operatorname{dg}(Q)W+W\operatorname{dg}(Q)+2\operatorname{dg}(A(W\odot Q))&\end{aligned}$$

and that

$$\begin{aligned}\sum_{(i,j)\in\mathcal{E}} W_{ij}(e_ie_j^*+e_je_i^*)Q(e_ie_j^*+e_je_i^*) &= \\ = W\odot Q+\operatorname{dg}(W\operatorname{diag}(Q)\mathbb{1}).&\end{aligned}$$

Plugging together all the contributions we obtain (3.11). □

Complete Graph

Assume that the graph $\mathcal{G}$ describing the feasible communication between nodes is the complete graph. Moreover, assume that each edge has the same probability of being selected. Hence

$$W=\frac{2}{N(N-1)}(\mathbb{1}\mathbb{1}^*-I). \tag{3.12}$$

We have the following

Proposition 3.1.2. *In the case of complete graph it holds*

$$\mathbb{E}[E_{i\leftrightarrow j}]=\frac{2}{N-1}\Omega,$$

and

$$\mathbb{E}[E_{i\leftrightarrow j}\Omega E_{i\leftrightarrow j}]=\frac{4}{N-1}\Omega.$$

PROOF From the technical Lemma 3.1.1 we have that

$$\begin{aligned}\mathbb{E}[E_{i\leftrightarrow j}] &= \operatorname{diag}(W\mathbb{1}) - W \\ &= \frac{2}{N}I - \frac{2}{N(N-1)}(\mathbb{1}\mathbb{1}^* - I) = \frac{2}{N-1}\Omega\end{aligned}$$

and, observing that

$$\begin{aligned}E_{i\leftrightarrow j}\Omega E_{i\leftrightarrow j} = E_{i\leftrightarrow j}^2 &= (e_i - e_j)(e_i - e_j)^*(e_i - e_j)(e_i - e_j)^* \\ &= 2(e_i - e_j)(e_i - e_j)^* = 2E_{i\leftrightarrow j},\end{aligned}$$

we obtain

$$\mathbb{E}[E_{i\leftrightarrow j}\Omega E_{i\leftrightarrow j}] = \frac{4}{N-1}\Omega.$$

□

We can now state the following

Proposition 3.1.3. *In the case of complete graph, the set*

$$\mathcal{J} = \left\{ \Sigma : \Sigma = \begin{bmatrix} a & b \\ b & c \end{bmatrix} \otimes \Omega \right\}$$

is invariant under the transformation in (3.8).

PROOF Let $\Sigma(k) = \Sigma$ with $\Sigma \in \mathcal{J}$. Then, applying Proposition 3.1.2 to (3.8), one can obtain

$$\begin{aligned}\Sigma_{yy}^+ &= \frac{N-2}{N-1}a\Omega + \frac{2(N-2)}{N\lambda(N-1)}b\Omega + \frac{2}{N^2\lambda^2}c\Omega \\ \Sigma_{yz}^+ &= \frac{N\lambda(N-2)-\alpha}{N\lambda(N-1)}b\Omega + \frac{1}{N\lambda}c\Omega \\ \Sigma_{zz}^+ &= \frac{\alpha^2}{N-1}a\Omega - \frac{2\alpha}{N-1}b\Omega + c\Omega\end{aligned}$$

therefore $\Sigma(k+1) \in \mathcal{J}$. □

Corollary. *In the case of a complete graph and under Assumption 3.1.1, we have*

$$\begin{aligned}\Sigma_{yy}(k) &= \xi_{yy}(k)\Omega, \\ \Sigma_{yz}(k) &= \xi_{yz}(k)\Omega, \\ \Sigma_{zz}(k) &= \xi_{zz}(k)\Omega,\end{aligned}$$

where

$$\xi^+ = \begin{bmatrix} \xi_{yy}^+ \\ \xi_{yz}^+ \\ \xi_{zz}^+ \end{bmatrix} = \begin{bmatrix} \frac{N-2}{N-1} & \frac{2(N-2)}{N\lambda(N-1)} & \frac{2}{N^2\lambda^2} \\ 0 & \frac{N\lambda(N-2)-\alpha}{N\lambda(N-1)} & \frac{1}{N\lambda} \\ \frac{\alpha^2}{N-1} & -\frac{2\alpha}{N-1} & 1 \end{bmatrix} \begin{bmatrix} \xi_{yy} \\ \xi_{yz} \\ \xi_{zz} \end{bmatrix} = \Phi_s \xi$$

PROOF The result just follows form the fact that the initial condition $\Sigma(0)$ in (3.9) is in $\mathcal{J}$ and, therefore we can parametrized the trajectories of the system as

$$\Sigma(k) = \begin{bmatrix} \xi_{yy}(k) & \xi_{yz}(k) \\ \xi_{yz}(k) & \xi_{zz}(k) \end{bmatrix} \otimes \Omega$$

because of the invariance stated in proposition above. The proof of the same proposition gives also the update equations for the three parameters of the covariance matrix. □

Theorem 3.1.4. *Consider the network of clocks described before with a symmetric gossip communication protocol over a complete graph and an edge selection probability matrix as in* (3.12)*. Then the covariance* Σ *of the synchronization error converges exponentially to zero if and only if*

$$\alpha < \frac{N\lambda}{2}\left\{\sqrt{N^2 - 2N + 5} - N + 1\right\} =: \bar{\alpha}(N)$$

PROOF The stability of the update equation can be studied by eigenvalue analysis of the matrix Φ_s. The characteristic polynomial is

$$\psi(z) = \det(zI - \Phi_s) = \frac{q_3 z^3 + q_2 z^2 + q_1 z + q_0}{N^3\lambda^3(N-1)^2},$$

where

$$\begin{aligned}
q_3 &= N^3\lambda^3(N-1)^2; \\
q_2 &= N^2\lambda^2(N-1)[\alpha - N\lambda(3N-5)]; \\
q_1 &= N\lambda[-2(N-1)\alpha^2 + N\lambda\alpha + N^2\lambda^2(N-2)(3N-4)]; \\
q_0 &= -2\alpha^3 - N^2\lambda^2(N-2)\alpha - N^3\lambda^3(N-2)^2.
\end{aligned}$$

An efficient way to study the stability of $\psi(z)$ is to apply the Routh criterion to the numerator of the continuous time version obtained by the Möbius transformation $z = (1+s)/(1-s)$

$$\psi_c(s) = \frac{r_3 s^3 + r_2 s^2 + r_1 s + r_0}{N^3\lambda^3(N-1)^2(s-1)^3},$$

where

$$\begin{aligned}
r_3 &= \alpha^3 - N\lambda(N-1)\alpha^2 + N^3\lambda^3(2N-3)^2; \\
r_2 &= -3\alpha^3 + N\lambda(N-1)\alpha^2 - N^2\lambda^2(2N-3)\alpha \\
&\quad + 2N^3\lambda^3(2N-3); \\
r_1 &= 3\alpha^3 + N\lambda(N-1)\alpha^2 + 2N^2\lambda^2(N-2)\alpha + N^3\lambda^3; \\
r_0 &= -\alpha^3 - N\lambda(N-1)\alpha^2 + N^2\lambda^2\alpha.
\end{aligned}$$

To have stability all the four terms r_i must have the same sign. We will now show that, under the technical assumptions $N > 2$ (which is a necessary condition to have a properly randomized algorithm) and $\lambda > 0$ (a necessary condition to have a properly Poisson process) all terms are positive under the condition $\alpha < \bar{\alpha}$.

Observe that the term r_1 is always positive then we have to test the positivity of the remaining terms.

SIGN OF THE TERM r_0 : It is clear that r_0 is positive if and only if

$$\alpha^2 + N\lambda(N-1)\alpha - N^2\lambda^2 < 0$$

which turns out to be true for

$$\alpha < \frac{N\lambda}{2}\left\{\sqrt{N^2 - 2N + 5} - N + 1\right\} = \bar{\alpha}.$$

Our goal now is to proof that $r_2, r_3 > 0$ if $\alpha < \bar{\alpha}$.

SIGN OF THE TERM r_2 : Observing that

$$N - 1 < \zeta < N,$$

where

$$\zeta := \sqrt{N^2 - 2N + 5},$$

we have

$$r_2(0) = N^3\lambda^3(4N - 6),$$
$$r_2(\bar{\alpha}) > \frac{3}{8}N^3\lambda^3(8N - 13),$$

both greater than zero for all $N \geqslant 2$.

To show that r_2 is positive in the entire interval $(0, \bar{\alpha})$ we study the derivative of r_2, that is

$$r_2'(\alpha) = -9\alpha^2 + 2N\lambda(N-1)\alpha - N^2\lambda^2(2N - 3).$$

We have that $r_2'(\alpha) = 0$ for

$$\alpha_{1,2} = \frac{N\lambda}{9}\left\{N - 1 \mp \sqrt{N^2 - 20N + 28}\right\}.$$

For $N \leqslant 18$, $\alpha_{1,2}$ are complex conjugate and hence r_2 is monotonically decreasing for $\alpha \in \mathbb{R}$ (see Figure 2). Otherwise, if $N > 18$, then r_2 has a local minimum in α_1 and a

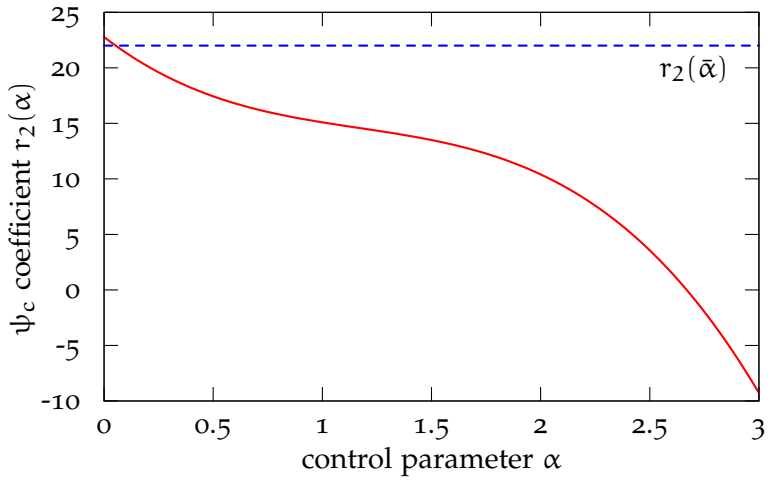

Figure 2: Behavior of $r_2(\alpha)$ for $N = 15$ and $\lambda = 0.05$ (solid line) and the evaluation $r_2(\bar{\alpha})$ (dashed line)

local maximum in α_2 (as reported in Figure 3). However, in this case we have that $\bar{\alpha} < \alpha_1$ if and only if

$$9\zeta + 2\gamma < 11(N-1), \quad \gamma := \sqrt{N^2 - 20N + 28}$$

or, equivalently, if and only if

$$\gamma\zeta < N^2 - 11,$$

which, in turn, conduct to

$$22N^3 - 95N^2 + 156N - 19 > 0, \quad \forall N > 0,$$

which implies that r_2 restricted to the interval $[0\ \bar{\alpha}]$ is still monotonically decreasing.

SIGN OF THE TERM r_3 : Similarly to r_2 we have that

$$r_3(0) = N^3\lambda^3(2N-3)^2 > 0,$$
$$r_3(\bar{\alpha}) > N^3\lambda^3(2N-3)^2 > 0.$$

One can see that r_3 is a concave function for $\alpha \in [0\ \bar{\alpha}]$. Indeed we have that

$$r_3''(\alpha) = 6\alpha - 2N\lambda(N-1),$$

where $r_3'' < 0$ if and only if

$$\alpha < \frac{1}{3}N\lambda(N-1) =: \rho$$

and it holds true that $\bar{\alpha} < \rho$ for all $N \geqslant 2$.

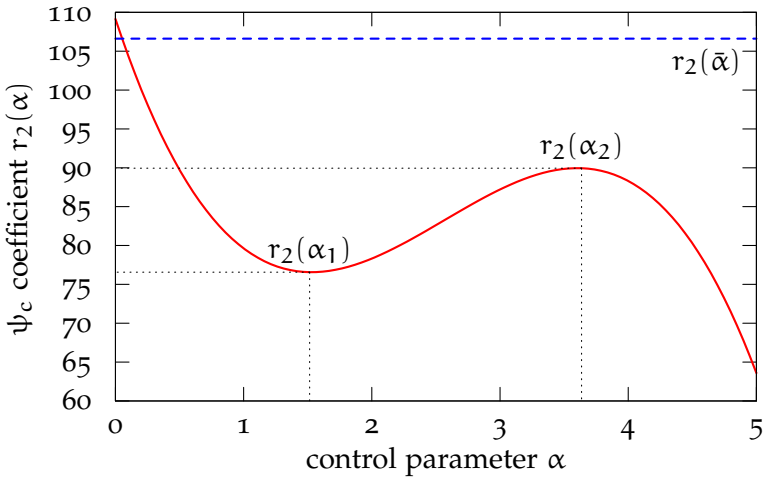

Figure 3: Behavior of $r_2(\alpha)$ for $N = 22$ and $\lambda = 0.05$ (solid line) and the evaluation $r_2(\bar{\alpha})$ (dashed line)

□

Corollary. *Under the same hypothesis of Theorem 3.1.4, a sufficient condition on α for the covariance Σ of the synchronization error to go to zero is that*

$$0 < \alpha \leqslant \lambda$$

PROOF It is not difficult to see that $\lambda < \bar{\alpha}(N)$, indeed

$$\frac{N\lambda}{2}\left\{1 - N + \sqrt{N^2 - 2N + 5}\right\} > \lambda$$

if and only if

$$N\sqrt{N^2 - 2N + 5} > 2 + N(N - 1),$$

or, equivalently, $4(N - 1) > 0$ clearly true for all $N > 1$. Furthermore results that the bound is tight if the number of agents is unknown. This is equivalent to assert that

$$\frac{N\lambda}{2}\left\{1 - N + \sqrt{N^2 - 2N + 5}\right\} \to \lambda, \quad N \to +\infty$$

that is

$$\lim_{N\to+\infty} \frac{2}{1 + \frac{1}{N} + \sqrt{1 - \frac{2}{N} + \frac{5}{N^2}}}\lambda = \lambda.$$

□

Remark 3.1.2. Instead of (2.5), we might consider the updating rule in (2.4). The algorithm resulting in this case satisfies the same properties of (3.4). In particular, for the complete graph, it is possible to see that there still exists a function $\bar{\alpha}(N)$, such that $\bar{\alpha}(N) > \lambda$ for all N, $\lim_{N\to\infty} \bar{\alpha}(N) = \lambda$ and such that the synchronization is attained if and only if $\alpha < \bar{\alpha}$. However the analysis in this case is much more involved; for this reason, in this dissertation, we decide to analyze the algorithm adopting (2.5).

Circulant Graphs

Assume that the graph $\mathcal{G}$ describing the feasible communication between nodes is a circulant graph, i.e. its adjacency matrix A is a circulant (symmetric) matrix. Moreover assume that each edge $(i,j) \in \mathcal{E}$ has the same probability of being selected. Hence

$$W = \frac{2}{\nu N} A,$$

where ν is the degree of the graph, i.e. the number of link of each node. We have the following

Proposition 3.1.5. *In the case of circulant graph with degree ν it holds*

$$\mathbb{E}[E_{i\leftrightarrow j}] = \frac{2}{\nu N}(\nu I - A), \tag{3.13}$$

and

$$\begin{aligned}\mathbb{E}[E_{i\leftrightarrow j} Q E_{i\leftrightarrow j}] = \frac{4}{\nu N}[\nu \operatorname{dg}(Q) + A \odot Q] \\ - \frac{4}{\nu N}(A \operatorname{dg}(Q) + \operatorname{dg}(AQ)),\end{aligned} \tag{3.14}$$

for every symmetric circulant matrix Q.

PROOF From the technical Lemma 3.1.1 we have that

$$\begin{aligned}\mathbb{E}[E_{i\leftrightarrow j}] &= \operatorname{diag}(W\mathbb{1}) - W \\ &= \frac{2}{\nu N}\operatorname{diag}(A\mathbb{1}) - \frac{2}{\nu N}A = \frac{2}{N}\left(I - \frac{1}{\nu}A\right).\end{aligned}$$

Then we have the following

- $\operatorname{diag}(W \operatorname{dg}(Q)\mathbb{1}) = \frac{2}{N}\operatorname{dg}(Q)$;
- $\operatorname{dg}(Q)W = W\operatorname{dg}(Q) = \frac{2}{\nu N} A \operatorname{dg}(Q)$;
- $\operatorname{dg}(A(W \odot Q)) = \frac{2}{\nu N}\operatorname{dg}(AQ)$,

where we used the facts that if M is circulant, then $\mathrm{dg}(M) = M_{11}\mathrm{I}$. Finally, substituting the above equations in (3.11) we obtain (3.14). □

Thus we have that the covariances update (3.7) become

$$\begin{aligned}
\Sigma_{yy}^{+} &= \Sigma_{yy} - \{Z\Sigma_{yy} + \Sigma_{yy}Z\} + \frac{1}{4}X + \frac{1}{N\lambda}\left\{\Sigma_{yz} + \Sigma_{yz}^{*}\right\} \\
&\quad - \frac{1}{N\lambda}\left\{Z\Sigma_{yz} + \Sigma_{yz}^{*}Z\right\} + \frac{2}{N^2\lambda^2}\Sigma_{zz}, \qquad (3.15a)\\
\Sigma_{yz}^{+} &= -\alpha\Sigma_{yy}Z + \frac{\alpha}{4}X + \Sigma_{yz} - Z\Sigma_{yz} \\
&\quad - \frac{\alpha}{N\lambda}\Sigma_{yz}^{*}Z + \frac{1}{N\lambda}\Sigma_{zz}, \text{ and} \qquad (3.15b)\\
\Sigma_{zz}^{+} &= \frac{\alpha^2}{4}X - \alpha\left\{Z\Sigma_{yz} + \Sigma_{yz}^{*}Z\right\} + \Sigma_{zz}, \qquad (3.15c)
\end{aligned}$$

where

$$Z := \frac{1}{2}\mathbb{E}[E_{i\leftrightarrow j}] = \frac{1}{\nu N}(\nu I - A),$$

and

$$\begin{aligned}
X &:= \mathbb{E}[E_{i\leftrightarrow j}QE_{i\leftrightarrow j}] \\
&= \frac{4}{\nu N}[\nu\,\mathrm{dg}(Q) + A\odot Q - (A\,\mathrm{dg}(Q) + \mathrm{dg}(AQ))].
\end{aligned}$$

Notice that from Assumption 3.1.1 results that Σ_{yy}, Σ_{yz} and Σ_{zz} are circulant matrix for all $t \geqslant 0$.

Observe now that a circulant matrix is completely determined by the values of one of its row or column. Let π_M be the generator of the circulant matrix M (e.g. $\pi_M = \mathrm{col}_1(M)$), and let define

$$\pi_{yy} := \pi_{\Sigma_{yy}}, \quad \pi_{yz} := \pi_{\Sigma_{yz}}, \quad \text{and} \quad \pi_{zz} := \pi_{\Sigma_{zz}}.$$

In order to find useful recursive equations for π_{yy}, π_{yz} and π_{zz}, we introduce the $N \times N$ matrices as follows

$$\begin{aligned}
&B := \mathrm{diag}(Ae_1), \quad C := e_1e_1^{*}, \quad \text{and} \\
&D := e_1e_1^{*}A + Ae_1e_1^{*}.
\end{aligned} \qquad (3.16)$$

We can now state the following

Proposition 3.1.6. *In the case of a circulant graph with degree ν and an edge selection probability $W = \frac{2}{\nu N}A$, results that*

$$\pi^{+} = \begin{bmatrix}\pi_{yy}^{+}\\ \pi_{yz}^{+}\\ \pi_{zz}^{+}\end{bmatrix} = \begin{bmatrix}F_{11} & F_{12} & F_{13}\\ F_{21} & F_{22} & F_{23}\\ F_{31} & F_{32} & F_{33}\end{bmatrix}\begin{bmatrix}\pi_{yy}\\ \pi_{yz}\\ \pi_{zz}\end{bmatrix} = F\pi, \qquad (3.17)$$

where

$$
\begin{aligned}
F_{11} &= \frac{1}{\nu N}[(N-2)\nu I + 2A + B + \nu C - D];\\
F_{12} &= \frac{2}{\nu N^2 \lambda}[(N-1)\nu I + A];\\
F_{13} &= \frac{2}{N^2\lambda^2} I;\\
F_{21} &= \frac{\alpha}{\nu N}(-\nu I + A + B + \nu C - D);\\
F_{22} &= \frac{1}{\nu N^2 \lambda}[(N^2\lambda - N\lambda - \alpha)\nu I + (N\lambda + \alpha)A];\\
F_{23} &= \frac{1}{N\lambda} I;\\
F_{31} &= \frac{\alpha^2}{\nu N}(B + \nu C - D);\\
F_{32} &= -\frac{2\alpha}{\nu N}(\nu I - A);\\
F_{33} &= I.
\end{aligned}
$$

PROOF From the definition of A, B, C, and D we have that

$$
\begin{aligned}
\pi_{A\Sigma_{yy}} &= A\pi_{yy}, \quad \pi_{A\Sigma_{yz}} = A\pi_{yz}, \quad \pi_{A\Sigma_{zz}} = A\pi_{zz},\\
\pi_{A\odot\Sigma_{yy}} &= B\pi_{yy}, \quad \pi_{A\odot\Sigma_{yz}} = B\pi_{yz}, \quad \pi_{A\odot\Sigma_{zz}} = B\pi_{zz},\\
\pi_{dg(\Sigma_{yy})} &= C\pi_{yy}, \quad \pi_{dg(\Sigma_{yz})} = C\pi_{yz}, \quad \pi_{dg(\Sigma_{zz})} = C\pi_{zz},
\end{aligned}
$$

and

$$
\begin{aligned}
\pi_{A\,dg(\Sigma_{yy})+dg(A\Sigma_{yy})} &= D\pi_{yy},\\
\pi_{A\,dg(\Sigma_{yz})+dg(A\Sigma_{yz})} &= D\pi_{yz},\\
\pi_{A\,dg(\Sigma_{zz})+dg(A\Sigma_{zz})} &= D\pi_{zz}.
\end{aligned}
$$

Hence, substituting the above equations in (3.15), and bearing in mind that if the matrices K and T are circulant, then $KT = TK$, we obtain (3.17). □

From directly computation, one can see that the $3N$-dimensional vectors

$$
v^{(1)} = \begin{bmatrix} \mathbb{1} \\ 0 \\ 0 \end{bmatrix}, \quad v^{(2)} = \begin{bmatrix} \mathbb{1} \\ \frac{N\lambda}{2}\mathbb{1} \\ 0 \end{bmatrix}, \quad \text{and} \quad v^{(3)} = \frac{N^2\lambda^2}{2}\begin{bmatrix} 0 \\ 0 \\ \mathbb{1} \end{bmatrix},
$$

belong to the same Jordan chain relative to the eigenvalue 1 of the operator matrix F, namely

$$
(F - I)v^{(1)} = 0, \quad (F - I)v^{(2)} = v^{(1)}, \quad \text{and} \quad (F - I)v^{(3)} = v^{(2)}.
$$

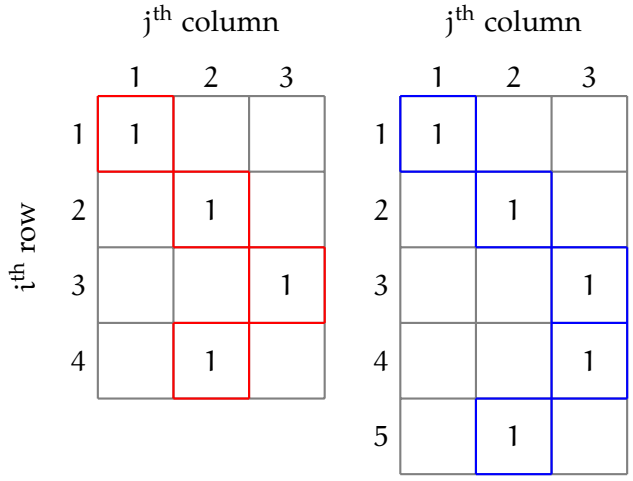

Figure 4: Structure of the matrix S for N = 4 (left) and N = 5 (right).

We persuade ourselves that these (generalized) eigenvectors are the only ones related to the eigenvalue 1, and that the remaining eigenvalues are inside the unit disc. In support of our guess we will provide some numerical insights. To do that, in order to further simplify the problem, we perform a system reduction exploiting the fact that the covariances are symmetric matrices. Indeed, only part of the generators is essential. More precisely, only the first $r := \lceil \frac{N+1}{2} \rceil$ elements[2] of the generators π_{yy}, π_{yz}, and π_{zz} are strictly necessary, because the other ones are a merely repetition. Consequently the system (3.17) could be rewritten as

$$\gamma^+ = \begin{bmatrix} \gamma^+_{yy} \\ \gamma^+_{yz} \\ \gamma^+_{zz} \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} \begin{bmatrix} \gamma_{yy} \\ \gamma_{yz} \\ \gamma_{zz} \end{bmatrix} = G\gamma,$$

where $G = (I_3 \otimes R)F(I_3 \otimes S)$, $R = S^\dagger$, and S is an $N \times r$ matrix such that

$$S_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 1, & \text{if } i = r+k,\ j = r-k+1,\ k = 1,\dots,r-1, \text{ and N odd}, \\ 1, & \text{if } i = r+k,\ j = r-k,\ k = 1,\dots,r-2, \text{ and N even}, \\ 0, & \text{otherwise.} \end{cases}$$

An illustration of the matrix S is shown in Figure 4.

So the 3r-dimensional G operator just computed encode the information that the covariances Σ_{yy}, Σ_{yz}, and Σ_{zz} are both circulant and symmetric matrices. Let $\Lambda(G)$ the spectrum of G

2 with $\lceil x \rceil$ we denote the smallest integer not less than x.

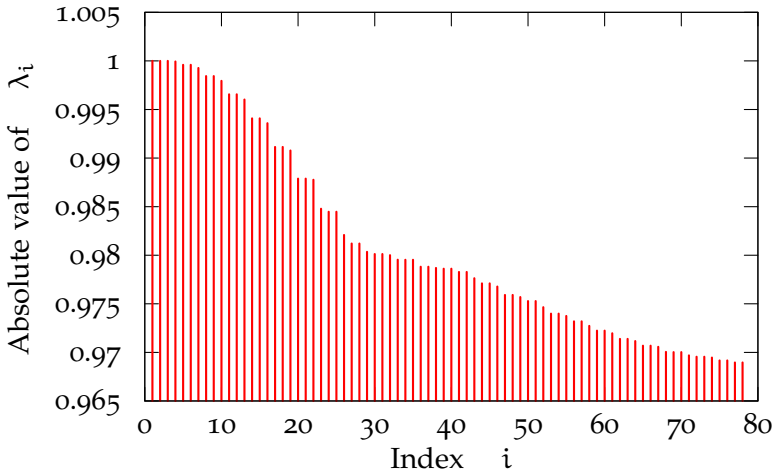

Figure 5: Spectrum of the operator G for $N = 50$, $\nu = 4$, and $\alpha = \lambda/2$.

and let sort the eigenvalues by their absolute value (descending order), namely

$$\Lambda(G) = \{\lambda_1, \lambda_2, \ldots, \lambda_{3r}\}, \quad |\lambda_1| \geqslant |\lambda_2| \geqslant \cdots \geqslant |\lambda_{3r}|.$$

We report an example of this set in Figure 5. From numerical simulation we notice that the absolute values of $\lambda_i, i = 1, \ldots, 3r$ cluster around 1 as N growth and, although is not supported by theoretical results, seems that all the eigenvalues belong to the unit disc.

3.2 ASYMMETRIC GOSSIP

In this case we suppose that for $i \in \{1, \ldots, N\}$ and for $k \in \mathbb{N}$, the information $x_i'(T_{tx,i}(k))$ is sent by node i to only one of its neighbors, which is randomly selected within the set $\mathcal{N}_i$.

Now, without loss of generality, assume that the node i transmits, at time $T_{tx,i}(k)$, the information $x_i'(T_{tx,i}(k))$ to node j. Based on the information received, the node j *instantaneously* applies to its current state $x_j(T_{tx,i}(k))$ the following correction

$$u = \begin{bmatrix} u' \\ u'' \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 \\ \alpha \end{bmatrix} \left(x_i'(T_{tx,i}(k)) - x_j'(T_{tx,i}(k)) \right),$$

where α, as before, is a control parameter such that $\alpha > 0$. From (2.6), it follows that

$$x_j'(T_{tx,i}^+(k)) = \frac{1}{2}\left(x_j'(T_{tx,i}(k)) + x_i'(T_{tx,i}(k))\right), \tag{3.18a}$$

$$x_j''(T_{tx,i}^+(k)) = x_j''(T_{tx,i}(k)) + \frac{\alpha}{2}\left(x_i'(T_{tx,i}(k)) - x_j'(T_{tx,i}(k))\right). \tag{3.18b}$$

Observe we have that $T_{tx,i}(k)$ represents an update time instant for node j, i. e. $T_{tx,i}(k) = T_{up,j}(k')$ for some $k' \in \mathbb{N}$.

From now on, all the deductive reasonings follow the same operating principles outlined in the symmetric case. Let us start by providing a convenient vector-form description of the evolution of the clocks' network. Combining (2.7) with (3.18), we can write

$$x(k+1) = \begin{bmatrix} I - \frac{1}{2}E_{i\to j} & \delta t_k D \\ -\frac{\alpha}{2}E_{i\to j} & I \end{bmatrix} x(k), \tag{3.19}$$

where

$$E_{i\to j} := e_j e_j^* - e_j e_i^*, \qquad t_k := T_{up}^-(k),$$
$$\delta t_k = t_{k+1} - t_k, \qquad x(k) := x(t_k^-).$$

In next sections, as done for the symmetric protocol, we provide a mean square analysis focusing our attention on the complete and the circulant graph topologies.

3.2.1 Mean-Square Analysis

Since for any $i, j \in \{1, \dots, N\}$, $i \neq j$, $E_{i\to j}\Omega = E_{i\to j}$, system (3.19) can be rewritten as

$$\begin{bmatrix} y(k+1) \\ z(k+1) \end{bmatrix} = \begin{bmatrix} I - \frac{1}{2}\Omega E(k) & \delta t_k \\ -\frac{\alpha}{2}\Omega D E(k) & I \end{bmatrix} \begin{bmatrix} y(k) \\ z(k) \end{bmatrix}, \tag{3.20}$$

where $y(k)$ and $z(k)$ are defined as in Section 3.1.1, and where $E(k) = E_{i\to j}$ if, during the k-th iteration node i and node j are, respectively, the transmitting and the receiving nodes. Clearly x' reaches the asymptotic synchronization if and only if

$$\lim_{k\to\infty} y(k) = 0.$$

The goal is to perform a mean-square analysis of (3.20). The objective is to study the evolution of

$$\Sigma(k+1) = \mathbb{E}[A(k)\Sigma(k)A^*(k)] \tag{3.21}$$

where

$$A(k) := \begin{bmatrix} I - \frac{1}{2}\Omega E(k) & \delta t_k \\ -\frac{\alpha}{2}\Omega D E(k) & I \end{bmatrix}.$$

Also in this case we perform our analysis assuming $D = I$. From the above recursive equation we obtain the following

$$\begin{aligned}\Sigma^+_{yy} = \Sigma_{yy} &- \frac{1}{2}\left\{\Omega\mathbb{E}[E_{i\to j}]\Sigma_{yy} + \Sigma_{yy}\mathbb{E}[E^*_{i\to j}]\Omega\right\} \\ &+ \frac{1}{4}\Omega\mathbb{E}[E_{i\to j}\Sigma_{yy}E^*_{i\to j}]\Omega + \frac{1}{N\lambda}\left\{\Sigma_{yz} + \Sigma^*_{yz}\right\} \\ &- \frac{1}{2N\lambda}\left\{\Omega\mathbb{E}[E_{i\to j}]\Sigma_{yz} + \Sigma^*_{yz}\mathbb{E}[E^*_{i\to j}]\Omega\right\} + \frac{2}{N^2\lambda^2}\Sigma_{zz},\end{aligned} \tag{3.22a}$$

$$\begin{aligned}\Sigma^+_{yz} = &-\frac{\alpha}{2}\Sigma_{yy}\mathbb{E}[E^*_{i\to j}]\Omega + \frac{\alpha}{4}\Omega\mathbb{E}[E_{i\to j}\Sigma_{yy}E^*_{i\to j}]\Omega + \Sigma_{yz} \\ &- \frac{1}{2}\Omega\mathbb{E}[E_{i\to j}]\Sigma_{yz} - \frac{\alpha}{2N\lambda}\Sigma^*_{yz}\mathbb{E}[E^*_{i\to j}]\Omega + \frac{1}{N\lambda}\Sigma_{zz},\end{aligned} \tag{3.22b}$$

$$\begin{aligned}\Sigma^+_{zz} = &\frac{\alpha^2}{4}\Omega\mathbb{E}[E_{i\to j}\Sigma_{yy}E^*_{i\to j}]\Omega \\ &- \frac{\alpha}{2}\left\{\Omega\mathbb{E}[E_{i\to j}]\Sigma_{yz} + \Sigma^*_{yz}\mathbb{E}[E^*_{i\to j}]\Omega\right\} + \Sigma_{zz},\end{aligned} \tag{3.22c}$$

therefore, as for the symmetric algorithm, the covariance matrix Σ updates according to a linear transformation

$$\Sigma(k+1) = F[\Sigma(k)]. \tag{3.23}$$

Let us introduce the following technical result.

Lemma 3.2.1. *Let $\mathcal{G} = (V, \mathcal{E})$ be a given graph without any self-loop and let W be the matrix defined as in* (3.5)*. Let Q any symmetric matrix. Then*

$$\mathbb{E}[E_{i\to j}] = \operatorname{diag}(W\mathbb{1}) - W \tag{3.24}$$

and

$$\begin{aligned}\mathbb{E}[E_{i\to j}QE^*_{i\to j}] = &\operatorname{diag}(W\mathbb{1})\operatorname{dg}(Q) \\ &+ \operatorname{diag}(\operatorname{dg}(Q)W\mathbb{1}) - 2\operatorname{diag}((W\odot Q)\mathbb{1}).\end{aligned} \tag{3.25}$$

PROOF We have that

$$\mathbb{E}[E_{i\to j}] = \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_j e_j^* - e_j e_i^*) = \operatorname{diag}(W\mathbb{1}) - W,$$

and also that

$$\begin{aligned}\mathbb{E}[E_{i\to j}QE_{i\to j}^*] &= \sum_{(i,j)\in\mathcal{E}} W_{ij}(e_je_j^* - e_je_i^*)Q(e_je_j^* - e_ie_j^*)\\ &= \sum_{(i,j)\in\mathcal{E}} W_{ij}(Q_{jj} - Q_{ij} - Q_{ji} + Q_{ii})e_je_j^*,\end{aligned}$$

or, equivalently

$$\begin{aligned}\mathbb{E}[E_{i\to j}QE_{i\to j}^*] = &\operatorname{diag}(W\mathbb{1})\operatorname{dg}(Q)\\ &- 2\operatorname{diag}((W\odot Q)\mathbb{1}) + \operatorname{diag}(\operatorname{dg}(Q)W\mathbb{1}).\end{aligned}$$

□

Notice that a relevant difference with the symmetric version of the protocol is the probability edge selection. Indeed, due to the one-way link pattern, results that each edge $(i,j)\in\mathcal{E}$ has half probability of being selected, namely

$$W = \frac{1}{N(N-1)}(\mathbb{1}\mathbb{1}^* - I), \tag{3.26}$$

for the complete graph, and

$$W = \frac{1}{\nu N}A, \tag{3.27}$$

for a circulant graph of degree ν.

Complete Graph

We can now state the following

Proposition 3.2.2. *In the case of complete graph it holds*

$$\mathbb{E}[E_{i\to j}] = \frac{1}{N-1}\Omega,$$

and

$$\mathbb{E}[E_{i\to j}\Omega E_{i\to j}^*] = \frac{2}{N}I.$$

PROOF From the technical Lemma 3.2.1 we have that

$$\begin{aligned}\mathbb{E}[E_{i\to j}] &= \operatorname{diag}(W\mathbb{1}) - W\\ &= \frac{1}{N}I - \frac{1}{N(N-1)}(\mathbb{1}\mathbb{1}^* - I) = \frac{1}{N-1}\Omega\end{aligned}$$

and

$$\begin{aligned}\mathbb{E}[E_{i\to j}QE_{i\to j}^{*}] &= \operatorname{diag}(W\mathbb{1})\operatorname{dg}(Q)\\ &\quad - 2\operatorname{diag}((W\odot Q)\mathbb{1}) + \operatorname{diag}(\operatorname{dg}(Q)W\mathbb{1})\\ &= \frac{N-1}{N^2}I + \frac{N-1}{N^2}I + \frac{2}{N^2}I = \frac{2}{N}I.\end{aligned}$$

□

Furthermore we have that

Proposition 3.2.3. *In the case of complete graph, the set*

$$\mathcal{J} = \left\{ \Sigma : \Sigma = \begin{bmatrix} a & b \\ b & c \end{bmatrix} \otimes \Omega \right\}$$

is invariant under the transformation in (3.23)*.*

PROOF Let $\Sigma(k) = \Sigma$ with $\Sigma \in \mathcal{J}$. Then, applying Proposition 3.2.2 to (3.23), one can obtain

$$\begin{aligned}\Sigma_{yy}^{+} &= \frac{2N^2-3N-1}{2N(N-1)}a\Omega + \frac{2N-3}{N\lambda(N-1)}b\Omega + \frac{2}{N^2\lambda^2}c\Omega\\ \Sigma_{yz}^{+} &= -\frac{\alpha}{2N(N-1)}a\Omega + \frac{2N^2\lambda-3N\lambda-\alpha}{2N\lambda(N-1)}b\Omega + \frac{1}{N\lambda}c\Omega\\ \Sigma_{zz}^{+} &= \frac{\alpha^2}{2N}a\Omega - \frac{\alpha}{N-1}b\Omega + c\Omega\end{aligned}$$

therefore $\Sigma(k+1) \in \mathcal{J}$. □

Corollary. *In the case of a complete graph and under Assumption 3.1.1, we have*

$$\begin{aligned}\Sigma_{yy}(k) &= \xi_{yy}(k)\Omega,\\ \Sigma_{yz}(k) &= \xi_{yz}(k)\Omega,\\ \Sigma_{zz}(k) &= \xi_{zz}(k)\Omega,\end{aligned}$$

where

$$\xi^{+} = \begin{bmatrix}\xi_{yy}^{+}\\ \xi_{yz}^{+}\\ \xi_{zz}^{+}\end{bmatrix} = \begin{bmatrix}\frac{2N^2-3N-1}{2N(N-1)} & \frac{2N-3}{N\lambda(N-1)} & \frac{2}{N^2\lambda^2}\\ -\frac{\alpha}{2N(N-1)} & \frac{2N^2\lambda-3N\lambda-\alpha}{2N\lambda(N-1)} & \frac{1}{N\lambda}\\ \frac{\alpha^2}{2N} & -\frac{\alpha}{N-1} & 1\end{bmatrix}\begin{bmatrix}\xi_{yy}\\ \xi_{yz}\\ \xi_{zz}\end{bmatrix} = \Phi_a\xi$$

PROOF The result just follows form the fact that the initial condition $\Sigma(0)$ in (3.9) is in $\mathcal{J}$ and, therefore we can parametrized the trajectories of the system as

$$\Sigma(k) = \begin{bmatrix}\xi_{yy}(k) & \xi_{yz}(k)\\ \xi_{yz}(k) & \xi_{zz}(k)\end{bmatrix} \otimes \Omega$$

because of the invariance stated in proposition above. The proof of the same proposition gives also the update equations for the three parameters of the covariance matrix. □

Theorem 3.2.4. *Consider the network of clocks described before with an asymmetric gossip communication protocol over a complete graph and an edge selection probability matrix as in* (3.26)*. Then the covariance* Σ *of the synchronization error converges exponentially to zero if and only if*

$$\alpha < \frac{N\lambda}{N-1}\left\{\sqrt{N^4-4N^3+9N^2-8N+3}-N^2+2N-2\right\} =: \bar{\alpha}(N)$$

PROOF The stability of the update equation can be studied by eigenvalue analysis of the matrix Φ_a. The characteristic polynomial is

$$\psi(z) = \det(zI-\Phi_a) = \frac{q_3z^3+q_2z^2+q_1z+q_0}{4N^4\lambda^3(N-1)^2},$$

where

$$\begin{aligned}
q_3 &= 4N^4\lambda^3(N-1)^2;\\
q_2 &= -2N^3\lambda^2(N-1)(6N^2\lambda-8N\lambda-\lambda-\alpha);\\
q_1 &= -N\lambda[4(N-1)^2\alpha^2-5N\lambda(N-1)\alpha\\
&\qquad -N^2\lambda^2(4N^2-4N-1)(3N-5)];\\
q_0 &= -2(N-1)\alpha^3-4N\lambda\alpha^2\\
&\quad -N^2\lambda^2(2N^2+N-7)\alpha\\
&\qquad -N^3\lambda^3(4N^3-12N^2+7N+3).
\end{aligned}$$

An efficient way to study the stability of $\psi(z)$ is to apply the Routh criterion to the numerator of the continuous time version obtained by the Möbius transformation $z = (1+s)/(1-s)$

$$\psi_c(s) = \frac{r_3s^3+r_2s^2+r_1s+r_0}{2N^4\lambda^3(N-1)^2(s-1)^3},$$

where

$$\begin{aligned}
r_3 &= -(N-1)\alpha^3+2N^2\lambda(N-2)\alpha^2-2N^2\lambda^2(2N-3)\alpha\\
&\qquad -N^3\lambda^3(4N^2-5N-1)(4N-5);\\
r_2 &= 3(N-1)\alpha^3-2N\lambda(N^2-2N-2)\alpha^2\\
&\quad +N^2\lambda^2(4N^2+3N-13)\alpha-2N^3\lambda^3(4N^2-3N-3);\\
r_1 &= -3(N-1)\alpha^3-2N\lambda(N^2-2N+4)\alpha^2\\
&\quad -2N^2\lambda^2(2N^2-N-4)\alpha-N^3\lambda^3(N+1);\\
r_0 &= (N-1)\alpha^3+2N\lambda(N^2-2N+2)\alpha^2-N^2\lambda^2(N+1)\alpha.
\end{aligned}$$

To have stability all the four terms r_i must have the same sign. We will now show that, under the technical assumptions $N \geqslant 2$ (which is a necessary condition to have a properly randomized algorithm) and $\lambda > 0$ (a necessary condition to have a properly Poisson process) all terms are negative under the condition $\alpha < \bar{\alpha}$.

Observe that the term r_1 is always negative then we have to test the negativity of the remaining terms.

SIGN OF THE TERM r_0: It is clear that r_0 is negative if and only if

$$(N-1)\alpha^2 + 2N\lambda(N^2-2N+2)\alpha - N^2\lambda^2(N+1) < 0$$

which turns out to be true for

$$\alpha < \frac{N\lambda}{N-1}\left\{\sqrt{N^4-4N^3+9N^2-8N+3} - N^2 + 2N - 2\right\} = \bar{\alpha}.$$

Our goal now is to proof that $r_2, r_3 < 0$ if $\alpha < \bar{\alpha}$.

SIGN OF THE TERM r_2: We have that

$$\begin{aligned} r_2(0) &= -2N^3\lambda^3(4N^2-3N-3), \\ r_2(\bar{\alpha}) &= 2N^3\lambda^3\Big\{8N^2(\zeta-N^2) + 2N(15N^2-7\zeta) \\ &\qquad + 21\zeta - 73N^2 - 35\Big\}, \end{aligned}$$

where

$$\zeta := \sqrt{N^4-4N^3+9N^2-8N+3}.$$

One can easily see that $r_2(0) < 0$ for all $N > 0$ whereas $r_2(\bar{\alpha}) < 0$ if and only if

$$(8N^2-14N+21)\zeta < 8N^4 - 30N^3 + 73N^2 + 35,$$

or, equivalently, if and only if

$$\zeta < \underbrace{N^2-2N+3}_{q(N)} + \frac{\overbrace{84N-28}^{r(N)}}{8N^2-14N+21},$$

where $q(N)$ and $r(N)$ denote respectively the quotient and the remainder of the polynomial long division

$$d(N) = \frac{8N^4-30N^3+73N^2+35}{8N^2-14N+21}.$$

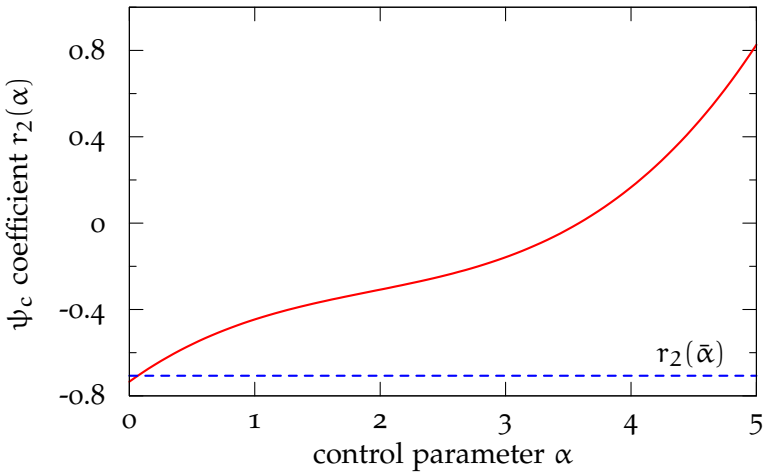

Figure 6: Behavior of $r_2(\alpha)$ for $N = 10$ and $\lambda = 0.01$ (solid line) and the evaluation $r_2(\bar{\alpha})$ (dashed line).

It is not difficult to see that $\zeta < q(N)$ and this prove the negativity of $r_2(\bar{\alpha})$. Sure enough we have

$$\zeta^2 < q^2(N) \quad \text{if and only if} \quad N^2 - 4N + 6 > 0,$$

clearly true for all $N \in \mathbb{N}$.

To show that r_2 is negative in the entire interval $(0, \bar{\alpha})$ we study the derivative of r_2, that is

$$r_2'(\alpha) = 9(N-1)\alpha^2 - 4N\lambda(N^2 - 2N - 2)\alpha + N^2\lambda^2(4N^2 + 3N - 13).$$

We have that $r_2'(\alpha) = 0$ for

$$\alpha_{1,2} = \frac{N\lambda}{9(N-1)}\Big\{2(N^2 - 2N - 2) \mp \sqrt{4N^4 - 52N^3 + 9N^2 + 176N - 101}\Big\}.$$

For $N \leqslant 12$, $\alpha_{1,2}$ are complex conjugate and hence r_2 is monotonically increasing for $\alpha \in \mathbb{R}$ (see Figure 6). Otherwise, if $N > 12$, then r_2 has a local maximum in α_1 and a local minimum in α_2 (as reported in Figure 7). However, in this case, we have that $\bar{\alpha} < \alpha_1$ if and only if $A < B$, where

$$A = 9\zeta + \gamma, \quad \gamma := \sqrt{4N^4 - 52N^3 + 9N^2 - 176N - 101};$$
$$B = 11N^2 - 22N + 14,$$

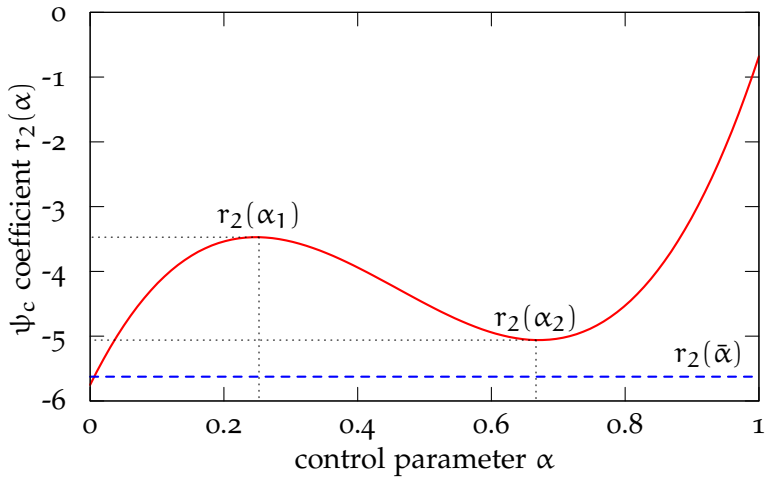

Figure 7: Behavior of $r_2(\alpha)$ for $N = 15$ and $\lambda = 0.01$ (solid line) and the evaluation $r_2(\bar{\alpha})$ (dashed line).

or, equivalently, $A^2 < B^2$ that leads to

$$\gamma\zeta < 2N^4 - 6N^3 + 3N^2 - 8N + 3,$$

which, in turn, conduct to

$$\begin{aligned} &44N^7 - 205N^6 + 292N^5 + 413N^4 \\ &\quad - 1844N^3 + 2372N^2 - 1384N + 312 > 0, \quad \forall N \geqslant 2, \end{aligned}$$

which implies that r_2 restricted into the interval $[0\ \bar{\alpha}]$ is still monotonically increasing.

SIGN OF THE TERM r_3: Similarly to r_2, observing that

$$(N-1)^2 < \zeta < N^2,$$

we have

$$\begin{aligned} r_3(0) &= -N^3\lambda^3(4N-5)(4N^2-5N-1) < 0, \\ r_3(\bar{\alpha}) &< -N^3\lambda^3(4N-5)(4N^2-5N+3) < 0. \end{aligned}$$

One can see that, if $N \geqslant 5$, then r_3 is a convex function for $\alpha \in [0\ \bar{\alpha}]$. Indeed we have that

$$r_3''(\alpha) = -6(N-1)\alpha + 4N^2\lambda(N-2),$$

where $r_3'' > 0$ if and only if

$$\alpha < \frac{2}{3}\frac{N-2}{N-1}N^2\lambda =: \rho$$

and, for $N \geqslant 5$, it holds true that $\bar{\alpha} < \rho$.

Finally for $N = 2, 3, 4$ we obtain respectively:

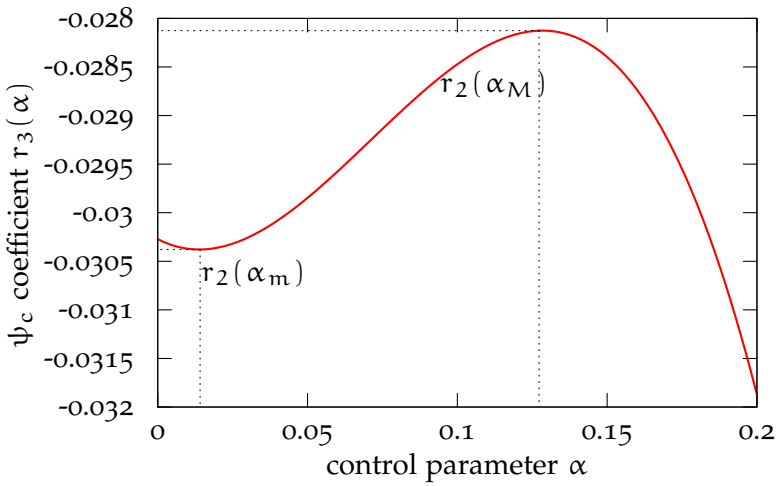

Figure 8: Behavior of $r_3(\alpha)$ for $N = 4$ and $\lambda = 0.01$.

- $r_3'(\alpha) = -3\alpha^2 - 8\lambda < 0 \quad \forall\alpha$;
- $r_3'(\alpha) = -6(\alpha - 3\lambda)^2 \leqslant 0 \quad \forall\alpha$;
- $r_3'(\alpha) = -9\alpha^2 + 128\lambda\alpha + 160\lambda^2 \geqslant 0$ if and only if

  $$\alpha_m := 1.38\lambda < \alpha < 12.84\lambda =: \alpha_M,$$

  so α_M is a local maximum and $r_3(\alpha_M) < 0$ (as illustrated in Figure 8).

□

Corollary. *Under the same hypothesis of Theorem 3.2.4, a sufficient condition on α for the covariance Σ of the synchronization error to go to zero is that*

$$0 < \alpha \leqslant \frac{\lambda}{2}$$

PROOF It is not difficult to see that $\lambda/2 < \bar{\alpha}(N)$ for all $N \geqslant 2$, indeed

$$\frac{N\lambda}{N-1}\left\{\sqrt{N^4 - 4N^3 + 9N^2 - 8N + 3} - N^2 + 2N - 2\right\} > \frac{\lambda}{2},$$

if and only if

$$12N^3 - 21N^2 + 10N - 1 > 0,$$

clearly true for all $N > 1$. Furthermore results that the bound is tight if the number of agents is unknown. This is equivalent to assert that

$$\lim_{N\to\infty} \frac{N\lambda}{N-1}\left\{\sqrt{N^4 - 4N^3 + 9N^2 - 8N + 3} - N^2 + 2N - 2\right\} \to \frac{\lambda}{2},$$

that is

$$\lim_{N\to+\infty} \frac{1+\frac{1}{N}}{\sqrt{1-\frac{4}{N}+\frac{9}{N^2}-\frac{8}{N^3}+\frac{3}{N^4}}+1-\frac{2}{N}+\frac{2}{N^2}}\lambda = \frac{\lambda}{2}.$$

□

Circulant Graphs

As highlighted in the previous section, the edge selection probability matrix W is as in (3.27), therefore we have the following

Proposition 3.2.5. *In the case of circulant graph with degree ν it holds*

$$\mathbb{E}[E_{i\to j}] = \frac{1}{\nu N}(\nu I - A), \tag{3.28}$$

and

$$\mathbb{E}[E_{i\to j} Q E_{i\to j}^*] = \frac{2}{\nu N}[\nu\, dg(Q) - dg(AQ))], \tag{3.29}$$

for every symmetric circulant matrix Q.

PROOF From the technical Lemma 3.2.1 we have that

$$\begin{aligned}\mathbb{E}[E_{i\to j}] &= \operatorname{diag}(W\mathbb{1}) - W \\ &= \frac{1}{\nu N}\operatorname{diag}(A\mathbb{1}) - \frac{1}{\nu N}A = \frac{1}{N}\left(I - \frac{1}{\nu}A\right).\end{aligned}$$

Then we have the following

- $\operatorname{diag}(dg(Q)W\mathbb{1}) = \frac{1}{N}\, dg(Q)$;
- $dg((W\odot Q)\mathbb{1}) = \frac{1}{\nu N}\, dg(AQ)$,

where we used the facts that, if M is circulant, then $dg(M) = M_{11}I$. Finally, substituting the above equations in (3.25) we obtain (3.29). □

Thus we have that the covariances update (3.22) become

$$\begin{aligned}\Sigma_{yy}^+ &= \Sigma_{yy} - \{\Omega Z \Sigma_{yy} + \Sigma_{yy} Z \Omega\} + \frac{1}{4}\Omega X \Omega \\ &\quad + \frac{1}{N\lambda}\left\{\Sigma_{yz} + \Sigma_{yz}^*\right\} - \frac{1}{N\lambda}\left\{\Omega Z \Sigma_{yz} + \Sigma_{yz}^* Z \Omega\right\} \\ &\quad + \frac{2}{N^2\lambda^2}\Sigma_{zz}, \end{aligned} \tag{3.30a}$$

$$\begin{aligned}\Sigma_{yz}^+ &= -\alpha\Sigma_{yy} Z\Omega + \frac{\alpha}{4}\Omega X\Omega + \Sigma_{yz} - \Omega Z \Sigma_{yz} \\ &\quad - \frac{\alpha}{N\lambda}\Sigma_{yz}^* Z\Omega + \frac{1}{N\lambda}\Sigma_{zz}, \text{ and}\end{aligned} \tag{3.30b}$$

$$\Sigma_{zz}^+ = \frac{\alpha^2}{4}\Omega X\Omega - \alpha\left\{\Omega Z\Sigma_{yz} + \Sigma_{yz}^* Z\Omega\right\} + \Sigma_{zz}, \tag{3.30c}$$

where

$$Z := \frac{1}{2}\mathbb{E}[E_{i\leftrightarrow j}] = \frac{1}{2\nu N}(\nu I - A),$$

and

$$X := \mathbb{E}[E_{i\leftrightarrow j} Q E_{i\leftrightarrow j}] = \frac{2}{\nu N}[\nu \operatorname{dg}(Q) - \operatorname{dg}(AQ))].$$

Notice that from Assumption 3.1.1 results that Σ_{yy}, Σ_{yz} and Σ_{zz} are circulant matrix for all $t \geqslant 0$.

As done in the symmetric case, we want to find useful recursive equations for the generators π_{yy}, π_{yz}, and π_{zz} of the covariances just mentioned. In order to simplify the notation, let us introduce the following $N \times N$ matrices

$$\Theta := e_1 e_1^* A, \quad L := \mathbb{1} e_1^*, \quad \text{and} \quad M := \mathbb{1} e_1^* A. \tag{3.31}$$

We can now state the following

Proposition 3.2.6. *In the case of a circulant graph with degree ν and an edge selection probability $W = \frac{1}{\nu N} A$, results that*

$$\pi^+ = \begin{bmatrix} \pi^+_{yy} \\ \pi^+_{yz} \\ \pi^+_{zz} \end{bmatrix} = \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} \pi_{yy} \\ \pi_{yz} \\ \pi_{zz} \end{bmatrix} = F\pi, \tag{3.32}$$

where

$$\begin{aligned}
F_{11} &= \frac{1}{2\nu N^2}[2(N-1)N\nu I + 2NA + \nu NC - N\Theta - \nu L + M]; \\
F_{12} &= \frac{1}{\nu N^2 \lambda}[(2N-1)\nu I + A]; \\
F_{13} &= \frac{2}{N^2\lambda^2} I; \\
F_{21} &= \frac{\alpha}{2\nu N^2}(-\nu N I + NA + \nu NC - N\Theta - \nu L + M); \\
F_{22} &= \frac{1}{2\nu N^2 \lambda}[(2N^2\lambda - N\lambda - \alpha)\nu I + (N\lambda + \alpha)A]; \\
F_{23} &= \frac{1}{N\lambda} I; \\
F_{31} &= \frac{\alpha^2}{2\nu N^2}(\nu NC - N\Theta - \nu L + M); \\
F_{32} &= -\frac{\alpha}{\nu N}(\nu I - A); \\
F_{33} &= I.
\end{aligned}$$

PROOF Observe that we are interested in the generators

$$\pi_{\Omega Z Q}, \quad Q = \Sigma_{yy}, \Sigma_{yz} \quad \text{and} \quad \pi_{\Omega X \Omega}.$$

Moreover, notice that

$$\pi_{\Omega Z Q} = \pi_{Z Q} \quad \text{and} \quad \pi_{\Omega X \Omega} = \pi_X - \frac{1}{N}\pi_{X \mathbb{1}\mathbb{1}^*}.$$

Now, from Proposition 3.1.6 we have that

$$\begin{aligned} \pi_{A\Sigma_{yy}} &= A\pi_{yy}, & \pi_{A\Sigma_{yz}} &= A\pi_{yz}, & \pi_{A\Sigma_{zz}} &= A\pi_{zz}, \\ \pi_{dg(\Sigma_{yy})} &= C\pi_{yy}, & \pi_{dg(\Sigma_{yz})} &= C\pi_{yz}, & \pi_{dg(\Sigma_{zz})} &= C\pi_{zz}, \end{aligned}$$

and, from the definition of Θ, L, and M in (3.31) result

$$\begin{aligned} \pi_{dg(A\Sigma_{yy})} &= \Theta\pi_{yy}, & \pi_{dg(\Sigma_{yy})\mathbb{1}\mathbb{1}^*} &= L\pi_{yy}, \\ \pi_{dg(A\Sigma_{yz})} &= \Theta\pi_{yz}, & \pi_{dg(\Sigma_{yz})\mathbb{1}\mathbb{1}^*} &= L\pi_{yz}, \\ \pi_{dg(A\Sigma_{zz})} &= \Theta\pi_{zz}, & \pi_{dg(\Sigma_{zz})\mathbb{1}\mathbb{1}^*} &= L\pi_{zz}, \end{aligned}$$

and

$$\begin{aligned} \pi_{dg(A\Sigma_{yy})\mathbb{1}\mathbb{1}^*} &= M\pi_{yy}, \\ \pi_{dg(A\Sigma_{yy})\mathbb{1}\mathbb{1}^*} &= M\pi_{yy}, \\ \pi_{dg(A\Sigma_{yy})\mathbb{1}\mathbb{1}^*} &= M\pi_{yy}. \end{aligned}$$

Hence, substituting the above equations in (3.30), and bearing in mind that if the matrices K and T are circulant, then $KT = TK$, we obtain (3.32). □

3.3 SOME SIMULATIONS

In this section we provide some numerical results illustrating the asymmetric synchronization algorithm we propose in this thesis. We run our simulations over three different topologies of network, namely, the complete graph, the circulant graph, and the random geometric graph. In all the simulations we considered a network of $N = 50$ clocks, and we assumed that the transmissions' time instants were generated by N independent Poisson processes of the same intensity $\lambda = 0.1$. For circulant graph we assumed that $\nu = 4$, i. e. every clock can exchange information to only four neighboring clocks. All the random geometric graphs were connected graphs and were generated by choosing the $N = 50$ points uniformly distributed in the unit square and by connecting with an edge each pair of points at distance less

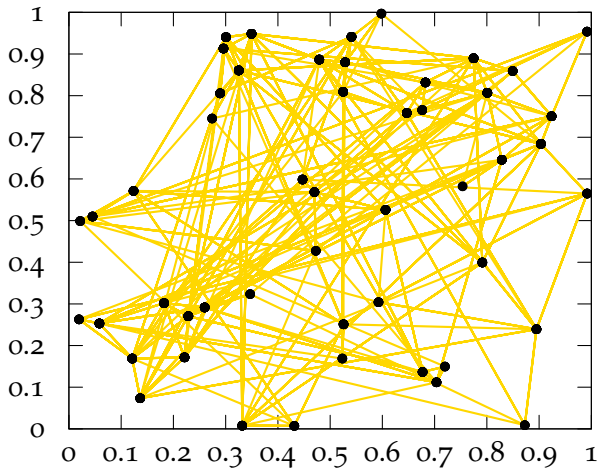

Figure 9: Random geometric graph with $N = 50$ agents. The minimum, maximum and average degree of nodes are respectively of 1, 10 and 5.4.

than 0.15. An example of a geometric graph is illustrated in Figure 9.

We report our results in Figures 10–15. In all the plots we depict the behavior of $\log \|y(k)\|$ obtained by averaging over 100 Monte Carlo runs, randomized with respect to the initial conditions and, as far as the geometric topology is concerned, also with the respect to the graph. In particular, for $i \in \{1, \dots, N\}$, $x'(0)$ has been randomly chosen within the interval $[-1, 1]$ while $\bar{f}_i$ has been randomly chosen within the interval $[1 - \epsilon, 1 + \epsilon]$, where $0 \leqslant \epsilon \leqslant 1$ (the values of ϵ used will be specified in the captions of the figures).

In Figures 10, 11, and 12, we analyzed the behavior of $\log \|y(k)\|$ for different values of α while keeping fixed the value of ϵ to the value 10^{-4}. One can see that the value of α heavily influences the speed of convergence to the synchronization. In particular, small values of α drastically slower down the algorithm.

In Figures 13, 14, and 15 we analyzed the behavior of $\log \|y(k)\|$ for different values of ϵ, while keeping fixed the value of α to the value of $\lambda/8$, $\lambda/50$, and $\lambda/100$ for the complete, circulant, and random geometric graph respectively. It is remarkable that, for all topologies, the algorithm asymptotically achieves the synchronization, even though the drifts $\bar{f}_i$, $i \in \{1, \dots, N\}$, are significantly spread, i.e., $\epsilon = 0.1$. However, as expected, it turns out that the smaller the value of ϵ is, the better the performance of the algorithm are.

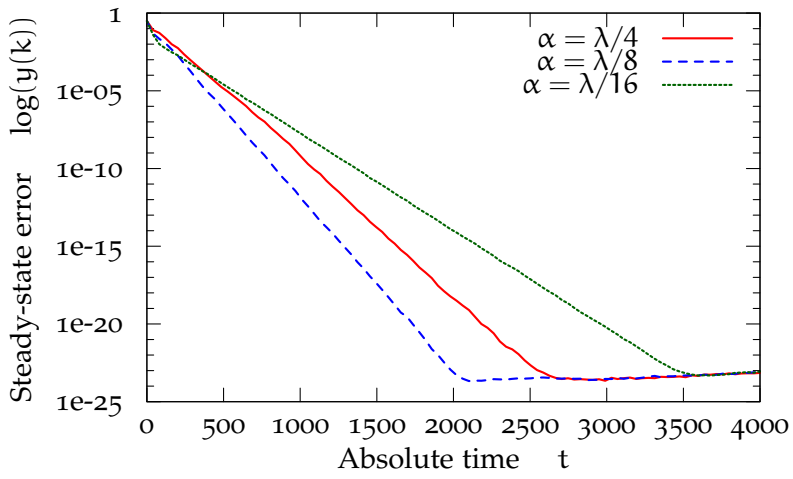

Figure 10: Behavior of the algorithm for the complete graph topology as α changes: $\lambda/4$ (solid line), $\lambda/8$ (dashed line), and $\lambda/16$ (dotted line).

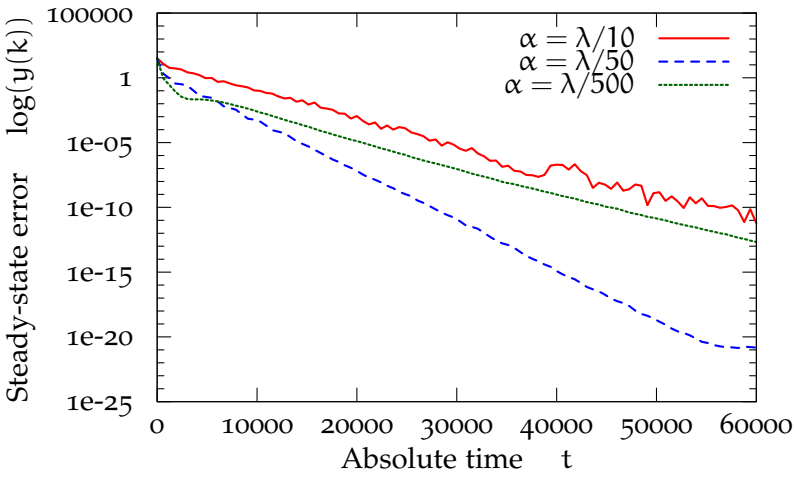

Figure 11: Behavior of the algorithm for the circulant graph topology as α changes: $\lambda/4$ (solid line), $\lambda/8$ (dashed line), and $\lambda/16$ (dotted line).

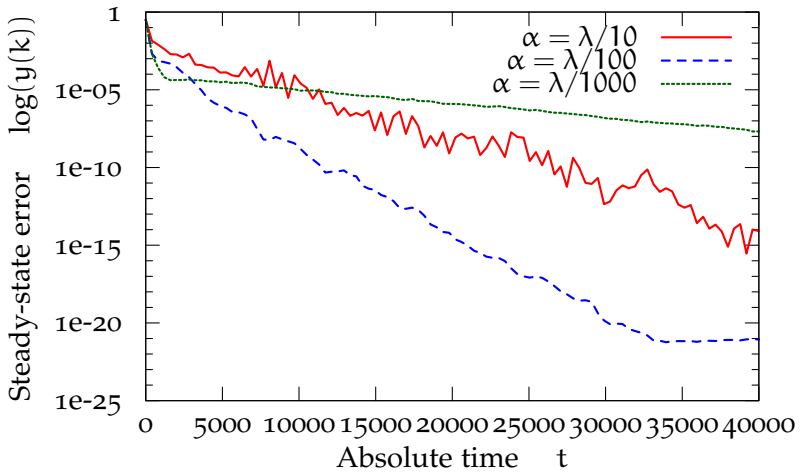

Figure 12: Behavior of the algorithm for the random geometric graph topology as α changes: $\lambda/10$ (solid line), $\lambda/10$ (dashed line), and $\lambda/1000$ (dotted line).

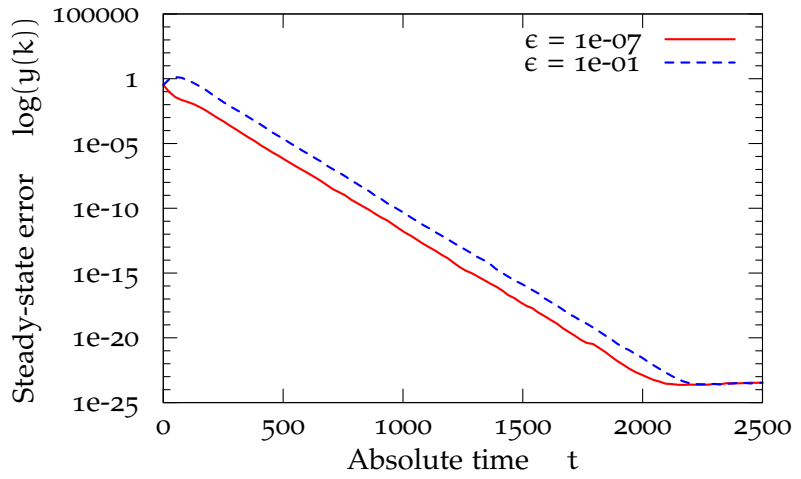

Figure 13: Behavior of the algorithm for the complete graph topology as ϵ changes: 10^{-7} (solid line) and 0.1 (dashed line).

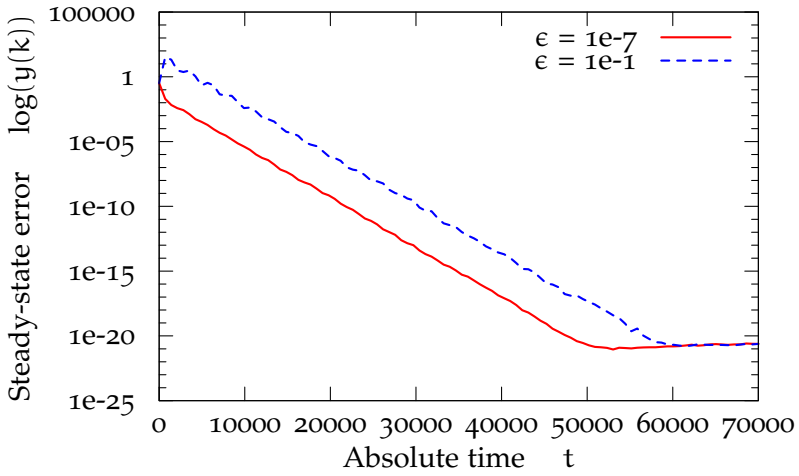

Figure 14: Behavior of the algorithm for the circulant graph topology as ϵ changes: 10^{-7} (solid line) and 0.1 (dashed line).

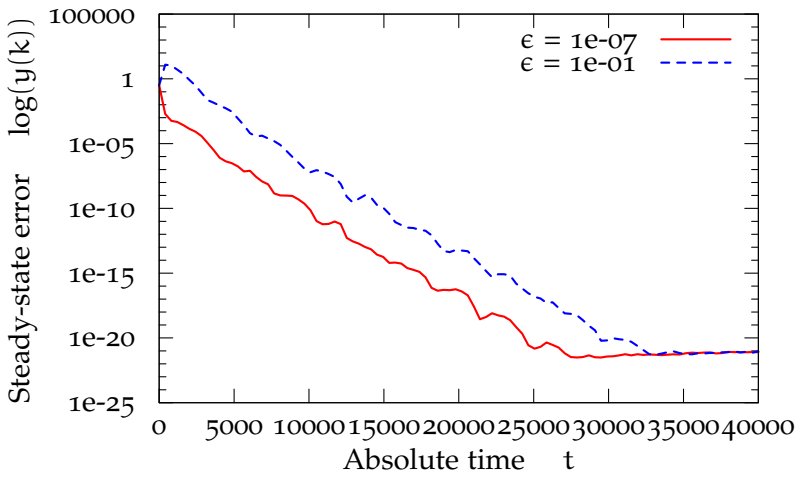

Figure 15: Behavior of the algorithm for the random geometric graph topology as ϵ changes: 10^{-7} (solid line) and 0.1 (dashed line).

4 ALGORITHMS COMPARISON

In this chapter we provide a numerical comparison of our protocol with Average Time-Sync the algorithm introduced by Schenato and Fiorentin [2009]. We focus our attention to performance and accuracy. More precisely, we start considering the ideal noiseless scenario and evaluating the speed of convergence of both algorithms, then we look at the noisy communication channel and we compare the steady-state synchronization error.

4.1 THE AVERAGE TIMESYNCH PROTOCOL

First of all let us summarize the ATS protocol. The algorithm is divided into three main steps: the relative skew estimation, the skew compensation, and the offset compensation.

4.1.1 Relative Skew Estimation

This step of the protocol is concerned with deriving an algorithm to estimate for each clock i the relative skew, i. e. the oscillator period, with respect to its neighbors j. Namely, every node i tries to estimate the relative skew $f_{ij} := f_j/f_i$ with respect to its neighbor nodes j. This is accomplished by transmitting the current local time $x'_j(T_{tx,j}(k))$ to node j which *instantaneously* records its own local time $x'_i(T_{tx,j}(k))$. Therefore, node j records in its memory the pair $(x'_i(T_{tx,j}(k)), x'_j(T_{tx,j}(k)))$. When a new packet from node j arrives to node i, the same procedure is applied to get the new pair $(x'_i(T_{tx,j}(k')), x'_j(T_{tx,j}(k')))$ and the estimate of the relative skew f_{ij} is performed as follows

$$\eta_{ij}^{+} = \rho_\eta \eta_{ij} + (1-\rho_\eta)\frac{x'_j(k') - x'_j(k)}{x'_i(k') - x'_i(k)}, \tag{4.1}$$

where, for the sake of notational simplicity we define

$$x'_l(h) := x'_l(T_{tx,j}(h)),$$

and $\rho_\eta \in (0,1)$ is a tuning parameter. The authors prove that, if there is no measurement error and the oscillator frequencies are constant as stated in Assumption 2.2.1, then the variable η_{ij} converges to $\bar{f}_{ij}$. They also highlight that (4.1) acts a low pass filter

where the parameter ρ_η is used to tune the trade-off between the speed of convergence (ρ_η close to zero) and noise immunity (ρ_η close to unity).

4.1.2 Skew Compensation

This step is the core of the protocol, as it forces all clocks to converge to a common virtual clock rate, α, as defined in (2.8). Basically is a distributed consensus algorithm where any node keeps its own estimate of the global variable α, and it updates its value by averaging it relative to the estimate of its neighbors, i. e.

$$\hat{a}_i^+ = \rho_a \hat{a}_i + (1 - \rho_a)\eta_{ij}\hat{a}_j, \tag{4.2}$$

where $\hat{a}_j$ is the virtual clock skew estimate of $j \in \mathcal{N}_i$.

The authors prove that if $\hat{a}_i(0) = 1$, $\rho_a \in (0, 1)$, and assuming that $\eta_{ij} = f_{ij}$ for all i, j and the underlying communication graph is *strongly connected*, then

$$\lim_{t\to\infty} \hat{a}_i(k) f_i = \alpha, \quad \forall i$$

exponentially fast.

4.1.3 Offset Compensation

At this point it is only necessary to compensate for possible offset errors. Once again, the protocol adopt a consensus algorithm to update the virtual clock offset as follows

$$\hat{o}_i = \hat{o}_i + (1 - \rho_o)(\hat{\tau}_j - \hat{\tau}_i), \tag{4.3}$$

where

$$\hat{\tau}_i = \hat{a}_i x_i' + \hat{o}_i, \quad \text{and}$$
$$\hat{\tau}_j = \hat{a}_j x_j' + \hat{o}_j,$$

are the virtual time estimates of clock i and j respectively and are computed at the *same time* instant. In their work the authors, as done for the skew compensation, prove that

$$\lim_{k\to\infty} \hat{\tau}_i(k) = \tau_j(k) \quad \forall (i, j)$$

exponentially fast.

Remark 4.1.1. The algorithm requires extra memory to record the virtual reference estimate $\hat{\tau}_i$, the relative skew estimates η_{ij} and the last clock pairs $(x_i'(k), x_j'(k))$ for every node i. Moreover, at every transmission instant $T_{tx,i}(k)$, node i must send the three variables x_i', $\hat{a}_i$, and $\hat{\tau}_i$ to its neighbor j.

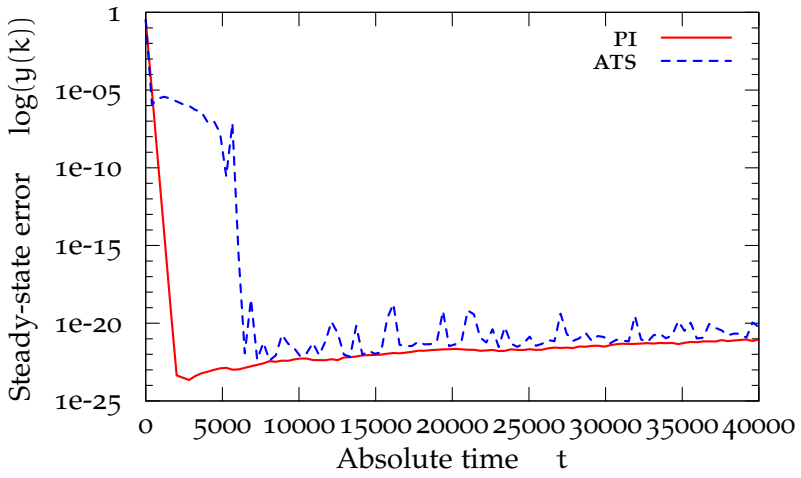

Figure 16: Comparison between the PI protocol and the ATS algorithm for a complete graph topology in a noiseless scenario.

4.2 SOME SIMULATIONS

In this section we provide a numerical comparison between the asymmetric synchronization algorithm we propose in this dissertation and the ATS protocol. As done in the previous chapter, we run our simulation over three different topologies of network, namely, the complete graph, the circulant graph, and the random geometric graph. In all the simulations we considered a network of $N = 50$ units, and we assumed that the transmissions' time instant were generated by N independent Poisson process of the same intensity $\lambda = 0.1$. For circulant graph we assumed that $\nu = 4$, i.e. every clock can exchange information to only four neighboring clocks. All the random geometric graphs were connected graphs and were generated by choosing the $N = 50$ points uniformly distributed in the unit square and by connecting with an edge each pair of points at distance less than 0.15.

We report our results in Figures 16–21. In all the plots we depict the behavior of $\log \|y(k)\|$ obtained by averaging over 100 Monte Carlo runs, randomized with respect to the initial conditions and, as far as the geometric topology is concerned, also with the respect to the graph. In particular, for $i \in \{1, \dots, N\}$, $x'(0)$ has been randomly chosen within the interval $[-1, 1]$ while $\bar{f}_i$ has been randomly chosen within the interval $[1 - \epsilon, 1 + \epsilon]$, where $\epsilon = 10^{-5}$.

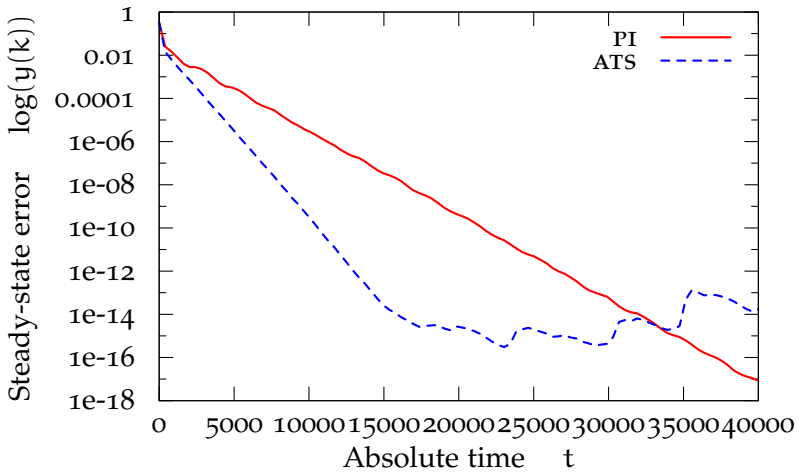

Figure 17: Comparison between the PI protocol and the ATS algorithm for a circulant graph topology in a noiseless scenario.

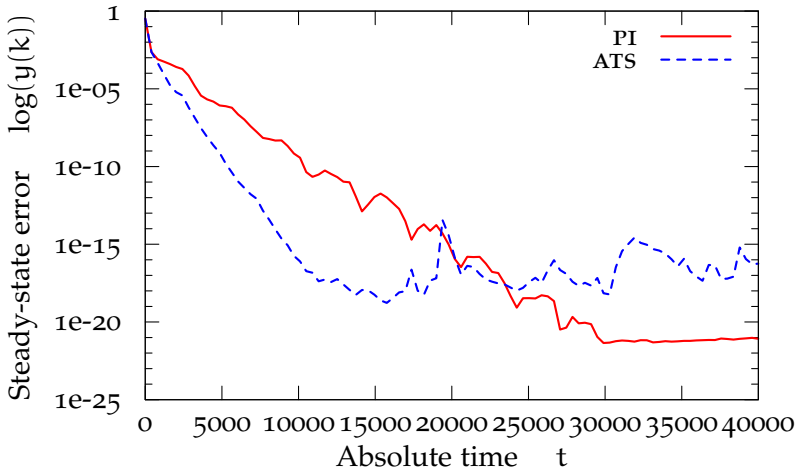

Figure 18: Comparison between the PI protocol and the ATS algorithm for a random geometric graph topology in a noiseless scenario.

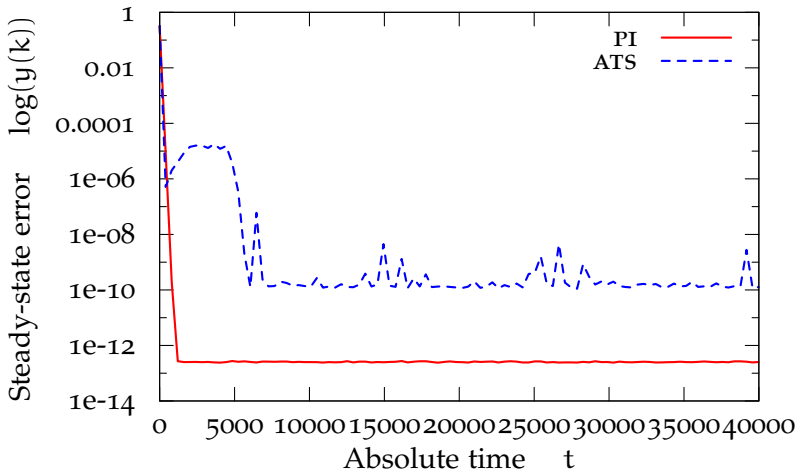

Figure 19: Comparison between the PI protocol and the ATS algorithm for a complete graph topology and a noisy communication channel

In Figures 16, 17, and 18, we compared the behavior of $\log \|y(k)\|$ in the noiseless scenario. The tuning parameters for the ATS protocol were

$$\rho_\eta = 0, \quad \rho_a = 0.5, \quad \text{and} \quad \rho_o = 0.5,$$

while the control parameter α for the asynchronous PI algorithm were

$$\alpha = \lambda/8, \quad \alpha = \lambda/50, \quad \text{and} \quad \alpha = \lambda/100,$$

for the complete, circulant, and complete graph respectively. One can see, except for the complete graph topology, that the ATS protocol is faster than the PI algorithm.

In Figures 19, 20, and 21 we compared the behavior of $\log \|y(k)\|$ in a noisy communication channel. We assumed the noise additive and bounded by the maximum magnitude of 10^{-6}. Specifically, at every transmission instant $T_{tx,i}(k)$ a number uniformly distributed in the interval $[0, 10^{-6}]$ was added to any transmitted variable, i.e. x_i' (for both algorithms), $\hat{a}_i$, and $\hat{\tau}_i$ (only for the ATS protocol). It is remarkable that, for all the topologies, the steady-state of the PI algorithm is decidedly better than the ATS protocol.

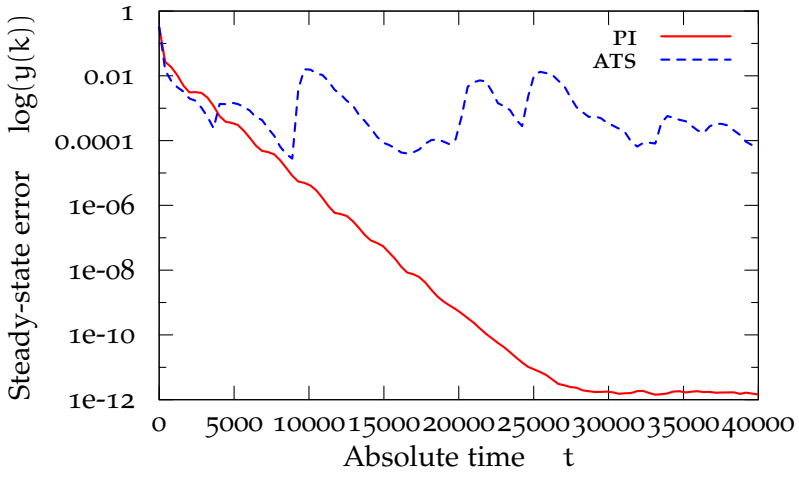

Figure 20: Comparison between the PI protocol and the ATS algorithm for a circulant graph topology and a noisy communication channel

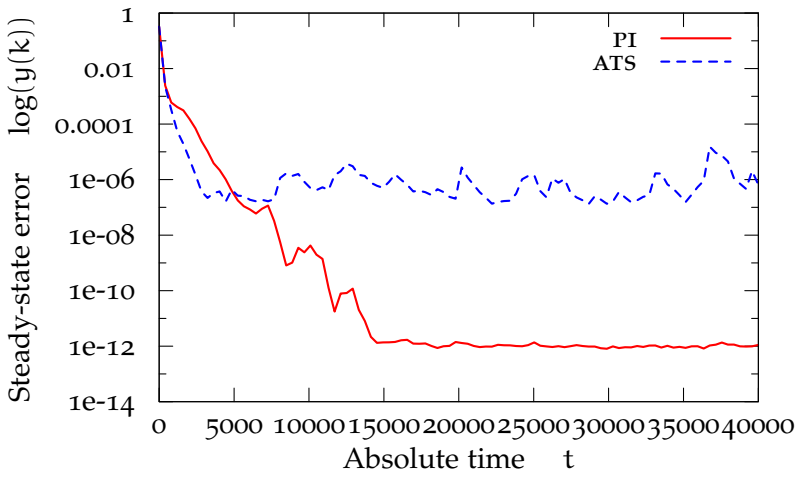

Figure 21: Comparison between the PI protocol and the ATS algorithm for a random geometric graph topology and a noisy communication channel

5 CONCLUSIONS AND FUTURE WORK

We developed a version of the PI algorithm that relies either on a symmetric gossip or on an asymmetric gossip communication scheme to achieve synchronization of a network of clocks. We provided a theoretical stability analysis of the protocol, with respect to the control parameter α, if the underlying graph is the complete graph. In particular we proved that, if the control parameter is under the value of λ (symmetric case) or $\lambda/2$ (asymmetric version), where λ represent the Poisson processes' intensity, then the algorithm scales with the number of nodes. This makes the strategy independent of the network size and easier to implement in a completely distributed fashion. Moreover, we provide some interesting insights if the underlying graph is circulant. Indeed, even thought not supported by theoretical results, we numerically showed that all the eigenvalues responsible for the dynamics of the synchronization error are inside the unit disc. Finally we compared our algorithm to another fully distributed protocol presented in literature showing strength and weaknesses of both strategies.

Future direction of investigation include different communication graphs and protocols (e.g. multicast communication), robustness stability and performance analysis, and modeling some of the most common non-idealities like packet drops, time delivery delays and time-varying speed of the oscillators.

BIBLIOGRAPHY

Bolognani, S., Carli, R. and Zampieri, S. (2009), «A PI consensus controller with gossip communication for clock synchronization in wireless sensors networks», in «Proceedings of the 1st IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys'09)», Venice, Italy.

Boyd, S., Ghosh, A., Prabhakar, B. and Shah, D. (2006), «Randomized gossip algorithms», *IEEE/ACM Transactions on Networking*, vol. 14, pp. 2508–2530.

Carli, R. (2008), *Topics on the Avarage Consensus Problem*, PhD Thesis, Università degli Studi di Padova, Padua, Italy. (Cited on page 15.)

Carli, R. and Zampieri, S. (2010), «Networked clock synchronization based on second order linear consensus algorithms», in «Procedings of the 49th IEEE Conference on Decision and Control», pp. 7259–7264, Atlanta, Georgia, USA. (Cited on pages 2 and 9.)

Carli, R., Chiuso, A., Schenato, L. and Zampieri, S. (2008), «A PI Consensus Controller for Networked Clocks Synchronization», in «Proceedings of the 17th IFAC World Congress», Seoul, South Korea. (Cited on pages 2 and 9.)

Carli, R., Chiuso, A., Schenato, L. and Zampieri, S. (2011), «Optimal synchronization for networks of noisy double integrators», To appear in IEEE Transactions on Automatic Control. (Cited on page 2.)

Elson, J., Girod, L. and Estrin, D. (2002), «Fine-Grained Network Time Synchronization Using Reference Broadcasts», in «Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI'02)», pp. 147–163, Boston, Massachusetts, USA. (Cited on page 1.)

Ganeriwal, S., Kumar, R. and Srivastava, M. B. (2003), «Timing-sync Protocol for Sensor Networks», in «Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys'03)», Los Angeles, California, USA. (Cited on page 1.)

Maróti, M., Kusy, B., Simon, G. and Lédeczi, A. (2004), «The Flooding Time Synchronization Protocol», in «Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys'04)», pp. 39–49, Baltimore, Maryland, USA. (Cited on page 1.)

Olfati-Saber, R., Fax, J. A. and Murray, R. M. (2007), «Consensus and Cooperation in Multi-Agent Networked Systems», *Proceedings of IEEE*, vol. 95 (1), pp. 215–233.

Schenato, L. and Fiorentin, F. (2009), «Average TimeSync: A Consensus-Based Protocol for Time Synchronization in Wireless Sensor Networks», in «Proceedings of the 1st IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys'09)», Venice, Italy. (Cited on pages 2 and 43.)

Simeone, O. and Spagnolini, U. (2007), «Distributed time synchronization in wireless sensor networks with coupled discrete-time oscillators», *EURASIP Journal on Wireless Communication and Networking*, vol. 2007. (Cited on pages 1 and 2.)

Solis, R., Borkar, V. S. and Kumar, P. R. (2006), «A New Distributed Time Synchronization Protocol for Multihop Wireless Networks», in «Proceedings of the 45th IEEE Conference on Decision and Control (CDC'06)», pp. 2734–2739, San Diego, California, USA. (Cited on page 2.)

Werner-Allen, G., Tewari, G., Patel, A., Welsh, M. and Nagpal, R. (2005), «Firefly-Inspired Sensor Network Synchronicity with Realistic Radio Effects», in «ACM Conference on Embedded Networked Sensor Systems (SenSys'05)», San Diego, California, USA. (Cited on pages 1 and 2.)

COLOPHON

This thesis was typeset with LaTeX 2ε on a GNU/Linux system using a different view of the ClassicThesis package by André Miede. The typographic style was inspired by Robert Bringhurst as presented in *The Elements of Typographic Style*. The look edition of this dissertation is due to the excellent work of Lorenzo Pantieri in his ArsClassica package.

NOTE: The text font belong to the Hermann Zapf's *Palatino* type face family (Type 1 PostScript fonts *URW Palladio L*). The mathematical equation was typeset with the *AMS Euler* font by Hermann Zapf and Donald E. Knuth. The sans serif type faces are the *Classico* (a Zapf's *Optima* typeface clone) by Bob Tennet (Type 1 PostScript fonts *URW Classico*).

Final version: April 2, 2011.

DECLARATION

I declare this work research is mine with the exception of what explicitly mentioned in the text.

I also declare the material within the dissertation could be published, wholly or partially, by my supervisor Prof. Sandro Zampieri and/or my assistant supervisor Dr. Ruggero Carli.

Padova, marzo 2011

Edoardo D'Elia