

RESEARCH OUTPUTS / RÉSULTATS DE RECHERCHE

Generating directed networks with prescribed Laplacian spectra

Nicoletti, Sara; Carletti, Timoteo; Fanelli, Duccio; Battistelli, Giorgio; Chisci, Luigi

Published in:

Journal of Physics: Complexity

DOI:

<https://doi.org/10.1088/2632-072X/abbd35>

Publication date:

2020

Document Version

Peer reviewed version

[Link to publication](#)

Citation for published version (HARVARD):

Nicoletti, S, Carletti, T, Fanelli, D, Battistelli, G & Chisci, L 2020, 'Generating directed networks with prescribed Laplacian spectra', *Journal of Physics: Complexity*, vol. 2, no. 1, pp. 015004. <https://doi.org/10.1088/2632-072X/abbd35>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

ACCEPTED MANUSCRIPT • OPEN ACCESS

Generating directed networks with prescribed Laplacian spectra

To cite this article before publication: Sara Nicoletti *et al* 2020 *J. Phys. Complex.* in press <https://doi.org/10.1088/2632-072X/abbd35>

Manuscript version: Accepted Manuscript

Accepted Manuscript is “the version of the article accepted for publication including all changes made as a result of the peer review process, and which may also include the addition to the article by IOP Publishing of a header, an article ID, a cover sheet and/or an ‘Accepted Manuscript’ watermark, but excluding any other editing, typesetting or other changes made by IOP Publishing and/or its licensors”

This Accepted Manuscript is © 2020 The Author(s). Published by IOP Publishing Ltd.

As the Version of Record of this article is going to be / has been published on a gold open access basis under a CC BY 3.0 licence, this Accepted Manuscript is available for reuse under a CC BY 3.0 licence immediately.

Everyone is permitted to use all or part of the original content in this article, provided that they adhere to all the terms of the licence <https://creativecommons.org/licenses/by/3.0>

Although reasonable endeavours have been taken to obtain all necessary permissions from third parties to include their copyrighted content within this article, their full citation and copyright line may not be present in this Accepted Manuscript version. Before using any content from this article, please refer to the Version of Record on IOPscience once published for full citation and copyright details, as permissions may be required. All third party content is fully copyright protected and is not published on a gold open access basis under a CC BY licence, unless that is specifically stated in the figure caption in the Version of Record.

View the [article online](#) for updates and enhancements.

Generating directed networks with prescribed Laplacian spectra

Sara Nicoletti

*Dipartimento di Ingegneria dell'Informazione,
Università di Firenze, Via S. Marta 3, 50139 Florence, Italy and
Università degli Studi di Firenze, Dipartimento di Fisica e Astronomia,
CSDC and INFN, via G. Sansone 1, 50019 Sesto Fiorentino, Italy*

Timoteo Carletti

*naXys, Namur Institute for Complex Systems, University of Namur,
8 Rempart de la Vierge, B5000 Namur, Belgium*

Duccio Fanelli

*Università degli Studi di Firenze, Dipartimento di Fisica e Astronomia,
CSDC and INFN, via G. Sansone 1, 50019 Sesto Fiorentino, Italy*

Giorgio Battistelli and Luigi Chisci

*Dipartimento di Ingegneria dell'Informazione,
Università di Firenze, Via S. Marta 3, 50139 Florence, Italy*

Complex real-world phenomena are often modeled as dynamical systems on networks. In many cases of interest, the spectrum of the underlying graph Laplacian sets the system stability and ultimately shapes the matter or information flow. This motivates devising suitable strategies, with rigorous mathematical foundation, to generate Laplacians that possess prescribed spectra. In this paper, we show that a weighted Laplacian can be constructed so as to *exactly* realize a desired *complex* spectrum. The method configures as a non trivial generalization of existing recipes which assume the spectra to be real. Applications of the proposed technique to (i) a network of Stuart-Landau oscillators and (ii) to the Kuramoto model are discussed. Synchronization can be enforced by assuming a properly engineered, signed and weighted, adjacency matrix to rule the pattern of pairing interactions.

I. INTRODUCTION

Complex networks play a role of paramount importance for a wide range of problems, of cross-disciplinary breadth. In several cases of interest, networks define the skeleton of pairwise interaction between coupled populations, families of homogeneous constituents anchored to a given node of the collections. The nature of existing paired relationships between mutually entangled populations is encoded in the weight of the links, that bridge adjacent nodes. Cooperative and competitive interactions can, in principle, be accommodated for by allowing for the weights to take positive or negative values. Local and remote non linear couplings shape the ensuing dynamics, and possibly steer the system towards a stationary stable equilibrium which is compatible with the assigned initial condition. The stability of the fixed points can be analyzed by studying the dynamical system in its linearized version. For reaction-diffusion systems defined on networks, the stability of the inspected equilibrium is ultimately dictated by the spectrum of the discrete Laplacian matrix [1–3]. The eigenvalues of the Laplacian define, in fact, the support of the dispersion relation, the curve that sets the rate for the exponential growth of the imposed perturbation. More specifically, external disturbances can be, in general, decomposed on the basis formed by the eigenvectors of the Laplacian operator. Each eigenvector defines an independent mode, which senses the web of intricate paths made accessible across the network: the perturbation can eventually develop, or, alternatively, fade away, along the selected direction, depending on the corresponding entry of the dispersion relation, as fixed by its associated eigenvalues. Stability is an attribute of paramount importance as it relates to resilience, the ability of a given system to oppose to external perturbations that would take it away from the existing equilibrium. Similarly, synchronization, a widespread phenomenon in distributed systems, can be enforced by properly adjusting the spectrum of the matrix which encodes for intertwined pairings. Based on the above, it is therefore essential to devise suitably tailored recipes for generating networks, which display a prescribed Laplacian spectrum, compatible with the stability constraint [4, 5].

The problem of recovering a network from a set of assigned eigenvalues has been tackled in the literature both from an algorithmic [6–8] and formal [9–11] standpoints. In [10], a procedure is discussed to generate an undirected and weighted graph from its spectrum. The result extends beyond the well-known theorem of Botti and Merris [12] which states that the reconstruction of non-weighted graphs is, in general, impossible since almost all (non-weighted) trees share their spectrum with another non-isomorphic tree. In [13], a method is proposed to obtain a, directed or undirected, graph whose eigenvalues are constrained to match specific bounds, which ultimately reflect the nodes degrees, as well as the associated weights. In [14], a mathematically rigorous strategy is instead developed to yield weighted graphs which *exactly* realize any desired spectrum. As discussed in [14], the method translates into an efficient approach to control the dynamics of various archetypal physical systems via suitably designed Laplacian spectra. The results are however limited to undirected Laplacians, characterized by a real spectrum. The purpose of this paper is to expand beyond these lines, by proposing and testing a procedure which enables one to recover a signed Laplacian operator which displays a prescribed complex spectrum. Signed Laplacians are often used in the literature for applications which relate to social contagion, cluster synchronization or repulsive-attractive interactions [19, 20]. In engineering, they are often employed in modeling microgrids dynamics [21].

The paper is organized as follows. The first section is devoted to illustrating the devised method, focusing on the mathematical aspects. We then turn to discussing the implementation of the scheme and introducing the sparsification algorithms that are run to cut unessential links. In the subsequent section, we elaborate on the conditions that are to be met to generate a positively weighed network. This discussion is carried out with reference to a specific setting. Then, we apply the newly introduced technique to the study of an ensemble made of coupled Stuart-Landau oscillators [22–24] and to (a simplified version of) the Kuramoto model [27, 28]. Finally, in the last section, we sum up the contributions and provide concluding remarks.

II. A RECIPE TO OBTAIN A LAPLACIAN WITH ASSIGNED COMPLEX EIGENVALUES.

Consider a network made of Ω nodes and denote by A the (weighted) adjacency matrix, where structural information is encoded. More precisely, the element A_{ij} is different from zero when a directed link exists from j to i . The entries of the matrix A are real numbers and their signs reflect the specificity of the interaction at play: negative signs stand for inhibitory (or antagonistic) couplings, while positive entries point to excitatory (or cooperative) interaction. From the adjacency matrix, one can define its associated Laplacian operator. This is the matrix L , whose elements are $L_{ij} = A_{ij} - k_i \delta_{ij}$, where $k_i = \sum_j A_{ij}$ represents the natural extension of the concept of (incoming) connectivity to the case of a weighted network and δ_{ij} denotes the Kronecker delta.

We shall here discuss a procedure to generate a Laplacian matrix, which displays a prescribed set of eigenvalues. As anticipated above, we will focus in particular on directed Laplacians, which yields, in general, complex spectra. Concretely, we begin by introducing a collection of $\Omega = 2N + 1$ complex quantities defined as ¹

$$\{\Lambda_i\} = \{\Lambda_1, \Lambda_2, \dots, \Lambda_{2N}, \Lambda_{2N+1} = 0\} \quad (1)$$

The first $2N$ elements come in complex conjugate pairs and we set in particular $\Lambda_i = \Lambda_{i+N}^*$, $\forall i = 1, \dots, N$, where $(\cdot)^*$ stands for complex conjugate. The aim of this section is to develop a rigorous procedure to construct a directed (and weighted) graph G with $2N + 1$ vertices, whose associated Laplacian has $\{\Lambda_i\}$ for eigenvalues. Recall that $\{\Lambda_i\}$ contains the null element, since this latter is, by definition, a Laplacian's eigenvalue.

The procedure that we are going to detail in what follows exploits the eigenvalue decomposition of the Laplacian matrix. To this end we will seek to introduce a proper eigenvector

¹ We shall assume all the eigenvalues but 0 to be complex numbers. Let us remark that the method developed readily adapts to the case where also real eigenvalues are present. In this case, the eigenvectors associated to real eigenvalues are generated according to the prescriptions of [14]. Eigenvectors linked to complex eigenvalues are instead assigned following the procedure outlined here.

basis such that

$$L = VDV^{-1} \quad (2)$$

is a Laplacian. In Eq. (2), D is a diagonal matrix where the sought Laplacian eigenvalues are stored. More specifically, $D_{ii} = \Lambda_{i-1}$ for $i = 2, \dots, 2N + 1$ and $D_{11} = \Lambda_{2N+1} = 0$. The problem is hence traced back to constructing V , whose columns are the right eigenvectors of L . We also recall that rows of the inverse matrix V^{-1} are the left eigenvectors of L . As outlined in [15], Laplacian (right and left) eigenvectors should satisfy a set of conditions:

1. The columns of V , which refer to complex conjugate eigenvalues, must be complex conjugate too.
2. The same condition holds for the rows of V^{-1} .
3. Moreover, the columns of V (resp. the rows of V^{-1}) corresponding to eigenvalues different from 0 should sum up to zero.
4. Finally, the right eigenvector relative to the null eigenvalue should be uniform (i.e. display identical components).

In light of the above, we put forward for V the following structure:

$$V = \begin{pmatrix} c & v^T & v^{T*} \\ cu & iU & -iU \\ cu & U & U \end{pmatrix} \quad (3)$$

where c is an arbitrary constant, i stands for the imaginary unit, U is an invertible $N \times N$ matrix having real entries. Further, vector $u = (1 \dots 1)^T$ has dimension $N \times 1$ and vector v is defined as

$$v^T = -(1 + i)u^T U \quad (4)$$

The first column of V is hence a uniform vector, corresponding to the eigenvector associated to the null eigenvalue. By construction, every other column sums up to zero, that is (4) holds. In the following, we will write $D_{jj} = \alpha_j + i\beta_j$, which, in turn, implies $D_{j+N,j+N} = \alpha_j - i\beta_j$, for $j = 2, \dots, N+1$. Here, α_j and β_j are real quantities and respectively denote the real and imaginary parts of the j -th eigenvalue. To proceed further, one needs to determine the inverse of V .

To achieve this goal we begin by considering a generic matrix W , which satisfies the general constraints that are in place for V^{-1} . In formulae:

$$W = \begin{pmatrix} d & du^T & du^T \\ (1-i)w & S & -iS^* \\ (1+i)w^* & S^* & iS \end{pmatrix} \quad (5)$$

where

$$w = -\frac{1}{1-i}(Su - iS^*u) \quad (6)$$

Note that the j th and $(N+j)$ th rows of W , for $j = 1, \dots, N+1$, are complex conjugated, as required. Moreover, summing all the elements of each row (but the first) yields zero, a condition that the inverse of V should meet, as anticipated above. Building on these premises, we shall here determine the unknown S , w and d so as to match the identity $WV = I$, where I stands for the $(2N+1) \times (2N+1)$ identity matrix. This implies, in turn, that $W \equiv V^{-1}$ due to the uniqueness of the inverse matrix.

A straightforward manipulation yields the following conditions for, respectively, d and w

$$\begin{cases} d = \frac{1}{c(2N+1)} \\ wv^T = i(S - iS^*)EU \\ wv^{T*} = (S - iS^*)EU \end{cases} \quad (7)$$

where use has been made of the identity $u^T u = N$ and where

$$E = uu^T = \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{pmatrix} \quad (8)$$

The quantity d is completely specified by the first of Eqs. (7) and solely depends on c and N , the size of the system. By making use of the identities (4) and (6), one can progress in the analysis of the second and third conditions (7) to eventually get:

$$SB + S^*A = I \quad (9)$$

$$SA - S^*B = 0 \quad (10)$$

where:

$$A = EU - iEU - iU \quad (11)$$

$$B = iEU + EU + iU \quad (12)$$

It is, therefore, immediate to conclude that

$$S = (B + AB^{-1}A)^{-1} \quad (13)$$

$$S^* = (B + AB^{-1}A)^{-1}AB^{-1} \quad (14)$$

The analysis can be pushed further to relate S to matrices U and E . The calculation, detailed in Appendix A, yields

$$S = -\frac{i}{2}U^{-1} \left[I - \frac{1+i}{1+2N}E \right] \quad (15)$$

The expression for S^* can be immediately obtained by taking the complex conjugate of the above equation. In conclusion, the matrix W defined in (5) is the inverse matrix of V , provided that d and S are respectively assigned as specified above.

Clearly, matrix L defined in (2) has the desired spectrum (1). We should, however, prove that L is a Laplacian. This amounts to showing that L is a real matrix, whose columns sum up to zero. The proof is given hereafter.

Proposition. Matrix L is real.

From Eqs. (3) and (5), one can readily compute the elements of L via matrix products and, taking advantage of the block structures of V and W , prove that L is real. $\Re(\cdot)$ is introduced to represent the real part of (\cdot) . The generic element L_{st} can be written as:

$$L_{st} = (VDW)_{st} = \sum_{k=1}^{2N+1} V_{sk}(DW)_{kt} = \sum_{k=1}^{2N+1} V_{sk} \sum_{k'=1}^{2N+1} D_{kk'}W_{k't} \quad (16)$$

$$= \sum_{k=2}^{2N+1} V_{sk}D_{kk}W_{kt} \quad (17)$$

due to the diagonal structure of the matrix D and recalling that $D_{11} = 0$. By making use of the specific form of V and W , one gets:

$$L_{st} = \sum_{k=2}^{N+1} V_{sk}D_{kk}W_{kt} + \sum_{k=N+2}^{2N+1} V_{sk}D_{kk}W_{kt} \quad (18)$$

$$= \sum_{k=2}^{N+1} V_{sk}D_{kk}W_{kt} + \sum_{k=2}^{N+1} (V_{sk}D_{kk}W_{kt})^* \quad (19)$$

$$= \sum_{k=2}^{N+1} 2\Re(V_{sk}D_{kk}W_{kt}) \quad (20)$$

since $\alpha\beta + \alpha^*\beta^* = 2\Re(\alpha\beta)$, for any complex numbers α and β . One can thus conclude that L , as generated by the above procedure, is real.

Proposition. Each column of L sums up to zero.

Because of the diagonal structure of D :

$$\sum_i L_{ij} = \sum_{il} V_{il}D_{ll}V_{lj}^{-1} \quad (21)$$

Then:

$$\sum_i L_{ij} = \sum_l V_{lj}^{-1} D_{ll} \sum_i V_{il} = 0 \quad (22)$$

since (i) $D_{11} = 0$ and (ii) the components of all eigenvectors corresponding to non-null eigenvalues, sum up to zero. Notice that this result can also be proven by observing that the uniform vector $d\mathbf{1}$ is the left eigenvector corresponding to the null eigenvalue, that is

$$d\mathbf{1}L = 0 \quad (23)$$

From (23), it follows that $\sum_i L_{ij} = 0$, for every j .

Proposition. L is balanced.

We can also show that $\sum_j L_{ij} = 0$ i.e. that the sum of all the elements of any given row i returns zero. According to (2), the first column of V is the right eigenvector corresponding to eigenvalue 0, namely

$$Lc\mathbf{1} = cL\mathbf{1} = 0 \quad (24)$$

This implies, in turn, $\sum_j L_{ij} = 0 \forall i$, which ends the proof. The Laplacian is hence balanced, as the sums on the rows and on the (corresponding) columns return the same result.

From the generated Laplacian operator, one can readily calculate the adjacency matrix of the underlying network. In general, for any assigned spectrum, the recovered adjacency matrix is fully connected, meaning that there exist links connecting each pair of nodes. Notice that links are weighted and signed. The weights can be small or have a modest impact on the spectrum of the associated Laplacian. This motivates the implementation of a dedicated sparsification procedure, which seeks to remove unessential links, in terms of their reflection on the ensuing Laplacian spectrum. The next section is devoted to elaborating along these lines.

III. EXAMPLES AND SPARSIFICATION.

In this section we discuss a sparsification procedure, which aims at a posteriori simplifying the structure of the recovered network. To this end, we begin by generating a network following the strategy outlined in the preceding section, and which yields an assigned spectrum for the associated Laplacian. The Laplacian spectrum that we seek to recover consists of $\Omega = 2N + 1$ complex entries, the eigenvalues, which are here confined in the left portion of the complex plane, by setting $\Re(\Lambda_j) = \alpha_j < 0$ for $j \geq 2$, see blue crosses in Fig. 1 (a). This choice is somehow arbitrary, and ultimately amounts to enforce stability into a linear system of the form:

$$\frac{dx_i}{dt} = \sum_j L_{ij} x_j \quad (25)$$

where x_i is the i -th entry of the Ω -dimensional state vector x . In the final part of the paper, we will turn to considering more complex scenarios where the stability of the examined dynamics is also influenced by local reaction terms.

The network that we obtain, following the scheme outlined in the preceding sections yielding a Laplacian with the prescribed spectrum, is in general fully connected. In other words, a weighted link exists between any pair of nodes. The weights of the link can be, in principle, very small and, as such, bear a modest imprint on the ensuing Laplacian spectrum. Motivated by this observation, we perform an a posteriori sparsification of the obtained network: this aims to identifying and then removing the finite subset of links that appear to have a modest impact on the eigenvalues of the associated Laplacian.

The first sparsification procedure that we have considered, aims at removing unessential links while confining the spectrum of the Laplacian operator within a bounded region of the complex plane. More precisely, we focus on the links which display a weight in the range $(-\sigma, \sigma)$, where σ is a small, arbitrarily chosen, cut off. All links whose weight is smaller than σ in absolute value are selected, in a random order. The selected link is removed and the modified

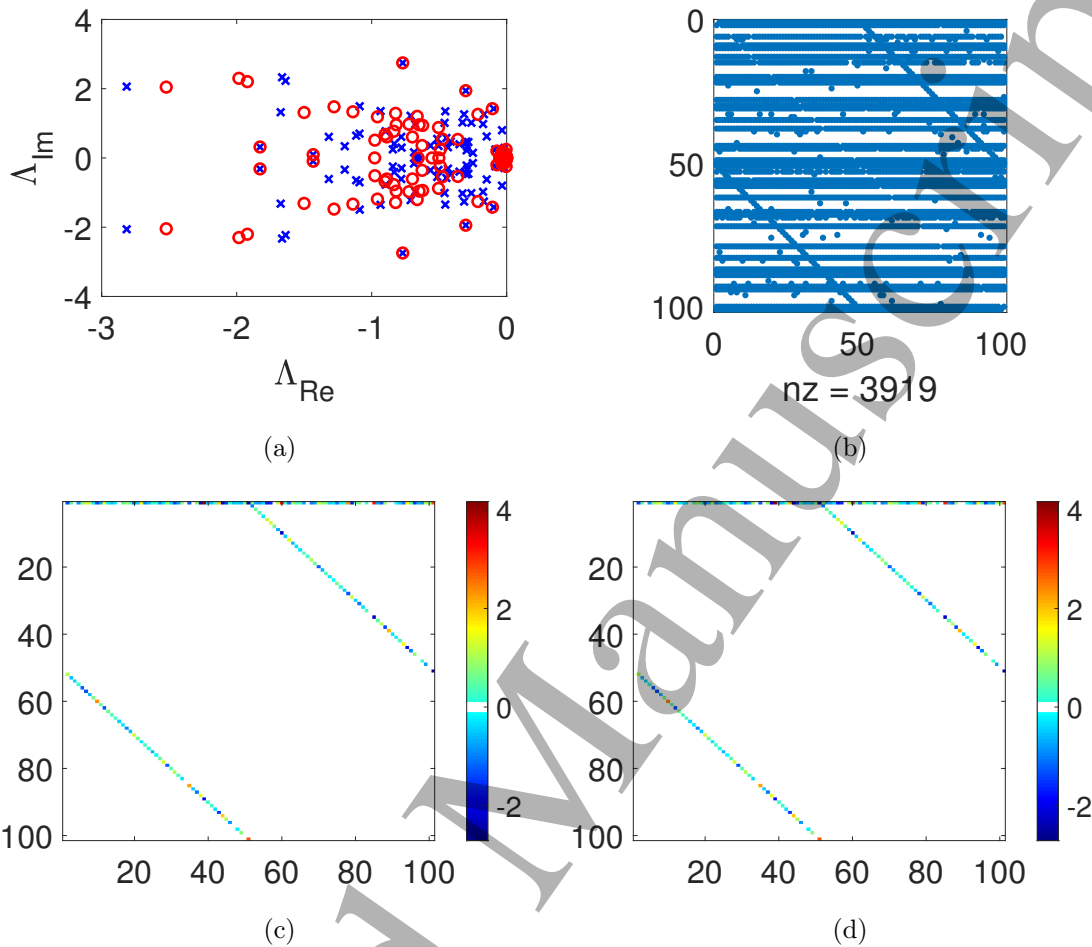


Figure 1. The recipe discussed in the main text is applied to generate a network of $\Omega = 2N + 1 = 101$ nodes, whose associated Laplacian displays the spectrum depicted (blue stars) in panel (a). The spectrum of the Laplacian obtained from the sparsified network is shown with red circles. We here follow the first sparsification recipe as illustrated in the main body of the paper. More specifically, we trim unessential links, chosen among those that bear very modest weights, while confining the Laplacian spectrum in a bounded domain, located in the negative portion of the complex plane. Here, σ is 0.01 and $\delta = 0.5$. In panel (b), the sparsity pattern of the adjacency matrix obtained upon application of the sparsification algorithm is shown. The entries of the adjacency matrices, before and after the sparsification are respectively plotted, with an appropriate color-code, in panels (c) and (d). The main structure of the network is preserved upon application of the devised sparsification protocol.

Laplacian spectrum computed. Denote by $\tilde{\Lambda}_j$, for $j = 2, \dots, 2N + 1$, the Laplacian eigenvalues obtained upon removal of the link. The change to the network arrangement becomes permanent, if $|\min_j [\Re(\tilde{\Lambda}_j)] - \min_j [\Re(\Lambda_j)]| < \delta$ and $|\max_j [\Im(\tilde{\Lambda}_j)] - \max_j [\Im(\Lambda_j)]| < \delta$, for $j = 2, \dots, N$. Here $\Im(\cdot)$ stands for the imaginary part of (\cdot) and δ is an arbitrary threshold which quantifies the amount of perturbation that is deemed acceptable for the problem at hand. As a further condition, we check that $\Re(\tilde{\Lambda}_j) < 0$ for $j = 2, \dots, 2N + 1$, which, in turn, corresponds to preserving the stability of the linear system (25). Clearly, the order of extraction of the links, which are candidate to be trimmed, matters. Different realizations of the procedure of progressive sparsification illustrated above might hence result in distinct final outcomes. In Fig. 1(a), the eigenvalues obtained after the sparsification algorithm are plotted (red circles) for one choice of the cutoff δ . The sparsity pattern of the adjacency matrix obtained at the end of the above procedure is displayed in panel (b) of Fig. 1. In panels (c) and (d) of Fig. 1 we plot, with an appropriate color code, the entries of the adjacency matrices, before and after the sparsification. Only weights which are significantly different from zero (see annexed colorbars) are displayed. As appreciated by visual inspection, the skeleton of the network is not altered by the applied sparsification. To monitor how the eigenvalues get redistributed within the bounded domain to which they belong, we introduce the following indicators:

$$I_x = \sum_{i=2}^{2N+1} (\beta_i)^2 \quad (26)$$

$$I_y = \sum_{i=2}^{2N+1} \left(\alpha_i - \frac{1}{2N+1} \sum_{j=2}^{2N+1} \alpha_j \right)^2 \quad (27)$$

The quantity I_x measures the dispersion along the imaginary axis, by weighting the squared distance of each eigenvalue from the horizontal axis. Conversely, I_y reflects the scattering of the eigenvalues about their mean, in the direction of the real axis. In Fig. 2, I_x and I_y , normalized to their respective values obtained before application of the sparsifi-

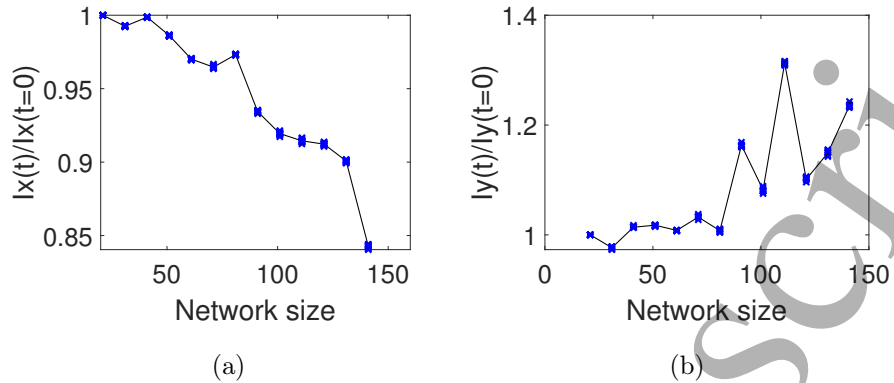


Figure 2. Panel (a): I_x as measured at the end of the sparsification and normalized to the corresponding value, before the sparsification is plotted versus N , the size of the explored network. Panel (b): I_y , calculated after the sparsification and normalized to the corresponding value, before the sparsification is depicted versus N . In both cases, blue symbols are computed, as the average over 15 different realizations of the generated network (with the same given spectrum). For each value of N , the real and imaginary components of the eigenvalues are random number, drawn from a normal distribution. The solid line is a guide for the eye. Here, $\delta = 0.5$ and $\sigma = 0.01$.

cation algorithm, are shown versus N , an indicator of the size of the generated networks. The sparsification procedure shrinks the eigenvalues in the x -direction, while the opposite tendency is observed for the distribution along the y -direction.

The second sparsification method implements a more stringent constraint. Just like before, we select the links with weights in the range $(-\sigma, \sigma)$, where σ acts as a small threshold amount. Unlike the former case, we now eliminate the selected link only if the change produced on the modulus of *each* of the N eigenvalues is smaller than δ , namely if $|\Lambda_j - \tilde{\Lambda}_j| < \delta$, for $j = 2, \dots, 2N+1$. In Fig. 3, the eigenvalues obtained after the sparsification algorithm are plotted (red circles) for two choices of the cutoff δ . The number of links that can be effectively removed grows with δ , the size of the allowed perturbation, as clearly demonstrated in Fig. 4.

As mentioned above, this latter method implements a local, hence stringent, constraint. By allowing modest re-arrangements of individual eigenvalues yields a spectral distribution

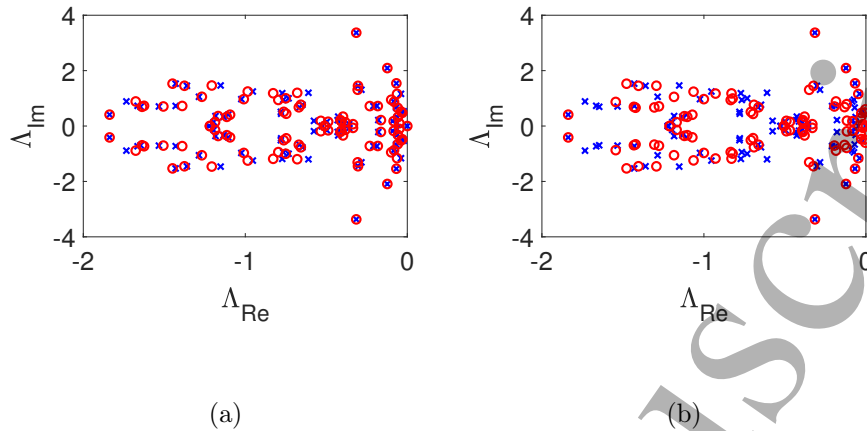


Figure 3. *Effect of the second method of sparsification on a network made of $\Omega = 2N + 1 = 101$ nodes. The original spectrum is plotted with (blue) stars. The modified one with (red) circles. Panel (a) refers to $\delta = 0.07$ while panel (b) to $\delta = 0.2$. Here, $\sigma = 0.01$.*

in the complex plane which is by construction close to the original one. This method is better suited when the room for maneuver is limited, due to the specific constraints that the reactive model implements. On the other hand the number of pruned links can be scanty, when small local perturbations are solely allowed for. A global re-organization of the operator's spectrum, which entails a significant reduction in the associated number of active links, might instead be accomplished when the eigenvalues are constrained to fall within a compact domain in the complex plane, as follows the former sparsification procedure.

Summing up, we have developed and tested a procedure to generate a network which returns an associated Laplacian matrix with a prescribed complex spectrum. The weighted network obtained following the above procedure is, in general, fully connected. Dedicated sparsification strategies can be applied to remove the links which carry a small weight, and bear a modest imprint on the ensuing Laplacian spectrum. In the following, we will consider a specific setting of the aforementioned generation scheme, which makes it possible for the Laplacian elements to be computed analytically.

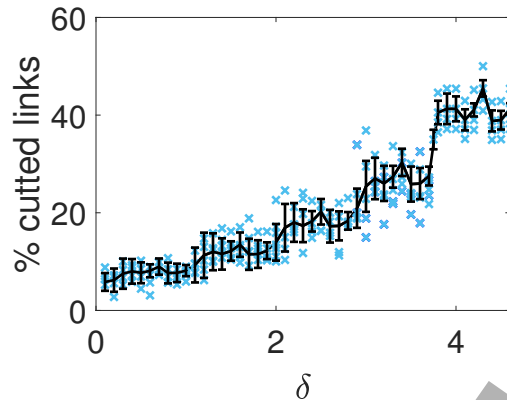


Figure 4. The percentage of links that are cut against the allowed perturbation δ . Blue symbols refer to 5 independent realizations, for each value of δ . The black line goes through the average values, computed from the collection of independent runs, at fixed δ . Here, $N = 50$ ($\Omega = 2N + 1 = 101$) and $\sigma = 0.01$.

IV. FOCUSING ON THE SPECIAL CASE $U = qI$

In the previous sections we described a general method to generate a Laplacian matrix which displays a designated spectrum. The method assumes a generic matrix U , which can be randomly assigned. In the following, we will focus on the specific case where U is proportional to the identity matrix and progress with the analytic characterization of the obtained Laplacian. As we shall argue in the following, working in this framework allows us to derive a set of closed conditions for constraining the weights of the underlying network to strictly positive values. To proceed in this direction we set:

$$U = qI \quad (28)$$

where I stands for the identity matrix and q is scalar.

A straightforward calculation returns the following expression for matrix S :

$$S = \begin{pmatrix} a + ib & a - ia & \dots & a - ia \\ a - ia & a + ib & \ddots & a - ia \\ \dots & \ddots & \ddots & \vdots \\ a - ia & \ddots & \ddots & a + ib \end{pmatrix} \quad (29)$$

while w is a uniform vector with identical entries equal to $Na + ((N - 1)a - b)i$ and the quantities d, a, b are specified by

$$d = \frac{1}{(2N + 1)c} \quad (30)$$

$$a = \frac{-1}{2(2N + 1)q} \quad (31)$$

$$b = 2Na \quad (32)$$

From equations (16), (18) one can obtain a closed expression for each element of the Laplacian, as function of the eigenvalues. The interested reader can find the detailed computations in Appendix B. In the following the final formulae are reported.

The diagonal elements satisfy:

$$L_{11} = \frac{2}{2N + 1} \sum_{k=2}^{N+1} \alpha_k \quad (33)$$

$$L_{ss} = \frac{2N\alpha_s + \beta_s}{2N + 1} \quad s = 2, \dots, N + 1 \quad (34)$$

$$L_{ss} = \frac{2N\alpha_s - \beta_s}{2N + 1} \quad s = N + 2, \dots, 2N + 1, \quad (35)$$

where use has been made of the identity $\Re(D_{kk}) = \alpha_k$. The first row and column are given

by:

$$L_{1t} = \frac{1-2N}{1+2N}\alpha_t - \beta_t + \frac{2}{2N+1} \sum_{k=2, k \neq t}^{N+1} \alpha_k \quad t = 2, \dots, N+1 \quad (36)$$

$$L_{1t} = \frac{1-2N}{1+2N}\alpha_t + \beta_t + \frac{2}{2N+1} \sum_{k=2, k \neq t-N}^{N+1} \alpha_k \quad t = N+2, \dots, 2N+1 \quad (37)$$

$$L_{t1} = \frac{-\alpha_t + \beta_t}{2N+1} \quad t = 2, \dots, N+1 \quad (38)$$

$$L_{t1} = \frac{-\alpha_t - \beta_t}{2N+1} \quad t = N+2, \dots, 2N+1. \quad (39)$$

while the remaining elements are:

$$L_{st} = \frac{-\alpha_s + \beta_s}{2N+1} \quad s = 2, \dots, N+1 \text{ and } t = N+2, \dots, 2N+1 \text{ with } t \neq s-N \quad (40)$$

$$L_{st} = \frac{-\alpha_s - \beta_s}{2N+1} \quad s = N+2, \dots, 2N+1 \text{ and } t = 2, \dots, N+2 \text{ with } t \neq s-N \quad (41)$$

$$L_{s,s+N} = \frac{-\alpha_s - 2N\beta_s}{2N+1} \quad s = 2, \dots, N+1 \quad (42)$$

$$L_{s,s-N} = \frac{-\alpha_s + 2N\beta_s}{2N+1} \quad s = N+2, \dots, 2N+1 \quad (43)$$

$$L_{st} = \frac{-\alpha_s + \beta_s}{2N+1} t, s = 2, \dots, N+1 \text{ and } s \neq t \quad (44)$$

$$L_{st} = \frac{-\alpha_s - \beta_s}{2N+1} s, t = N+2, \dots, 2N+1 \text{ and } s \neq t \quad (45)$$

A. Controlling the sign of non-diagonal Laplacian entries.

The Laplacian matrix obtained with the procedure illustrated above has both positive and negative entries. Signed Laplacians are often used in consensus problems, where negative weights model antagonistic interactions. In other contexts, when e.g. the Laplacian is stemming from diffusive interactions, non diagonal entries are constrained to positive values. In the following, we will provide a set of necessary conditions for the assigned spectrum to eventually yield a Laplacian with positive extra diagonal elements, $L_{ij} > 0$ for $i \neq j$. Clearly, $L_{ii} < 0$, as summing on the rows should return zero. The underlying network,

therefore, displays positive weights, as its adjacency matrix is basically obtained from the Laplacian matrix by replacing the diagonal elements with zeros.

Further, we will set $\Re(\Lambda_j) = \alpha_j < 0$ for $j \geq 2$, an assumption which corresponds to dealing with a stable linear system of the type given in Eq. (25). This requirement immediately yields $L_{11} < 0$, as it follows from relation (33). We will also operate in the setting analyzed above, i.e. assuming $U = qI$. The obtained expressions for the Laplacian elements allow to recast the sought conditions on their signs as:

$$\begin{cases} 2N\alpha_t < \beta_t < -2N\alpha_t \\ \beta_t > -\frac{1-2N}{2N+1}\alpha_t - \frac{2}{2N+1} \sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \beta_t < \frac{1-2N}{2N+1}\alpha_t + \frac{2}{2N+1} \sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \alpha_t < \beta_t < -\alpha_t \\ \frac{\alpha_t}{2N} < \beta_t < -\frac{\alpha_t}{2N} \end{cases} \quad (46)$$

where the inequalities hold for $t = 2, \dots, N+1$.

The above conditions can be simplified, after some algebraic manipulations, to return

$$\frac{4N}{4N^2-1} \sum_k \alpha_k < \alpha_t < \frac{2}{2N+1} \sum_k \alpha_k \quad (47)$$

and

$$\alpha_t - \frac{2}{2N+1} \sum_k \alpha_k < \beta_t < -\alpha_t + \frac{2}{2N+1} \sum_k \alpha_k. \quad (48)$$

The interested reader can access the detailed steps in Appendix C.

Assume that the assigned Laplacian spectrum matches the above conditions, while having $\alpha_k < 0$ for $k > 2$. Then the Laplacian matrix obtained with the above procedure, with $U = qI$, displays positive non diagonal entries.

Let us explore the consequences of conditions (47) and (48). To this end, introduce $\bar{\alpha}$, the average of the non-negative real parts of the Laplacian eigenvalues, i.e. $\bar{\alpha} = (\sum_k \alpha_k)/(2N)$.

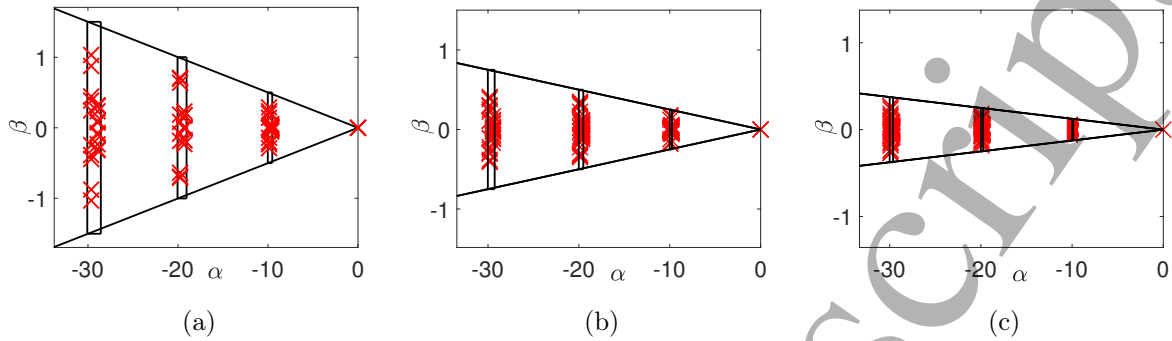


Figure 5. *Examples of discrete spectra (red symbols) which yield a Laplacian with positive non diagonal entries. The rectangular boxes identify the region where eigenvalue should fall for the ensuing network to display positive weights and are traced according to the conditions derived in the main body of the paper. Here, $\Omega = 2N + 1 = 21$ (a), $\Omega = 2N + 1 = 41$ (b) and $\Omega = 2N + 1 = 81$ (c).*

A straightforward analysis allows us to conclude that the generated Laplacian returns positive non-diagonal elements, if the assigned non trivial eigenvalues (i.e. Λ_k , with $k > 1$) fall in a bounded rectangular domain of the complex plane. More specifically, the rectangular region is symmetric, with respect to the horizontal (real) axis, and extends along the vertical direction (imaginary axis) of $\pm 2N\bar{\alpha}/(4N^2 - 1)$. The rectangle is completed by two vertical sides, positioned at $\frac{4N^2}{4N^2-1}\bar{\alpha}$ and $\frac{2N}{2N+1}\bar{\alpha}$. Working at fixed N , the larger is $|\bar{\alpha}|$ the more extended is the rectangle along the vertical direction. Conversely, when making N larger, the rectangle shrinks in the horizontal direction and becomes eventually degenerate for $N \rightarrow \infty$. In other words, for large values of N , eigenvalues should align on a vertical segment positioned at $\bar{\alpha}$, and whose extension increases linearly with $|\bar{\alpha}|$ (while decreasing with N). This is shown in Fig. 5, where three different spectra are depicted (for different choices of $\bar{\alpha}$) which yield a Laplacian with positive off-diagonal elements.

In the following section, we will apply the method of Laplacian generation here discussed to the study of two prototypical examples of dynamical systems on networks.

V. SELECTED APPLICATIONS

In this Section we consider two different models of interacting oscillators, defined on a network. In both cases, the coupling between individual oscillators is implemented via a discrete Laplacian operator, which reflects the specific network arrangement. It will be shown that a suitable network arrangement can be a priori established, building on the procedure illustrated above, so as to make the inspected systems stable against external perturbations.

A. Coupled Stuart-Landau oscillators

Consider an ensemble made of $2N + 1$ nonlinear oscillators and denote by W_i their associated complex amplitude. We assume the oscillators to be mutually coupled via a diffusive-like interaction which is mathematically modeled by a discrete Laplacian operator. Each oscillator obeys a complex Stuart-Landau equation. The dynamics of the system can be cast in the form:

$$\frac{d}{dt}W_j = W_j - (1 + ic_2)|W_j|^2W_j + (1 + ic_1)K \sum_k L_{jk}W_k \quad (49)$$

where c_1 and c_2 are real parameters. The index j runs from 1 to $2N + 1$, the total number of oscillators. Here, K is a suitable parameter setting the coupling strength. Without loss of generality, in what follows it is assumed that $K = 1$. $L_{ij} = A_{ij} - k_i\delta_{ij}$ is the Laplacian, A_{ij} is the generic entry of the directed and weighted adjacency matrix A and $k_i = \sum_j A_{ij}$.

The system admits a homogeneous limit cycle solution in the form $W_{LC}(t) = e^{-ic_2t}$. To characterize the stability of the cycle, one can introduce a non homogeneous perturbation in polar coordinates as:

$$W_i(t) = W_{LC}[1 + \rho_i(t)]e^{i\theta_i(t)} \quad (50)$$

By linearizing around the limit cycle solution ($\rho_i(t) = 0, \theta_i(t) = 0$), one gets:

$$\frac{d}{dt} \begin{pmatrix} \rho_j \\ \theta_j \end{pmatrix} = \begin{pmatrix} -2 & 0 \\ -2c_2 & 0 \end{pmatrix} \begin{pmatrix} \rho_j \\ \theta_j \end{pmatrix} + \begin{pmatrix} 1 & -c_1 \\ c_1 & 1 \end{pmatrix} \sum_k L_{jk} \begin{pmatrix} \rho_k \\ \theta_k \end{pmatrix} \quad (51)$$

To proceed further, expand the perturbations ρ_j and θ_j on the Laplacian eigenvectors $v_{(\alpha)}$, that is

$$\begin{pmatrix} \rho_j \\ \theta_j \end{pmatrix} = \sum_{\alpha=1}^{2N+1} \begin{pmatrix} \rho^{(\alpha)} \\ \theta^{(\alpha)} \end{pmatrix} e^{\lambda_{\alpha} t (v_{(\alpha)})_i} \quad (52)$$

By inserting this expansion in (51) and using the relation

$$\sum_j L_{ij} (v_{(\alpha)})_j = \Lambda^{(\alpha)} (v_{(\alpha)})_i \quad (53)$$

for $\alpha = 1, \dots, 2N + 1$, we obtain a condition formally equivalent to the expression of the continuous dispersion relation

$$\lambda_{max}(\Lambda^{(\alpha)}) = -\Lambda^{(\alpha)} - 1 + \sqrt{-c_1^2 (\Lambda^{(\alpha)})^2 - 2c_1 c_2 \Lambda^{(\alpha)} + 1} \quad (54)$$

If $\lambda_{Re} = \Re(\lambda_{max})$ is positive for some $\Lambda^{(\alpha)}$, the perturbation grows exponentially in time, and the initial homogeneous state turns out to be unstable. Conversely, if $\lambda_{Re} = \Re(\lambda_{max}) < 0$, for every $\Lambda^{(\alpha)}$, the perturbation gets re-absorbed and the system converges back to the fully synchronized state. The condition $\lambda_{Re} < 0$ can be further processed analytically, as discussed in [15]. In particular, it can be shown that the latter condition is fulfilled, if the Laplacian eigenvalues fall in a specific portion of the parameter plane, which reflects the choice made for the reaction parameters c_1, c_2 and K . The region of interest is the one enclosed between the two solid lines, displayed in Fig. 6(a) for the specific selection of the parameters. The blue symbols depicted in Fig. 6(a) are randomly generated so as to fall in the region of the complex plane where stability holds. They represent the spectrum of the Laplacian that we seek to recover following the method illustrated above. In Fig. 6(b), λ_{Re} is plotted against $-\Lambda_{Re} = -\Re(\Lambda)$, confirming the stability of the homogeneous solution.

We now proceed by generating a Laplacian matrix, which is constructed so as to yield the spectrum depicted in Fig. 6(a), see crosses (in blue). Circles (in red) refer to the spectrum obtained after the sparsification. In this case we operated with the more stringent sparsification protocol, the second of those introduced above. This latter appears more suited to the case at hand because of the shape of the domain where the eigenvalues should eventually fall to enforce stability. As an additional condition to the algorithm, we accepted only the sparsification moves which prevent the eigenvalues to invade the region of the complex plane associated to instability.

The sparsified adjacency matrix A defines the network of interactions between coupled oscillators, as follows Eqs. (49). We hence integrate numerically the governing equations, assuming the initial state to be a perturbation of the homogenous synchronized equilibrium. As expected, the perturbation fades away and the system regains its unperturbed, fully synchronized, equilibrium.

B. Coupled Kuramoto oscillators

As a second example we set to study the Kuramoto model. Consider a system made of $2N + 1$ oscillators, denote by θ_i the phase of the i -th oscillator, and ω_i its natural frequency. The oscillators evolve as dictated by the following system of $2N + 1$ coupled differential equations:

$$\dot{\theta}_i = \omega_i + \sum_{j=1}^{2N+1} A_{ij} \sin(\theta_j - \theta_i) \quad i = 1, \dots, 2N + 1 \quad (55)$$

Here, A_{ij} stands for the entries of the adjacency matrix A which sets the interactions between pairs of oscillators. The matrix is, in principle, weighted, and may display positive and negative entries as reflecting the specific interaction (excitatory or inhibitory) being at play.

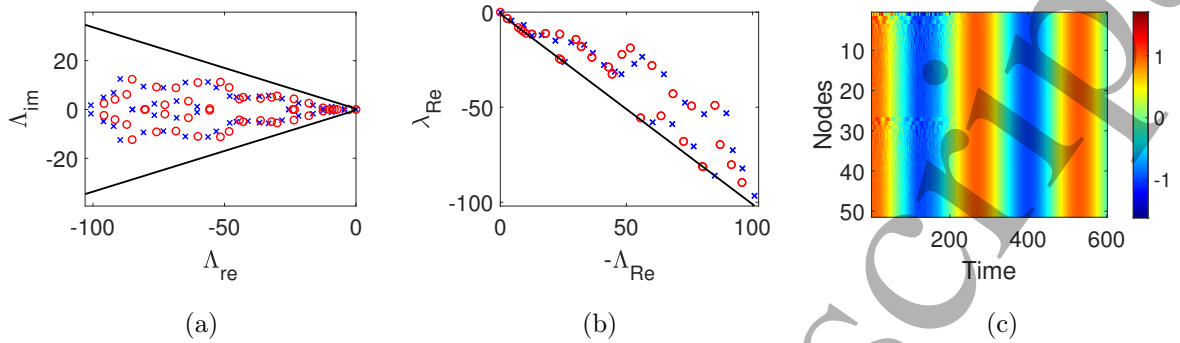


Figure 6. Panel (a): the system of coupled Landau-Stuart oscillators is stable if the eigenvalues of the Laplacian operator fall in the portion of the complex plane comprised between the two solid lines. Crosses (in blue) represent the (randomly assigned) eigenvalues of the Laplacian that we aim at obtaining by means of the procedure introduced in this paper. Circles (in red) refer to the spectrum obtained as follows the sparsification (see main body of the paper). In the sparsification procedure we assumed $\sigma = 0.01$ and $\delta = 5$. The percentage of pruned links is around 35 % of the total.. Panel (b): λ_{Re} is plotted against $-\Lambda_{Re} = -\Re(\Lambda)$. This is an alternative way to show that the system is stable with the prescribed Laplacian spectrum. The solid line stands for the dispersion relation obtained in the continuum limit, when the discrete Laplacian is replaced by a standard differential operator. Panel (c): the time evolution of the real components $\Re(W_j)$ is shown, with an appropriate color code. Here, the weighted network which specifies the coupling between the nodes is obtained from the sparsified Laplacian. Here, $c_1 = 3$, $c_2 = 2.4224$ and $K = 1$.

As an additional assumption, we will here focus in the simplified setting where $\omega_i = \omega \forall i$. We can then introduce the new variable $\psi_i = \theta_i - \omega t$, and write the governing equation in the equivalent form:

$$\dot{\psi}_i = \sum_{j=1}^{2N+1} A_{ij} \sin(\psi_j - \psi_i) \quad i = 1, \dots, 2N + 1 \quad (56)$$

A homogeneous solution always exists with $\psi_i = \Psi \forall i$, and for any constant $\Psi \in [0, 2\pi)$, as it can be immediately checked by substitution. To assess the stability of the solution, one sets $\psi_i = \Psi + \delta_i$, and expands (56) at the leading order in the δ_i . In this way, one gets:

$$\dot{\delta}_i = \sum_{j=1}^{2N+1} A_{ij}(\delta_j - \delta_i) = \sum_{j=1}^{2N+1} L_{ij}\delta_j \quad (57)$$

where $L_{ij} = A_{ij} - k_i\delta_{ij}$ is the Laplacian operator which stems from A_{ij} . The stability of the simplified Kuramoto model here considered is controlled by a linear system of the type introduced in (25), with the obvious replacement of x_i with δ_i . The system proves hence stable if the (non trivial) eigenvalues of the Laplacian operator display negative real parts. Our aim, here, is to generate a Laplacian (and therefore a matrix of binary weighted connections among oscillators) which warrants the stability of the system. To this end we assign the eigenvalues (which appear in conjugate pairs) to belong to the negative portion of the complex plane, see (blue) crosses in Fig. 7(a). The null eigenvalue is clearly included into the spectrum. Running the procedure discussed in the first part of the paper, we obtain the corresponding Laplacian and compute the associated adjacency matrix. The first of the two sparsification procedures is then applied to remove unessential links, yielding the spectrum of the associated Laplacian depicted with (red) circles in Fig. 7(a). The Kuramoto model (56) is then integrated numerically by assuming the recovered expression for A . As predicted, the system is stable to external perturbations as one can clearly appreciate by inspection of Fig. 7(b).

VI. CONCLUSIONS

Studying the dynamics of an ensemble made of interacting units on a network is central for a large plethora of applications. In many cases of interests, individual units evolve under the influence of homologous constituents, the interaction stemming in general from binary exchanges. Distinct fundamental units are assigned to different nodes of the collection, paired via physical or virtual links. For a relevant subclass of problems, the stability of the ensuing equilibrium can be traced back to the spectrum of the Laplacian operator, computed from the adjacency matrix which defines the network arrangement. Symmetric

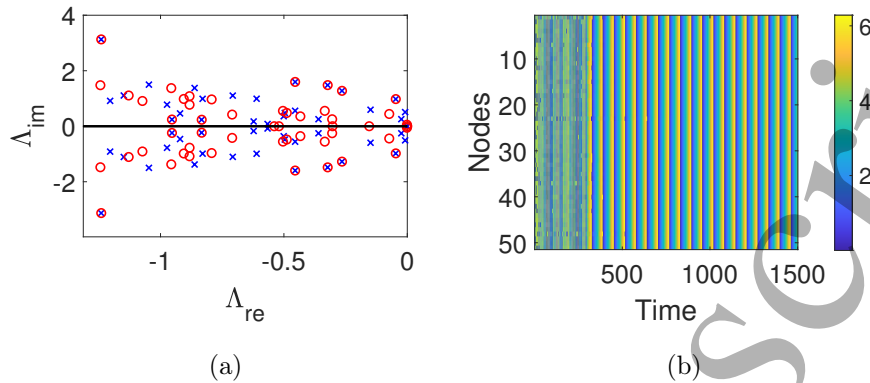


Figure 7. Panel (a): Crosses (in blue) represent the spectrum of the Laplacian operator that we seek to recover. The eigenvalues are distributed in the left portion of the complex plane to assure stability of the inspected Kuramoto model. Circles (in red) stand for the spectrum of the Laplacian after the sparsification procedure. We here apply the first sparsification algorithm as discussed in the main body of the paper. More specifically, unessential links with modest weights are removed, while confining the Laplacian spectrum in a bounded domain, located in the negative portion of the complex plane. Here, $\sigma = 0.01$ and $\delta = 0.2$. The percentage of trimmed links is around 9 % of the total. Panel (b): θ_i vs. time. Here, the adjacency matrix employed in the numerical integration follows the sparsification. As an initial condition, we perturb the homogenous solution (assumed $\Psi = 0$) by a random heterogeneous amount. After a transient, the perturbation gets absorbed and the oscillators evolve in unison. Here $\Omega = 2N + 1 = 51$.

networks yield Laplacian operators with real spectrum, while directionality in the couplings reflects in an imaginary spectrum. Methods exist which allow one to generate a symmetric network, hence Laplacian, with a prescribed real spectrum. Starting from these premises, we have proposed and tested a novel procedure to generate a (signed and directed) Laplacian which returns an a priori assigned (complex) spectrum. A special case has also been considered, which enables one to recover closed analytical expressions for the entries of the sought Laplacian. Working in this setting, we can elaborate on the conditions that have to be matched for the ensuing Laplacian to solely display positive non diagonal elements. Dedicated sparsification procedures are also discussed to help removing unessential links

in terms of their impact on the associated spectrum. The algorithm for Laplacian generation has been successfully tested with reference to two prototypical examples of coupled oscillators. Taken together, our work explores possible strategies for network generation with the emphasis placed on dynamical, rather than structural features. The dynamics is indirectly modulated by the spectrum of the Laplacian operator, which is here constraining the generation algorithm.

-
- [1] M. E. J. Newman, *Networks: An Introduction*. Oxford: Oxford University Press (2010).
 - [2] V. Latora, V. Nicosia, G. Russo, *Complex Networks: Principles, Methods and Applications*. Cambridge University Press (2017).
 - [3] A. Barrat, M. Barthélemy, A. Vespignani, *Dynamical Processes on Complex Networks*. Cambridge University Press (2008).
 - [4] G. Cencetti, F. Bagnoli, G. Battistelli, L. Chisci, D. Fanelli, *Spectral Control for Ecological Stability*, *Eur. Phys. J. B* **91**, 264 (2018).
 - [5] G. Cencetti, P. Clusella, D. Fanelli, *Pattern Invariance for Reaction-Diffusion Systems on Complex Networks*, *Scie. Rep.* **8**, 16226 (2018).
 - [6] M. Ipsen, A. S. Mikhailov, *Evolutionary Reconstruction of Networks*, *Phys. Rev. E* **66**, 046109 (2002).
 - [7] F. Comellas, J. Diaz-Lopez, *Spectral Reconstruction of Complex Networks*, *Physica (Amsterdam)* **387A**, 6436 (2008).
 - [8] D. Cvetkovič, *Spectral Recognition of Graphs*, *Yugoslav Journal of Operations Research* **22**, 145 (2012).
 - [9] B. D. McKay, *On the Spectral Characterisation of Trees*, *Ars Combinatoria* **3**, 219 (1977).
 - [10] L. Halbeisen, N. Hungerbühler, *Reconstruction of Weighted Graphs by their Spectrum*, *Eur. J. Combinatorics* **21**, 641 (2000).
 - [11] E. R. Van Dam, W. H. Haemers, *Which Graphs are Determined by Their Spectrum?*, *Linear Algebra Appl.* **373**, 241 (2003).
 - [12] P. Bötti and R. Merris, *Almost All Trees Share a Complete Set of Immanental Polynomials*, *J. Graph Theory*, **17**, 468 (1993).

- [13] A. E. Motter, *Bounding Network Spectra for Network Design*, New J. Phys. **9**, 182 (2007).
- [14] A. Forrow, F. G. Woodhouse, J. Dunkel, *Functional Control of Network Dynamics Using Designed Laplacian Spectra*, Phys. Rev. X **8**, 041043 (2018).
- [15] G. Cencetti, F. Bagnoli, G. Battistelli, L. Chisci, F. Di Patti, D. Fanelli, *Topological Stabilization for Synchronized Dynamics on Networks*, Eur. Phys. J. B **90**, 9 (2017) .
- [16] D. A. Spielman, N. Srivastava, *Graph Sparsification by Effective Resistances*, SIAM J. Comput. **40**, 1913 (2011).
- [17] W. Ellens, F. M. Spieksma, P. Van Mieghem, A. Jamakovic, R. E. Kooij, *Effective Graph Resistance*, Linear Algebra Appl. **435**, 2491 (2011).
- [18] J. A. Kelner, A. Levin, *Spectral Sparsification in the Semi-streaming Setting*, Theory Comput. Syst. **53**, 243 (2013).
- [19] J. C. Bronski, L. Deville, *Spectral Theory for Dynamics on Graphs Containing Attractive and Repulsive Interactions*, SIAM J. Appl. Math. **74**(1), 83 (2014).
- [20] C. Altafini, *Consensus Problems on Networks With Antagonistic Interactions*, IEEE Transactions on Automatic Control **58**(4), 935 (2013).
- [21] W. Chen, D. Wang, J. Liu, T. Başar, K. H. Johansson, L. Qiu, *On Semidefiniteness of Signed Laplacians with Application to Microgrids*, IFAC-PapersOnLine **49-22**, 097 (2016).
- [22] A. van Harten, *On the Validity of the Ginzburg-Landau Equation*, J. Nonlinear Sci. **1**, 397 (1991).
- [23] I. S. Aranson, L. Kramer, *The World of the Complex Ginzburg-Landau Equation*, Rev. Mod. Phys. **74**, 99 (2001).
- [24] V. Garca-Morales, K. Krischer, *The Complex Ginzburg-Landau Equation: An Introduction*, Contem. Phys. **53**, 79 (2012).
- [25] H. Nakao, *Complex Ginzburg-Landau Equation on Networks and its Non-Uniform Dynamics*, Eur. Phys. J. Spec. Top. **223**, 2411 (2014).
- [26] F. Di Patti, D. Fanelli, F. Miele, T. Carletti, *Benjamin-Feir Instabilities on Directed Networks*, Chaos, Solitons and Fractals **96**, 8 (2017).
- [27] Y. Kuramoto, *Self-Entrainment of a Population of Coupled Non-Linear Oscillators*, Int. Symp. Math. Prob. Theor. Phys. **39**, 420 (1975).
- [28] S. H. Strogatz, *From Kuramoto to Crawford: Exploring the Onset of Synchronization in Populations of Coupled Oscillators*, Physica (Amsterdam) **143D**, 1 (2000).

Appendix A: On the explicit expression of S

Given two arbitrary constants α and β , the following identity holds

$$(I + \alpha E)(I + \beta E) = I + (\alpha + \beta)E + \alpha\beta E^2 \quad (\text{A1})$$

for any matrix E . If E is the matrix defined in (8) one gets:

$$E^2 = NE \quad (\text{A2})$$

and then

$$(I + \alpha E)(I + \beta E) = I + (\alpha + \beta + \alpha\beta N)E \quad (\text{A3})$$

If $\beta = -\frac{\alpha}{1+\alpha N}$, from (A3) we get

$$(I + \alpha E)^{-1} = (I + \beta E) \quad (\text{A4})$$

The above result can be used to derive the structure of matrix S , as reported in the main body of the paper. To this end, we begin by rewriting the matrices A and B as

$$A = [(1 - i)E - iI]U = -i[I + (i + 1)E]U \quad (\text{A5})$$

$$B = [(i + 1)E + iI]U = i[I + (1 - i)E]U \quad (\text{A6})$$

From (A4) we obtain

$$B^{-1} = -iU^{-1}[I + (i + 1)E]^{-1} = -iU^{-1}\left[I + \frac{i - 1}{1 + (1 - i)N}E\right] \quad (\text{A7})$$

and then, after some calculations, we get

$$\begin{aligned} B + AB^{-1}A &= i\left[I + (1 - i)E + I + \frac{3i + 2iN + 1}{1 + (1 - i)N}E\right]U \\ &= 2i\left[I + \frac{1 + i}{1 + (1 - i)N}E\right]U \end{aligned}$$

In conclusion,

$$S = (B + AB^{-1}A)^{-1} = -\frac{i}{2}U^{-1} \left[I - \frac{1+i}{1+2N}E \right]$$

because of

$$\left[I + \frac{1+i}{1+(1-i)N}E \right]^{-1} = \left[I - \frac{1+i}{1+2N}E \right] \quad (\text{A8})$$

which proves the results.

Appendix B: About the computation of L_{ii}

The aim of this section is to detail the computations needed to obtain explicitly the entries of the Laplace matrix L under the assumption $U = qI$ starting thus from Eqs. (16) and (18).

Let us begin by computing the diagonal elements of L , namely L_{ii} for $i = 1, \dots, N$, . For $i = 1$, one gets:

$$L_{11} = \sum_{k=2}^{N+1} V_{1k} D_{kk} W_{k1} \quad (\text{B1})$$

From (4) and (6) we obtain for $k = 2, \dots, N+1$:

$$V_{1k} = -1 - i \quad (\text{B2})$$

$$W_{k1} = \frac{-1+i}{2(2N+1)} \quad (\text{B3})$$

Then

$$L_{11} = 2 \sum_{k=2}^{N+1} \Re \left[(-1-i) D_{kk} \frac{-1+i}{2(2N+1)} \right] \quad (\text{B4})$$

$$= 2 \sum_{k=2}^{N+1} \Re \left[D_{kk} (-1-i) \frac{-1+i}{2(2N+1)} \right] \quad (\text{B5})$$

$$= \frac{2}{2N+1} \sum_{k=2}^{N+1} \alpha_k \quad (\text{B6})$$

where use has been made of the identity $\Re(D_{kk}) = \alpha_k$.

We can proceed in analogy for the others diagonal elements of the Laplacian matrix. In particular, for $s = 2, \dots, N + 1$, we have that

$$W_{ss} = S_{ss} = \frac{-2Ni - 1}{2(2N + 1)}, \quad (\text{B7})$$

and then

$$L_{ss} = 2 \sum_{k=2}^{N+1} \Re[D_{kk} V_{sk} W_{ks}] \quad (\text{B8})$$

$$= 2\Re(D_{ss} V_{ss} W_{ss}) \quad (\text{B9})$$

$$= 2\Re\left[(\alpha_s + i\beta_s)i \frac{-2Ni - 1}{2(2N + 1)}\right] \quad (\text{B10})$$

$$= \frac{2N\alpha_s + \beta_s}{2N + 1} \quad (\text{B11})$$

Conversely, for $s = N + 2, \dots, 2N + 1$, we have

$$W_{ss} = iS_{ss} = i \frac{-2Ni - 1}{2(2N + 1)}, \quad (\text{B12})$$

and consequently:

$$L_{ss} = 2\Re\left[(\alpha_s - i\beta_s)i \frac{-2Ni - 1}{2(2N + 1)}\right] \quad (\text{B13})$$

$$= \frac{2N\alpha_s - \beta_s}{2N + 1} \quad (\text{B14})$$

Let us proceed now with the other elements of the first row of L . For $t = 2, \dots, N + 1$ we

can write

$$L_{1t} = 2 \sum_{k=2}^{N+1} \Re(V_{1k} D_{kk} W_{kt}) \quad (\text{B15})$$

$$= 2 \left[\Re(V_{1t} D_{tt} W_{tt}) + \sum_{k=2, k \neq t}^{N+1} \Re(V_{1k} D_{kk} W_{kt}) \right] \quad (\text{B16})$$

$$= 2 \left\{ \Re \left[(-1 - i)(\alpha_t + i\beta_t) \frac{-2Ni - 1}{2(2N + 1)} \right] + \right. \quad (\text{B17})$$

$$\left. \sum_{k=2, k \neq t}^{N+1} \Re \left[(-1 - i)(\alpha_k + i\beta_k) \frac{i - 1}{2(2N + 1)} \right] \right\} \quad (\text{B18})$$

$$= \frac{1 - 2N}{1 + 2N} \alpha_t - \beta_t + \frac{2}{2N + 1} \sum_{k=2, k \neq t}^{N+1} \alpha_k \quad (\text{B19})$$

and for $t = N + 2, \dots, 2N + 1$ we get

$$L_{1t} = 2 \left\{ \Re \left[(-1 + i)(\alpha_t - i\beta_t) i \frac{-2Ni - 1}{2(2N + 1)} \right] + \right. \quad (\text{B20})$$

$$\left. \sum_{k=N+2, k \neq t}^{2N+1} \Re \left[(-1 + i)(\alpha_k - i\beta_k) i \frac{i - 1}{2(2N + 1)} \right] \right\} \quad (\text{B21})$$

$$= \frac{1 - 2N}{1 + 2N} \alpha_t + \beta_t + \frac{2}{2N + 1} \sum_{k=N+2, k \neq t}^{2N+1} \alpha_k \quad (\text{B22})$$

$$= \frac{1 - 2N}{1 + 2N} \alpha_t + \beta_t + \frac{2}{2N + 1} \sum_{k=2, k \neq t-N}^{N+1} \alpha_k \quad (\text{B23})$$

For $s = 2, \dots, N + 1$ and $t = N + 2, \dots, 2N + 1$ with $t \neq s - N$, we get:

$$L_{st} = \sum_{k=2}^{2N+1} V_{sk} D_{kk} W_{kt} \quad (\text{B24})$$

$$= V_{ss} D_{ss} W_{st} + V_{s, s+N} D_{s+N, s+N} W_{s+N, t} \quad (\text{B25})$$

$$= i(\alpha_s + i\beta_s)(-i) \frac{-i - 1}{2(2N + 1)} + \quad (\text{B26})$$

$$(-i)(\alpha_s - i\beta_s) i \frac{i - 1}{2(2N + 1)} \quad (\text{B27})$$

$$= \frac{-\alpha_s + \beta_s}{2N + 1} \quad (\text{B28})$$

while, for $s = N + 2, \dots, 2N + 1$ and $t = 2, \dots, N + 2$ with $t \neq s - N$, the following expression holds:

$$L_{st} = V_{ss}D_{ss}W_{st} + V_{s,s-N}D_{s-N,s-N}W_{s-N,t} \quad (\text{B29})$$

$$= (\alpha_s - i\beta_s)\frac{-i-1}{2(2N+1)} + (\alpha_s + i\beta_s)\frac{i-1}{2(2N+1)} \quad (\text{B30})$$

$$= -\frac{\alpha_s + \beta_s}{2N+1} \quad (\text{B31})$$

For $s = 2, \dots, N + 1$

$$L_{s,s+N} = \quad (\text{B32})$$

$$V_{ss}D_{ss}W_{s,s+N} + V_{s,s+N}D_{s+N,s+N}W_{s+N,s+N} \quad (\text{B33})$$

$$= i(\alpha_s + i\beta_s)(-i)\frac{2Ni-1}{2(2N+1)} + \quad (\text{B34})$$

$$(-i)(\alpha_s - i\beta_s)i\frac{-2Ni-1}{2(2N+1)} = \quad (\text{B35})$$

$$-\frac{\alpha_s + 2N\beta_s}{2N+1} \quad (\text{B36})$$

while, for $s = N + 2, \dots, 2N + 1$, we obtain:

$$L_{s,s-N} = \quad (\text{B37})$$

$$V_{ss}D_{ss}W_{s,s-N} + V_{s,s-N}D_{s-N,s-N}W_{s-N,s-N} \quad (\text{B38})$$

$$= (\alpha_s - i\beta_s)\frac{2Ni-1}{2(2N+1)} + (\alpha_s + i\beta_s)i\frac{-2Ni-1}{2(2N+1)} = \quad (\text{B39})$$

$$\frac{\alpha_s - 2N\beta_s}{2N+1} \quad (\text{B40})$$

For $t = 2, \dots, N + 1$

$$L_{t1} = 2\Re(V_{tt}D_{tt}W_{t1}) = \quad (\text{B41})$$

$$2\Re\left[i(\alpha_t + i\beta_t)\frac{i-1}{2(2N+1)}\right] = \quad (\text{B42})$$

$$-\frac{\alpha_t - \beta_t}{2N+1} \quad (\text{B43})$$

and for $t = N + 2, \dots, 2N + 1$

$$L_{t1} = 2\Re(V_{tt}D_{tt}W_{t1}) \quad (\text{B44})$$

$$= 2\Re\left[(\alpha_t - i\beta_t)\frac{-i-1}{2(2N+1)}\right] \quad (\text{B45})$$

$$= \frac{-\alpha_t - \beta_t}{2N+1} \quad (\text{B46})$$

For $t, s = 2, \dots, N + 1$ and $s \neq t$

$$L_{st} = 2\Re(V_{ss}D_{ss}W_{st}) \quad (\text{B47})$$

$$= 2\Re\left[i(\alpha_s + i\beta_s)\frac{i-1}{2(2N+1)}\right] \quad (\text{B48})$$

$$= \frac{-\alpha_s + \beta_s}{2N+1} \quad (\text{B49})$$

and, finally, for $s, t = N + 2, \dots, 2N + 1$ and $s \neq t$, one gets:

$$L_{st} = 2\Re(V_{ss}D_{ss}W_{st}) \quad (\text{B50})$$

$$= 2\Re\left[(\alpha_s - i\beta_s)i\frac{i-1}{2(2N+1)}\right] = \quad (\text{B51})$$

$$-\frac{\alpha_s + \beta_s}{2N+1} \quad (\text{B52})$$

Summing up, we have here provided closed-form analytical expressions for all the entries of the Laplacian matrix, as a function of the assigned spectrum.

Appendix C: Positiveness of L

The aim of this section is to work out the algebraic steps needed to rewrite (46) in the simpler form given by (47) and (48).

Let us thus rewrite (46)

$$\begin{cases} 2N\alpha_t < \beta_t < -2N\alpha_t \\ \beta_t > -\frac{1-2N}{2N+1}\alpha_t - \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \beta_t < \frac{1-2N}{2N+1}\alpha_t + \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \alpha_t < \beta_t < -\alpha_t \\ \frac{\alpha_t}{2N} < \beta_t < -\frac{\alpha_t}{2N} \end{cases}$$

where the inequalities hold for $t = 2, \dots, N + 1$. Then the second and the third conditions of system (46) can be matched simultaneously provided that:

$$\alpha_t < \frac{2}{2N-1}\sum_{k \neq t} \alpha_k \quad \forall t = 2, \dots, N + 1 \quad (\text{C1})$$

Remark that:

$$2N\alpha_t < \alpha_t < \frac{\alpha_t}{2N} \quad (\text{C2})$$

holds for arbitrary values of $\alpha_t < 0$ and N . Hence, system (46) simplifies as follows:

$$\begin{cases} \frac{\alpha_t}{2N} < \beta_t < -\frac{\alpha_t}{2N} \\ \alpha_t < \frac{2}{2N-1}\sum_{k \neq t} \alpha_k \\ \beta_t > -\frac{1-2N}{2N+1}\alpha_t - \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \beta_t < \frac{1-2N}{2N+1}\alpha_t + \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \end{cases} \quad (\text{C3})$$

for $t = 2, \dots, N + 1$.

Let us now focus on the conditions for β_t . We assume that the following condition holds:

$$\frac{1-2N}{2N+1}\alpha_t + \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k > -\frac{\alpha_t}{2N} \quad (\text{C4})$$

and we set to explore its consequences. Eq. (C4) yields:

$$\alpha_t < \frac{4N}{4N^2 - 4N - 1}\sum_{k \neq t} \alpha_k = \frac{1}{N-1 - \frac{1}{4N}}\sum_{k \neq t} \alpha_k \quad (\text{C5})$$

For $N > 1$, conditions (C3) maps therefore in the following equivalent system:

$$\begin{cases} \frac{\alpha_t}{2N} < \beta_t < -\frac{\alpha_t}{2N} \\ \alpha_t < \frac{2}{2N-1} \sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \alpha_t < \frac{1}{N-1-\frac{1}{4N}} \sum_{k \neq t} \alpha_k \end{cases} \quad (\text{C6})$$

Remark that:

$$\frac{1}{N-1-\frac{1}{4N}} \sum_{k=2, k \neq t}^{N+1} \alpha_k < \frac{2}{2N-1} \sum_{k=2, k \neq t}^{N+1} \alpha_k \quad (\text{C7})$$

due to the inequality

$$\frac{1}{N-1-\frac{1}{4N}} > \frac{2}{2N-1} \quad (\text{C8})$$

which holds for each $N > 1$. Hence, system (C6) takes the form:

$$\begin{cases} \frac{\alpha_t}{2N} < \beta_t < -\frac{\alpha_t}{2N} \\ \alpha_t < \frac{1}{N-1-\frac{1}{4N}} \sum_{k \neq t} \alpha_k \end{cases} \quad (\text{C9})$$

Then, (C9) has no solutions, under the working hypothesis that we have put forward to deriving it. In fact:

$$\alpha_t + \frac{1}{N-1-\frac{1}{4N}} \alpha_t < \frac{1}{N-1-\frac{1}{4N}} \sum_k \alpha_k \quad (\text{C10})$$

that is

$$\alpha_t < \frac{1}{N-\frac{1}{4N}} \sum_k \alpha_k = \frac{4N}{4N^2-1} \sum_k \alpha_k \quad (\text{C11})$$

and summing on every t we get

$$\sum_t \alpha_t < \frac{4N^2}{4N^2-1} \sum_k \alpha_k \quad (\text{C12})$$

which, in turn, implies $\frac{4N^2}{4N^2-1} < 1$, a condition that is obviously never met. We now go back to revise ansatz (C4), and consider the alternative scenario:

$$\frac{1-2N}{2N+1} \alpha_t + \frac{2}{2N+1} \sum_{k=2, k \neq t}^{N+1} \alpha_k < -\frac{\alpha_t}{2N} \quad (\text{C13})$$

Then, (C3) becomes

$$\begin{cases} \beta_t > -\frac{1-2N}{2N+1}\alpha_t - \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \beta_t < \frac{1-2N}{2N+1}\alpha_t + \frac{2}{2N+1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \alpha_t < \frac{2}{2N-1}\sum_{k=2, k \neq t}^{N+1} \alpha_k \\ \alpha_t > \frac{1}{N-1-\frac{1}{4N}}\sum_{k \neq t} \alpha_k \end{cases} \quad (\text{C14})$$

Following a path analogous to the one discussed above, we get:

$$\frac{4N}{4N^2-1}\sum_k \alpha_k < \alpha_t < \frac{2}{2N+1}\sum_k \alpha_k \quad (\text{C15})$$

and

$$\alpha_t - \frac{2}{2N+1}\sum_k \alpha_k < \beta_t < -\alpha_t + \frac{2}{2N+1}\sum_k \alpha_k \quad (\text{C16})$$