Many complex biological and technological systems can be represented by multilayer networks where the nodes are coupled via several independent networks. Despite its significance from both the theoretical and application perspectives, synchronization in multilayer networks and its dependence on the network topology remain poorly understood. In this paper, we develop a universal connection graph-based method which removes a long-standing obstacle to studying synchronization in dynamical multilayer networks. This method opens up the possibility of explicitly assessing critical multilayer-induced interactions which can hamper network synchronization and reveals striking, counterintuitive effects caused by multilayer coupling. It demonstrates that a coupling which is favorable to synchronization in single-layer networks can reverse its role and destabilize synchronization when used in a multilayer network. This property is controlled by the traffic load on a given edge when the replacement of a lightly loaded edge in one layer with a coupling from another layer can promote synchronization, but a similar replacement of a highly loaded edge can break synchronization, forcing a “good” link go “bad.” This method can be transformative in the highly active research field of synchronization in multilayer engineering and social networks, especially in regard to hidden effects not seen in single-layer networks.

PACS numbers: 05.45.Xt, 87.19.La

I. INTRODUCTION

Complex networks are common models for many systems in physics, biology, engineering, and the social sciences [1–3]. Significant attention has been devoted to algebraic, statistical, and graph theoretical properties of networks and their relationship to network dynamics (see a review [4] and references therein). The strongest form of network cooperative dynamics is synchronization which has been shown to play an important role in the functioning of a wide spectrum of technological and biological networks [5–14], including adaptive and evolving networks [15–22].

Despite the vast literature to be found on network dynamics and synchronization, the majority of research activities have been focused on oscillators connected through single-layer network (one type of coupling) [23–41]. However, in many realistic biological and engineering systems the units can be coupled via multiple, independent systems and networks. Neurons are typically connected through different types of couplings such as excitatory, inhibitory, and electrical synapses, each corresponding to a different circuitry where the interplay affects network function [42, 43]. Pedestrians on a lively bridge are coupled via several layers of communication, including people-to-people interactions and a feedback from the bridge that can lead to complex pedestrian-bride dynamics [44–47]. In engineering systems, examples of independent networks include coupled grids of power stations and communication servers where the failure of nodes in one network may lead to the failure of dependent nodes in another network [48]. Such interconnected networks can be represented as multiplex or multilayer networks [49, 50] which include multiple systems and layers of connectivity. Multilayer-induced correlations can have significant ramifications for the dynamical processes on networks, including the effects on the speed of disease transmission in social networks [51] and the role of redundant interdependences on the robustness of multiplex networks to failure [52].

Typically, in single-layer networks of continuous time oscillators, synchronization becomes stable when the coupling strength between the oscillators exceeds a threshold value [23, 30]. This threshold depends on the individual oscillator dynamics and the network topology. In this context, a central question is to determine the critical coupling strength so that the stability of synchronization is guaranteed. The master stability function [23] or the connection graph method [30, 31] are usually used to answer this question in single-layer networks. Both methods reduce the dimensionality of the problem such that synchronization in a large, complex network can be predicted from the dynamics of the individual node and the network structure.

Synchronization in multilayer networks has been studied in [53–56]; however, its critical properties and explicit dependence on intralayer and interlayer network structures remain poorly understood. This is in particular due to the inability of the existing eigenvalue methods, including the master stability function [23] to give detailed insight into the stability condition of synchronization as the eigenvalues, corresponding to connection graphs composing a multilayer network, must be calculated via simultaneous diagonalization of two or more connectivity matrices. Simultaneous diagonalization of two or more matrices is impossible in general, unless the matrices commute [53, 54]. A nice approach based on simultaneous block diagonalization of two connectivity matrices was proposed in [54]. This most successful ap-
plication of the eigenvalue-based approach allows one to reduce the dimensionality of a large network to a smaller network whose synchronization condition can be used to evaluate the stability of synchronization in the large network. For some network topologies, this technique yields a substantial reduction of the dimensionality; however, this reduction is less significant in general. The reduced network typically contains weighted positive and negative connections, including self loops such that the role of multilayer network topologies and the location of an edge remain difficult to evaluate.

In this paper, we report our significant progress towards removing this long-standing obstacle to studying synchronization in multilayer networks. We develop a new general stability approach, called the Multilayer Connection Graph method, which does not depend on explicit knowledge of the spectrum of the connectivity matrices and can handle multilayer networks with arbitrary network topologies, which are out of reach for the existing approaches. An example of a multilayer network in this study is a network of Lorenz systems where some of the oscillators are coupled through the $x$ variable (first layer), some through the $y$ variable (second layer), and some through both (interlayer connections). Our Multilayer Connection Graph method originates from the connection graph method [30, 31] for single-layer networks; however, this extension is highly non-trivial and requires overcoming a number of technically challenging issues. This includes the fact that the oscillators from two $x$ and $y$ layers in the networks of Lorenz systems are connected through the intrinsic, nonlinear equations of the Lorenz system. As a result, multilayer networks can have drastically different synchronization properties from those of single-layer networks. In particular, our method shows that an interlayer traffic load on a link is the crucial quantity which can be used to foster or hamper synchronization in a nonlinear fashion. For example, it demonstrates that replacing a link with a light interlayer traffic load by a stronger pairwise converging coupling (a “good” link) via another layer may lower the synchronization threshold and improve synchronizability. At the same time, such a replacement of a highly loaded link can make the network unsynchronizable, forcing the pairwise stabilizing “good” link go “bad.”

The layout of this paper is as follows. First, in Sec. II, we present and discuss the network model. In Sec. III, we start with a motivating example of why the replacement of some links in a multilayer network can improve or break network synchronization. And what is an underlying mechanism? In Sec. IV, we formulate the Multilayer Connection Graph method for predicting synchronization in multilayer networks. In Secs. V and VI, we show how to apply the general method to specific network topologies. In Sec. VII, a brief discussion of the obtained results is given. Finally, the Appendix contains the complete derivation of the general method. MATLAB code for algorithms used for calculating network traffic loads are given in the Supplement.

II. NETWORK MODEL AND PROBLEM STATEMENT

We start with a general network of $n$ oscillators with two connectivity layers:

$$\frac{dx_i}{dt} = F(x_i) + \sum_{j=1}^n c_{ij} P x_j + \sum_{j=1}^n d_{ij} L x_j, \quad i = 1, ..., n, (1)$$

where $x_i = (x_{i1}, ..., x_{in})$ is the $s$ state vector containing the coordinates of the $i$-th oscillator, $F : \mathbb{R}^s \rightarrow \mathbb{R}^s$ describes the oscillators’ individual dynamics, $c_{ij}$ and $d_{ij}$ are the coupling strengths. $C = (c_{ij})$ and $D = (d_{ij})$ are $n \times n$ Laplacian connectivity matrices with zero-row sums and nonnegative off-diagonal elements $c_{ij} = c_{ji}$ and $d_{ij} = d_{ji}$, respectively [30]. These connectivity matrices $C$ and $D$ define two different connection layers (also denoted by $C$ and $D$, with $m$ and $l$ edges, respectively). The inner matrices $P$ and $L$ determine which variables couple the oscillators within the $C$ and $D$ layers, respectively. Without loss of generality, we will be considering the oscillators of dimension $s = 3$ and $x_i = (x_i, y_i, z_i)$. Therefore, the $C$ graph with the inner matrix $P = \text{diag}(1,0,0)$ will correspond to the first-layer connections via $x$, while the $D$ graph with the inner matrix $L = \text{diag}(0,1,0)$ will indicate the second-layer connections via $y$. Overall, the oscillators of the network are connected through a combination of the two layers (see Fig. 1 for an example of a combined two-layer graph). The graphs are assumed to be undirected [30]. Oscillators, comprising the network (1), can be periodic or chaotic. As chaotic oscillators are difficult to synchronize, they are usually used as test bed examples for probing the effectiveness of a given stability approach. The oscillators used in the numerical verification of our stability method are chaotic Lorenz [57] and double scroll oscillators [58].

In this paper, we are interested in the stability of complete synchronization defined by the synchronization manifold $M = \{x_1 = x_2 = ... = x_n\}$. Our main objective is to determine a threshold value for the coupling strengths required for the global stability of synchronization. We seek to predict this threshold or the absence thereof in the general network (1) from synchronization in the simplest two-node network and graph properties of the two-layer network structures.

It is important to emphasize that only Type I oscillators [26] are capable of synchronizing globally and retaining synchronization for any coupling strengths exceeding the synchronization threshold. Most known oscillators, including the Lorenz and double scroll oscillators belong to Type I systems. A much more narrow Type II class of oscillators contains $x$-coupled Rössler systems [23] in which synchronization becomes only locally stable and eventually loses its stability with an increase of coupling [24]. As a result, we limit our consideration to Type I networks in this paper; however, a numerically-assisted extension of our method to Type II networks can be performed with moderate effort and remains a subject
of future study.

(a)

(b)

(c)

(d)

FIG. 1. The puzzle: why do “good” links go “bad”? Synchronization in six-node networks of Lorenz systems (2). (a) Single-layer network, with all x edges (black). (b) The replacement of x edge 5-6 with a presumably better converging y coupling (blue) improves synchronization, as expected (see (d)). (c) A similar replacement of x edge 2-3 with a y edge (red) makes synchronization impossible by pushing the threshold to infinity (see (d)). (d) Systematic study of the coupling threshold \( c^* \) as a function of the y edge location that replaces an x edge in the original x-coupled network (a). The blue solid line indicates numerically calculated thresholds. The black solid line depicts the interlayer traffic \( b^{int} \) for the respective y edge. Note a significant increase of \( b^{int} \) that causes the network to become unsynchronizable as predicted by the method. The predicted coupling thresholds (blue dotted line) are computed from (3) using the exponential fit in Fig. 3 and scaling factors \( \beta = 0.18 \) and \( \gamma = 0.7180 \).

III. A MOTIVATING EXAMPLE AND A PUZZLE

To illustrate the complexity of assessing the role of multilayer connections and their controversial role in fostering or hindering synchronization, we begin with two-layer networks of chaotic Lorenz oscillators, depicted in Fig. 1(a-c). For the Lorenz oscillators, the vector equation (1) can be written in a more reader-friendly scalar form:

\[
\begin{align*}
\dot{x}_i &= \sigma(y_i - x_i) + \sum_{j=1}^{n} c_{ij} x_j, \\
\dot{y}_i &= r x_i - y_i - x_i z_i + \sum_{j=1}^{n} d_{ij} y_j, \\
\dot{z}_i &= -b z_i + x_i y_i, \quad i = 1, \ldots, n,
\end{align*}
\]

where the connectivity matrix \( C = (c_{ij}) \) describes the topology of x connections (black edges in Fig. 1(a-c)), and matrix \( D = (d_{ij}) \) describes the location of one y edge (blue or red edge). The parameters of the individual Lorenz oscillator are standard: \( \sigma = 10, r = 28, \) and \( b = 8/3 \). The strengths of the x and y coupling are homogeneous \( (c_{ij} = c, d_{ij} = d) \) and varied uniformly \( (c = d) \).

We are interested in the question of how the replacement of an x edge in the network of Fig. 1(a) with a y edge can affect synchronization. To address this question, we first need to understand synchronization properties of two-node single-layer networks of x-coupled and y-coupled Lorenz systems. It is well-known that if the coupling in a single-layer network (2) with either all x or all y connections exceeds a critical threshold, then synchronization becomes stable and persists for any \( c > c^* \) and \( d > d^* \), respectively. A rigorous upper bound for the critical coupling \( c^* \), explicit in parameters of the Lorenz oscillator can be found in [30].

Calculated numerically, these coupling thresholds are for \( c^* \approx 3.81 \) for the two-node x-coupled network and \( d^* \approx 1.42 \) for the y-coupled network. As the synchronization threshold \( d^* \) is significantly lower, then one could expect that replacing an x edge with a presumably better converging y coupling improves synchronization. This is true for the network of Fig. 1(b) when x edge 5-6 is replaced with a y edge, yielding a minor reduction in the synchronization threshold from \( c^* \approx 17.94 \) in the single-layer x-coupled network of Fig. 1(a) to \( c^* \approx 17.74 \) in the multilayer network of Fig. 1(b). A surprise comes from the network of Fig. 1(c) where the replacement of x edge with a y edge, that would be expected to improve synchronization, on the contrary makes the network unsynchronizable (see the coupling threshold jumping to infinity in Fig. 1(d)).

What is the origin of this highly counterintuitive effect? Why do edges in a multilayer network reverse their stabilizing roles depending on the edge location whereas they are well behaved in single-layer networks? As the connectivity matrices for x and y coupling in the networks of Fig. 1(b) and Fig. 1(c) do not commute and, therefore, the predictive power of the master stability function based methods [53–55] is severely impaired, this puzzle calls for an explanation and ultimately motivates the development of an effective, general method for assessing the stability of synchronization in multilayer networks.

In the following, we will develop such a method that
identifies the location of critical interlayer links which control stable synchronization and reveals its explicit dependence on an interlayer traffic load on a given edge.

IV. MULTILAYER CONNECTION GRAPH METHOD

In this section, we first present the main result of this paper and formulate the method. We will then demonstrate how to apply the method to specific network configurations.

Theorem 1 [The method]. Synchronization in the multilayer network (1) is globally stable if for each edge $k = 1, \ldots, m$:

\[
\begin{align*}
    c_{ij} &\equiv c_k > \frac{1}{n} \left( a_x \cdot b^x_k + \omega_x \cdot b^{int}_k + \alpha^x_k \right) \\
    d_{ij} &\equiv d_k > \frac{1}{n} \left( a_y \cdot b^y_k + \omega_y \cdot b^{int}_k + \alpha^y_k \right),
\end{align*}
\]

where $b^x_k = \sum_{j \neq i, k \in P_{ij}} |P_{ij}|$ is the sum of the lengths of all chosen paths $P_{ij}$ between any pair of oscillators $i$ and $j$ which belong to the $x$ graph $C$ and go through a given $x$ edge. The path length $|P_{ij}|$ is the number of edges comprising the path. Similarly, $b^{int}_k$ corresponds to the sum of the lengths of all chosen paths which contain $x$ and $y$ edges and belong to the combined, two-layer $xy$ graph and go through a given edge $k$ which may be an $x$ or $y$ edge. Constant $a_x$ ($a_y$) is the double coupling strength sufficient for synchronization in the network of two $x$-coupled ($y$-coupled) oscillators, composing the network. The constants $\omega_x$ and $\omega_y$ represent a combination of the double coupling strengths for $c_{12}$ and $d_{12}$ sufficient for synchronization in the two-oscillator network with both $x$ and $y$ connections. Finally, the constants $\alpha^x_k$ and $\alpha^y_k$ are chosen large enough such that they can globally stabilize the auxiliary stability systems written for the difference variables that correspond to an edge $k$:

\[
\begin{align*}
    \dot{X}_k &= \mathbf{X}_{ij} = x_j - x_i, \quad \text{for } \alpha^x_k:
    \dot{X}_k &= \left[ \int_0^1 DF(\beta x_j + (1 - \beta)x_i, d\beta) \right]X_k + \\
    &\quad \omega_x b^{int}_k LX_k - (a_x + \alpha^x_k)PX_k, \\
    \dot{X}_k &= \left[ \int_0^1 DF(\beta x_j + (1 - \beta)x_i, d\beta) \right]X_k + \\
    &\quad \omega_y b^{int}_k LX_k - (a_y + \alpha^y_k)PX_k, \quad \text{for } \alpha^y_k:
\end{align*}
\]

where the Jacobian $DF$ can be calculated explicitly via the parameters of the individual oscillator.

Proof. The proof closely follows the notations and steps in the derivation of the connection graph method [30] for single-layer networks up to a point where the stability argument becomes drastically different and yields the new terms $\omega_x b^{int}_k$ and $\omega_y b^{int}_k$ which play a pivotal role in synchronization of multilayer networks. The complete proof is given in the Appendix. □

Remark 1. In terms of traffic networks, the graph theoretical quantities $b^x_k$, $b^y_k$ and $b^{int}_k$ represent the total lengths of the chosen roads that go through a given edge $k$ which can be loosely analogized as a busy street. Therefore, we refer to them as "traffic" loads. In this view, the quantity $b^{int}_k$ is a traffic load on edge $k$, caused by interlayer travelers. It is important to notice that positive terms $+\omega_x b^{int}_k LX_k$ in the equation (4) and $+\omega_y b^{int}_k PX_k$ in (5) play a destabilizing role such that a heavily loaded edge with a high $b^{int}_k$ can represent a potential problem for making the systems (4) and (5) stable at all. This observation has dramatic consequences for synchronization in specific multilayer networks described in the following sections and also is a key to solving the puzzle of Fig. 1.

Remark 2. If both $x$ and $y$ connection graphs are connected such that all oscillators are coupled via both graphs, the stability of synchronization can be simply assessed by applying the connection graph method [30] for each of the $x$ and $y$ connected graphs and combining the two conditions as follows: $c_{ij} + d_{ij} = c_k + d_k > \frac{1}{n} \left( (a_x \cdot b^x_k + a_y \cdot b^y_k) \right)$ (cf. the condition (37) in the Appendix). As a result, one should not expect the effects due to the multilayer coupling discussed in the motivating example.

Remark 3. While the stability criterion (3) is completely rigorous, the theoretically calculated bounds may give large overestimates on the threshold coupling strength. As a result, we suggest to use a semi-analytical approach which combines numerically calculated exact bounds $a_x$, $a_y$, $\omega_x$, $\omega_y$, $\alpha^x_k$, and $\alpha^y_k$ with graph theoretical quantities $b^x$, $b^y$, and $b^{int}$. In this way, this method becomes an effective, predictive tool and plays the role of the master stability function for the general multilayer network (1) where a synchronization threshold or the absence thereof can be deduced from the properties of the individual oscillators comprising the network and the network topologies of the connection layers. In this numerically-assisted application of our method, the conservative estimate on traffic loads $b^x$, $b^y$, and $b^{int}$, that come from the Cauchy-Schwartz inequality (see the Appendix), can be replaced by $\beta b^x$, $\beta b^y$, and $\gamma b^{int}$, where the scaling factors $\beta$ and $\gamma$ are introduced to provide tighter numerical bounds.

Computing the stability conditions (3) and its constants is a complicated, multi-step process which involves calculations of (i) two stability diagrams (for a pair of $\omega^x, \omega^y$ and $\alpha^x, \alpha^y$) and (ii) graph theoretical quantities $b^x$, $b^y$, and $b^{int}$. In the following section, we will walk the reader through a step-by-step computation of the stability conditions (3) for specific multilayer networks and illustrate their implications for the stability of synchronization.
the stability conditions (4) and (5).

Consider the simplest two-node network (2) with both $a$ and $d$-coupled networks.

Step 1: Calculate $a_x$, $a_y$, $\omega_x$, and $\omega_y$.

Consider the simplest two-node network (2) with both $x$ and $y$ coupling: $c_{12} = c_{21} = c$ and $d_{12} = d_{21} = d$. Determine threshold coupling strengths that guarantee stable synchronization in (i) the two-node $x$-coupled network with $c^* = a_x/2$ and $d = 0$; (ii) the two-node $y$-coupled network with $d^* = a_y/2$ and $c = 0$; and (iii) the two-node $xy$ network with $c^* = \omega_x$ and $d^* = \omega_y$.

The numerically calculated thresholds for the $x$-coupled and $y$-coupled networks (2) reported in Sec. III are $c^* \approx 7.62/2$ and $d^* \approx 2.84/2$, respectively. This yields the double coupling strength constants $a_x = 7.62$ and $a_y = 2.84$ to be used in (3).

Note that different combinations of $c = \omega_x$ and $d = \omega_y$ in the $xy$-coupled network yield stable synchronization (see Fig. 2). Without loss of generality, we choose $c = \omega_x = 5$ and $d = \omega_y = 0.5$ as a point on the stability boundary in Fig. 2 and keep these values fixed for the prediction of the synchronization threshold in larger two-layer networks (2) with arbitrary topologies. This choice of the pair $(\omega_x, \omega_y)$ is somewhat arbitrary; however, it dictates the choice of constants in the stability diagrams for $\alpha^x_k$ and $\alpha^y_k$ (Fig. 3). It is often a good idea to choose $\omega_x$ and $\omega_y$ such that both are non-zero and lie somewhere in the middle range of $(\omega_x, \omega_y)$ to balance out the stability conditions (4) and (5).

Step 2: Calculate the stability diagrams to determine $\alpha^x_k$ and $\alpha^y_k$.

The auxiliary stability systems (4) and (5) for the scalar difference variables of coupled Lorenz oscillators (2)

$$X_k = x_j - x_i, \quad Y_k = y_j - y_i, \quad Z_k = z_j - z_i$$

can be rewritten in a more convenient form:

$$\begin{align*}
\dot{X}_k &= \sigma(Y_k - X_k) - (a_x + \alpha^x_k) X_k, \\
\dot{Y}_k &= (r - U^{(z)}(t)) X_k - Y_k - U^{(x)}(t) Z_k + \omega_x b^{int}_k Y_k, \\
\dot{Z}_k &= U^{(y)}(t) X_k + U^{(x)}(t) Y_k - b Z_k,
\end{align*}$$

$$\begin{align*}
\dot{X}_k &= \sigma(Y_k - X_k) + \omega_y b^{int}_k X_k, \\
\dot{Y}_k &= (r - U^{(z)}(t)) X_k - Y_k - U^{(x)}(t) Z_k - (a_y + \alpha^y_k) Y_k, \\
\dot{Z}_k &= U^{(y)}(t) X_k + U^{(x)}(t) Y_k - b Z_k,
\end{align*}$$

where $U^{(\xi)}(t) = \left(\xi_i + \xi_j\right)/2$ for $\xi = x, y, z$ are the corresponding sum variables. As in (3), the auxiliary stability system (6) corresponds to the differences between the nodes connected by an $x$ edge, and (7) corresponds to the differences between nodes coupled via a $y$ edge. If the connection layers overlap and the same nodes are connected through both $x$ and $y$ edges, then the auxiliary systems (6) and (7) should be applied to the corresponding $x$ and $y$ edges independently. Their contributions will then appear in the general stability conditions (3) for $c_k$ and $d_k$ for the same edge.

The auxiliary systems (6) and (7) are used to yield global stability conditions for the analytical criterion (3), where bounds on the sum differences $U^{(x)}$, $U^{(y)}$, and $U^{(z)}$ can be placed similar to the analysis of synchronization in single-layer networks [30]. A rigorous analysis of global stability of (6) and (7) will be performed in a more technical publication.

Here, we take a more practical route towards developing a numerically-assisted approach which simplifies the evaluation of the stability of (6) and (7). We linearize both systems (6) and (7) by making the differences $X_k$, $Y_k$, and $Z_k$ infinitesimal and replacing $U^{(x)} = x(t)$, $U^{(y)} = y(t)$, and $U^{(z)} = z(t)$, where $(x(t), y(t), z(t))$ is the synchronous solution governed by the uncoupled Lorenz oscillator. The linearized stability systems (6) and (7) take the form:

$$\begin{align*}
\dot{X}_k &= \sigma(Y_k - X_k) - (a_x + \alpha^x_k) X_k, \\
\dot{Y}_k &= (r - z(t)) X_k - Y_k - x(t) Z_k + \omega_x b^{int}_k Y_k, \\
\dot{Z}_k &= y(t) X_k + z(t) Y_k - b Z_k,
\end{align*}$$

$$\begin{align*}
\dot{X}_k &= \sigma(Y_k - X_k) + \omega_y b^{int}_k X_k, \\
\dot{Y}_k &= (r - z(t)) X_k - Y_k - x(t) Z_k - (a_y + \alpha^y_k) Y_k, \\
\dot{Z}_k &= y(t) X_k + z(t) Y_k - b Z_k,
\end{align*}$$

Notice that $\alpha^x_k$ [$\alpha^y_k$] must be large enough to stabilize (8) [9] in the presence of $\omega_x b^{int}_k$ [$\omega_y b^{int}_k$]. While $a_x$, $a_y$, $\omega_x$, and $\omega_y$ are chosen and fixed in Step 1, the traffic load $b^{int}_k$ on a given edge $k$ (which will be determined in Step...
controls the choice of $\alpha^x_k$ and $\alpha^y_k$. Thus, if the edge is highly loaded, then the contribution of the positive term \( +\omega_x b^\text{int}_k \) in the \( \gamma_k \)-equation of system (8) cannot always be compensated by increasing \(-\alpha^x_k X_k\) in the \( X_k \)-equation. Typically, this happens when the positive term exceeds the proper negative linear terms such as \(-Y_k\) (technically, through a combination of terms in the Routh-Hurwitz criterion). Therefore, the auxiliary system (8) can become unstable, independently of how large the stabilization coefficient \( \alpha^x_k \) is. The same argument relates to the destabilization of the auxiliary system (9) via the positive term \(+\omega_y b^\text{int}_k X_k\).

The complex relationship between these terms in regard to stabilizing (8) and (9) is shown in Fig. 3. Notice the coefficient \( \beta \) on the \( \beta \omega_x b^\text{int}_k \) and \( \beta \omega_y b^\text{int}_k \) axes. Because of the Cauchy-Schwarz inequality used in the proof, \( b^\text{int}_k \) provides an overestimate for the terms added to the auxiliary system and the scaling factor \( \beta \) is added to compensate for this overestimate. The diagrams of Fig. 3 confirm the existence of threshold values for \( \beta \omega_x b^\text{int}_k \) and \( \beta \omega_y b^\text{int}_k \) such that even infinitely large values of \( \alpha^x \) and \( \alpha^y \) cannot compensate for the caused instability and stabilize systems (8) and (9).

It is important to emphasize that the stability diagrams of Fig. 3, which later will be used for predicting thresholds in large networks, need to be calculated through simulation of two three-dimensional systems (8) and (9) driven by the synchronous solution \((x(t), y(t), z(t))\). Here, the terms \( \beta \omega_x b^\text{int}_k \) and \( \beta \omega_y b^\text{int}_k \) are treated as single, aggregated control parameters. Therefore, the threshold value for \( \alpha^x_k \) \( \alpha^y_k \) required to stabilize the auxiliary system (8) \( \alpha^y_k \) for a given edge \( k \) with \( b^\text{int}_k \) can simply be read off from Fig. 3. To better quantify this dependence it is used in predicting synchronization thresholds in networks of Lorenz oscillators, we approximate the stability boundary in Fig. 3(a) by the exponential function

\[
\alpha_x = 0.4645 \exp \left( 2.408/\beta \omega_x b^\text{int}_k \right).
\]  

The stability diagrams of Fig. 3 along with Fig. 2 account for the role of the individual oscillators composing the networks and the way these oscillators are coupled (through \( x \) and \( y \) coupling) in the stability of synchronization. These diagrams represent an analog of the master stability function in single-layer networks [23] and help in solving, once and for all, the question of stability for synchronization in two-layer networks involving the Lorenz oscillator through the criterion (3), where the role of multilayer network topologies is assessed via the calculation of pure graph theoretical quantities as shown in the next step.

**Step 3: Calculate traffic loads \( b^x_k \), \( b^y_k \), and \( b^\text{int}_k \).**

This calculation is similar to that of the connection graph method for single-layer networks [30], except that the traffic load should be partitioned into three groups: intralayer traffic loads \( b^x_k \) and \( b^y_k \) within the \( x \) and \( y \) layer, respectively, and interlayer traffic load \( b^\text{int}_k \) between the layers. To do so, we first choose a set of paths \( \{P_{ij}\} \ i, j = 1, ..., n, \ j > i \}, one for each pair of vertices \( i, j \), and determine their lengths \( |P_{ij}| \), the number of edges in each \( P_{ij} \). Then, for each edge \( k \) of the \( x \) \( y \) layer graph, we calculate the sum \( b^\text{int}_k \) of the lengths of all \( P_{ij} \) that contain both \( x \) and \( y \) edges and pass through \( k \). These constants depend on the choice of the paths \( P_{ij} \). Usually, one uses the shortest path from vertex \( i \) to vertex \( j \). Sometimes, however, a different choice of paths can lead to lower bounds [34].

We use the six-node multilayer network of Fig. 1(b) as an example for calculating \( b^x_k \), \( b^y_k \), and \( b^\text{int}_k \). To compute all of the paths that pass through a given edge, it is recommended that the reader algorithmically finds the shortest path between every pair of oscillators, and take note of the paths that go through edge \( k \) and differentiate the paths that entirely belong to only the \( x \) or \( y \) layers...
and the ones that contain a combination of $x$ and $y$ edges. As a result, we can find each edge’s traffic loads as follows:

$$b_{12}^1 = |P_{12}| + |P_{13}| + |P_{14}| + |P_{15}| + |P_{16}| = 1 + 2 + 3 + 3 + 4 = 13,$$

$$b_{23}^1 = |P_{23}| + |P_{24}| + |P_{25}| + |P_{26}| = 20,$$

$$b_{34}^1 = |P_{34}| + |P_{35}| + |P_{36}| = 6,$$

$$b_{45}^1 = |P_{45}| + |P_{46}| = 5,$$

$$b_{56}^1 = |P_{56}| = 1,$$

$$b_{12}^2 = |P_{12}| + |P_{13}| + |P_{14}| + |P_{15}| + |P_{16}| = 1 + 2 + 3 + 3 + 4 = 13,$$

$$b_{23}^2 = |P_{23}| + |P_{24}| + |P_{25}| + |P_{26}| + |P_{34}| = 15,$$

$$b_{34}^2 = |P_{34}| + |P_{35}| + |P_{36}| = 6,$$

$$b_{45}^2 = |P_{45}| + |P_{46}| = 5,$$

$$b_{56}^2 = |P_{56}| = 1,$$

$$b_{12}^3 = |P_{12}| + |P_{13}| + |P_{14}| + |P_{15}| + |P_{16}| = 1 + 2 + 3 + 3 + 4 = 13,$$

$$b_{23}^3 = |P_{23}| + |P_{24}| + |P_{25}| + |P_{26}| + |P_{34}| = 15,$$

$$b_{34}^3 = |P_{34}| + |P_{35}| + |P_{36}| = 6,$$

$$b_{45}^3 = |P_{45}| + |P_{46}| = 5,$$

$$b_{56}^3 = |P_{56}| = 1.$$

(11)

Note that the maximum interlayer traffic load on this network is fairly low and due to our choice of paths is $b_{56}^{1st} = 1$ although it could have also been minimized to zero, provided that all paths to node 6 bypass edge 56.

At the same time, the interlayer traffic load in the network of Fig. 1(c) is significantly higher since there are no alternatives to go around the “bottleneck” edge 2-3 when traveling from nodes 1 and 2 to nodes 4, 5, and 6. For the same choice of shortest paths, we get $b_{23}^{1st} = 19$. The remaining $b_{12}^*, b_{23}^*$ for the network of Fig. 1(c) can be calculated similarly to (11).

**Step 4: Putting pieces together to solve the puzzle.** Given the stability diagram of Fig. 3 with abrupt threshold dependences of $\alpha_k^*$ and $\alpha_k^b$ on increasing interlayer traffic load $b_k^*$, the effect of synchrony breaking when a highly loaded $x$ edge is replaced with a better pairwise stabilizing $y$ (see Sec. III) is no longer a puzzle and directly follows from the application of our stability method. Actually, in a historical retrospective, we first developed the general method that revealed this and other highly counterintuitive effects due to the multilayer structure and then constructed the network examples. To make the presentation more appealing before it becomes too technical, we have decided to put forward the motivating example. As our exhaustive study of various network configurations suggests, we hypothesize that six-node networks of Fig. 1 are minimum size networks of Lorenz oscillators that exhibit the synchrony breaking phenomenon.

To test the predictive power of our approach with the constants identified in Steps 1-3, we perform a systematic study of how one edge replacement, in which we replace only one $x$ edge in the single-layer, $x$-coupled network of Fig. 1(a) with a $y$ edge, affects synchronization. The edge replacement is performed in the order of the increasing interlayer traffic load on this edge, $b_k^{int}$. After computing the coupling threshold required to synchronize the new network, this edge reverts back to being an $x$ edge. This results in multiple networks of five $x$ edges and one $y$ edge. The two multilayer networks of Fig. 1(b) and Fig. 1(c) with the drastically different synchronization properties are two instances of this replacement process.

Fig. 1(d) presents the actual synchronization threshold values (blue solid line), the interlayer traffic loads $b_{ij}^{int}$ (black line) calculated similarly to (11), and the threshold values predicted by the numerically-assisted criterion (3) with constants $a_x = 7.20$, $a_y = 2.63$, $\omega_x = 5.00$ and $\omega_y = 0.50$ chosen above. The constants $\alpha_2^*$ and $\alpha_2^b$ are read off from the diagrams of Fig. 3(a) and Fig. 3(b), respectively. As the stability system (8) is much more sensitive to the changes in $b_k^{int}$ than (9) (cf. the onset of instability in Figs. 3(a-b)), the threshold values for $c_{ij}$ in the criterion (3) for the $x$ layer largely dominate over $d_{ij}$. Thus, since the synchronization threshold for the entire network (2) with uniform coupling $c = d$ is defined by the maximum of the thresholds $c_{ij}$ or $d_{ij}$ for each edge of the multilayer graph, the maximum threshold values predicted by the method and depicted in Fig. 1(d) are the ones corresponding to $x$ edges with coupling $c$. These threshold values are calculated using the numerically assisted modification of (3)

$$c > \max_k \left\{ c_k = \frac{1}{n} \left[ \frac{\gamma a_x \cdot b_k^* + \beta \omega_x \cdot b_k^{int} + \alpha_k^x (\beta \omega_y b_k^{int})}{\text{max}} \right] \right\},$$

where $\alpha_k^*$ is defined by the stability diagram of Fig. 3(a) via the approximating function (10) for each edge $k$, and $a_x$, $\omega_x$, and $b_k^{int}$ are determined in Steps 1-3. Notice the presence of an additional scaling factor $\gamma$ which is chosen to compensate for the conservative nature of the connection graph stability method when the network is entirely coupled through the $x$ variable. $\gamma$ scales down the term $\alpha_k^*$ to match the coupling needed to synchronize the six-node network of Fig. 1(a) with only $x$ edges. The scaling factor $\beta$ is then chosen for the network of Fig. 1(b) with the lowest $b_k^{int} = 4$ to match the actual synchronization threshold and then kept constant for predicting the thresholds in the other six-node networks with one replaced $x$ edge. Figure 1(d) shows that the predicted thresholds are fairly close to the actual ones, and the criterion (12) correctly predicts an increase or decrease of the coupling threshold for each six-node network and ultimately predicts synchrony break for the network with the replaced $x$ edge 2-3.

As Fig. 1(d) indicates that when lightly loaded edges (edges with fewer chosen paths passing through them) are replaced, the effect on the synchronization stability is fairly small. As discussed in the description of the motivating example, the replacement of $x$ edge 5-6 with a $y$ edge improves synchronization by slightly lowering the synchronization threshold. According to our stability criterion (3), this happens due to a slight decrease in the traffic load on the bottleneck edge $b_{23}^{int}$ (see (11)), compared to the original network of Fig. 1(a) with all $x$ edges where one additional path $P_{16}$ goes through edge 2-3. As a result, it decreases the contribution of the dominating term $a_x b_k^*$ in (3). At the same time, the contribution of the other factors $\omega_x b_k^{int} + \alpha_k^x$ remain insignificant, especially due to the fact that $\alpha_k^x$ still lies on a flat part of the approximating curve (10) before this exponential curve takes off at larger values of $b_k^{int}$. On the other hand, such a replacement of the bottleneck node 2-3 in the network of Fig. 1(c) significantly increases the intralayer traffic load $b_k^{int}$, requiring infinitely large $\alpha_k^{23}$ to stabilize the stability system (8) and causing synchrono-
VI. LARGER NETWORKS

To demonstrate that similar synchrony breakdown phenomena occur in larger networks and can be effectively predicted by our method, we consider a 20-node network of Lorenz (and then double-scroll) oscillators described in Fig. 4(a). The network is initially coupled entirely through the $x$ variable. To test our prediction that replacing edges with a high traffic load can make the network unsynchronizable, we index the edges according to their $b_k^x$. Edges similar to edge 10-12 have very few paths that pass through them, and subsequently have a low $b_k^x$ (and in turn, $b_k^{int}$, shown as the black curve in Fig. 4(c)). We successively replace $x$ edges (denoted by black edges in Fig. 4(a)) with $y$ edges (denoted by gray edges in Fig. 4(b)), according to this ordering until the network is completely connected through $y$ edges. The values of $b_k^{int}$ range from 0 (for edge 10-12 with edge ranking index 1 (see Fig. 4(c)), bypassed by all chosen interlayer paths) to 100 – 400 for highly loaded edges (for example, for edge 3-5 for which every path from node 3 of the $x$ layer graph to any other node in the $y$-layer must pass through it).

A. Networks of Lorenz oscillators

The coupling necessary to synchronize the $x$-coupled Lorenz network (2) described in Fig. 4(a) is $c \approx 86.95$. As outlying, low traffic edges are replaced with $y$ edges, there is almost no effect on the threshold for the coupling strength required to synchronize the network, evidenced by the lack of change in the actual coupling threshold for the first eight edges replaced in Fig. 4(c). As successively more loaded edges are replaced in the network (indicated by the dramatic increase in $b_k^{int}$), the network becomes more difficult to synchronize, until edge 13 (edge 3-5 which is depicted in red in Fig. 4(b)) is replaced. After which, synchronization is no longer feasible for the network for any additional edge replacement, until edge 25 (edge 14-15 in Fig. 4). Replacing edge 25, shown in Fig. 4(b) corresponds to finishing the successive edge replacement process, and results in a graph identical to the one in Fig. 4(a), but in which all of the edges represent $y$ coupling (gray) instead of $x$ coupling (black). This reinforces our traffic load predictions for the breakdown of synchrony in two ways: (i) after enough highly loaded edges are replaced (even with a normally favorable coupling type), the network can no longer synchronize for any coupling strength, (ii) replacing only one edge that is very highly loaded can make the network unsynchronizable, evidenced by the network having no synchronizing coupling value even when all but one edge has been replaced (see edge index 24 in Fig. 4(c)).

As in the six-node example of Fig. 1(a), we have obtained a good fit in Fig. 4(c) which only focuses on placing the stability conditions on $x$ edges in (3), because the stability term $\alpha_x$ required to stabilize the $x$ stability system (8) must be significantly higher than $\alpha_y$ in the $y$ stability system (9) (compare Figs. 3(a-b)). We use the same criterion (12) with the same constants $\alpha_x$, $\omega_x$, $\omega_y$, $\alpha_y$, $\beta$.
and \( \alpha_k \) to predict the synchronization threshold and only need to identify the traffic loads \( b_k^{\text{int}} \) and the scaling factors \( \gamma \) and \( \beta \) for a better fit, once and for all.

In contrast to the six-node network example where traffic load \( b_k^{\text{int}} \) can be easily calculated by hand as in (11), computing \( b_k^{\text{int}} \) for the 20-node or larger networks is a laborious task which was performed by an algebraic algorithm, implemented as MATLAB code and given in the Supplement. While the values of \( b_k^{\text{int}} \) heavily depend on the choice of paths from one node to another, our algorithm uses the natural choice of the shortest paths, computed via Dijkstra’s algorithm [39]. Optimizing the choices of not necessarily shortest paths that distribute traffic loads on edges more equally may yield even better predictions and fits.

B. Networks of double scroll oscillators

To illustrate the generality of synchrony break phenomenon when “good” but highly-loaded links go “bad”, we apply our numerically-assisted method to networks (1), comprised by chaotic double-scroll oscillators [38]

\[
\begin{align*}
\dot{x}_i &= \kappa(y_i - x_i - h(x)) + \sum_{j=1}^{n} c_{ij} x_j, \\
\dot{y}_i &= x_i - y_i + z_i + \sum_{j=1}^{n} d_{ij} y_j, \\
\dot{z}_i &= -\lambda y_i - \mu z_i, \quad i = 1, \ldots, n,
\end{align*}
\]

with

\[
h(x) = \begin{cases} 
  m_1(x + 1) - m_0 & x < -1 \\
  m_0 x & -1 \leq x \leq 1 \\
  m_1(x - 1) + m_0 & x > 1
\end{cases}
\]

and standard parameters \( \kappa = 10, m_0 = -1.27, m_1 = -0.68, \lambda = 15, \) and \( \mu = 0.038. \)

Similarly to networks of Lorenz oscillators (2), a pair of double-scroll oscillators (13) can be synchronized through either the \( x \) or \( y \) variable, and the minimum coupling strength required for synchronization in a two-node \( y \)-coupled network, \( d^* = 1.16 \) is much lower than the coupling threshold in the two-node \( x \)-coupled network,

\[
c^* = 5.94.
\]

In Fig. 5, we apply our method to predict the synchronization thresholds in the 20-node network of Fig. 4 as in the same network of Lorenz oscillators. When successively replacing \( x \) edges in the network, there is initially a decrease in the coupling threshold for synchronization, when peripheral edges or edges in highly connected regions of the graph with low traffic loads \( b_k^{\text{int}} \) are replaced with more favorable \( y \) edges that provide better pairwise convergence to synchronization. Then, as with the network of Lorenz oscillators, when edge 13 (edge 3-5) is replaced with a \( y \) edge, synchronization is no longer attainable. Synchronization then returns when the entire \( x \)-coupled network has been replaced with \( y \) edges. Notably, the synchrony break occurs at the same edge as in the network of Lorenz oscillators, suggesting that critical edges whose replacement hampers synchronization are mainly controlled by the network multilayer topology rather than the individual properties of the intrinsic oscillators, provided that the oscillators possess similar synchronization properties as the Lorenz and double scroll oscillators.

The solid curve in Fig. 5 displays the synchronization thresholds, calculated via the stability criterion (12) with \( a_x = 5.94 \times 2 = 11.88, \omega_x = 2.0, \beta = 0.0041, \gamma = 0.282, \) and the approximating function \( a_x = 1.556 \exp(3.711 \beta \omega_y b_k^{\text{int}}) \) with the same traffic loads \( b_k^{\text{int}} \) shown in Fig. 4. This approximating function is obtained from a stability diagram for coupled double scroll-oscillators which is computed similarly to Fig. 3 and displays a similar threshold effect as in Fig. 3 [not shown]. As in the Lorenz oscillator case, the auxiliary stability system (4) for \( a_x \) is much more sensitive to increasing \( b_k^{\text{int}} \) than the stability system (5) for \( a_y \), therefore one can only evaluate the stability condition (12) for the \( x \) coupling \( c \) to identify a bottle-neck for the synchronization threshold in the entire network.

Going back to the puzzle example, we have also performed a similar analysis of the six-node network of Fig. 1 where the Lorenz oscillators are replaced with the double-scroll oscillators [not shown]. Remarkably, this analysis indicates the same qualitative phenomena when the replacement of the lightly loaded edge 5-6 slightly lowers the synchronization threshold from \( c = 13.57 \) in the original \( x \)-coupled single-layer network of Fig. 1(a) to \( c = 13.36 \), and predicts the breakdown of synchrony when edge 2-3 is replaced as in the Lorenz network.
We have also simulated series of other 20-node networks (2) and then networks (13) where all oscillators were connected via $x$ layer graphs, whereas the $y$ coupling only connected some of the oscillators. In contrast to the networks of Fig. 1 and Fig. 4 where the critical highly-loaded links separate the network into disjoint $x$ and $y$ graph components, these networks do not show the effect of synchrony breaking as any pair of nodes is coupled directly or indirectly via the $x$ graph such that the coupling strengths $c$ can be made strong enough to stabilize synchronization. However, the synchronization thresholds in such networks depend on the location of added $y$ edges in a nonlinear fashion. In support of this claim, we draw the reader’s attention to the six-node example of Fig. 1(b) where the $x$ graph connects all six nodes and the replacement of edge 5-6 with an $y$ edge lowers the synchronization threshold. On the contrary, the replacement of the $x$ edge 3-5 with an edge $y$, which still preserves the connectedness of the $x$ graph, increases the synchronization threshold, as predicted by the method (see Fig. 1(c)). The 20-node networks with connected $x$ graphs yield similar effects. To avoid repetition, these results are not shown.

VII. CONCLUSIONS

While the study of synchronization in multilayer dynamical networks has gained significant momentum, the general problem of assessing the stability of synchronization as a function of multilayer network topology remained practically untouched due to the absence of general predictive methods. The existing eigenvalue methods, including the master stability function [23], which effectively predict synchronization thresholds in single-layer networks cannot be applied to multilayer networks in general. This is due to the fact that the connectivity matrices corresponding to two or more connection layers do not commute in general, and therefore, the eigenvalues of the connectivity matrices cannot be used. Therefore, synchronization in multilayer networks is usually studied on a case by case basis either via (i) full-scale simulations of all transversal Lyapunov exponents of the $(n-1) \times s$-dimensional system of variational equations [55], where $n$ is the network size and $s$ is the dimension of the intrinsic node dynamics, or more effectively via (ii) simultaneous block diagonalization of the connectivity matrices [54] which in some cases can reduce the problem of assessing synchronization in a large network to a smaller network which, however, contains positive and negative connections, including self loops such that the exact role of multilayer network topology and the addition or exchange of edges remains unclear.

In this paper, we have made a breakthrough in understanding synchronization properties of multilayer networks by developing a predictive method, called the Multilayer Connection Graph method, which does not rely on calculations of eigenvalues of the connectivity matrices, and therefore can handle multilayer networks. Originated from the connection graph method for synchronization in single-layer networks [30], our method combines stability theory with graph theoretical reasoning. Two key ingredients of the method are (i) the calculation of stability diagrams for the auxiliary $s$-dimensional system which indicate how the dynamics of the given oscillator comprising the network can be stabilized via one variable corresponding to one connection layer when an instability is introduced via the other variable from another connection layer and (ii) the calculation of traffic loads via a given edge on the multilayer connection graph. All together, these quantities allow for predicting the synchronization threshold and identify critical links that control synchronization in the original, potentially large, multilayer network.

Using the method, we have discovered striking, highly unexpected phenomena not seen in single-layer networks. In particular, we have shown that replacing a link with a light interlayer traffic load by a stronger pairwise converging coupling via another layer may improve synchronizability, as one would expect. At the same time, such a replacement of a highly loaded link may essentially worsen synchronizability and make the network unsynchronizable, turning the pairwise stabilizing “good” link into a destabilizing connection (“bad” link). The critical links whose replacement can lead to synchronization break are typically the ones that connect the layers such the oscillators from two layers become coupled through the intrinsic, nonlinear equations of the individual oscillator that correspond to a “relay” node passed by the only path from one layer to the other. As a result, the intrinsic dynamics of the individual node oscillator plays a pivotal role in the stability of synchronization. In this paper, we have limited our attention to Type I oscillators such as the Lorenz and double scroll oscillators that yield global synchronization that remains stable in a single-layer network once the coupling exceeds a critical threshold. Remarkably, when used in a multilayer network, both oscillators have indicated similar synchronization properties, suggesting that the location of critical edges in the considered network may remain unchanged for other Type I oscillators. While our rigorous method for proving global synchronization is only applicable to Type I oscillators, its semi-analytical version can be modified to handle Type II networks, including multilayer networks of Rössler systems [22]. This modification is a subject of future study.

To gain insight into the determining factors for the emergence of synchrony breaking, without potential confounds associated with the interplay between multiple layers and direction of links, we have considered two-layer undirected networks of identical oscillators. However, the extension of our results to multiple layers, directed networks and non-identical oscillators is fairly straightforward and will be reported elsewhere. In particular, the extension of our method to directed networks can be performed by adapting the generalized connection graph
method [31, 60] for single-layer directed networks, where directed edges are symmetrized and assigned additional weights according to the mean node unbalance.

Our method can also be modified to handle multilayer neuronal networks connected via electrical, excitatory, and inhibitory synapses which exhibit a number of counterintuitive synergistic effects: when (i) the addition of pairwise repulsive inhibition to single-layer excitatory networks can promote synchronization [43] and (ii) combined electrical and inhibitory coupling can induce synchronization even though each coupling alone promotes an anti-phase rhythm [61]. Our method promises to allow an analytical treatment of these effects in large neuronal networks which has been impaired by the absence of rigorous methods that can handle excitatory, inhibitory, and electrical neuronal circuitries simultaneously.

VIII. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (USA) under Grant No. DMS-1616345 and the U.S. Army Research Office under Grant No. W911NF-15-1-0267.

IX. APPENDIX: THE PROOF

In this appendix we derive the Multilayer Connection Graph method and prove Theorem 1. Our goal is to derive the conditions of global asymptotic stability of the synchronization manifold M in the system (1). To achieve this goal and develop the stability method, we follow the steps of the proof of the connection graph method [30]. The concept is similar, up to a certain step where a new stability argument is used.

In the network model (1) we introduce the difference variable

\[ X_{ij} = x_j - x_i, \quad i, j = 1, ..., n, \] (14)

whose convergence to zero will imply the transversal stability of the synchronization manifold M.

Subtracting the \( i \)-th equation from the \( j \)-th equation in system (1), we obtain the equations for the transversal stability of \( M \)

\[
X_{ij} = F(x_j) - F(x_i) + \sum_{k=1}^{n} \left\{ c_{jk} P X_{jk} - c_{ik} P X_{ik} + d_{jk} L X_{jk} - d_{ik} L X_{ik} \right\}, \quad i, j = 1, ..., n. \] (15)

To obtain the explicit dependence of \( F(x_j) - F(x_i) \) on \( X_{ij} \), we introduce the following vector notation

\[
F(x_j) - F(x_i) = \left[ \int_0^1 DF(\beta x_j + (1-\beta)x_i) d\beta \right] X_{ij},
\]

where \( DF \) is the \( s \times s \) Jacobian matrix of \( F \). This notation is simply a compact form of the mean value theorem, \( f(B) - f(A) = f'(C)(B - A) \), applied to the vector functions \( F(x_j) \) and \( F(x_i) \), where the Jacobian \( DF \) is evaluated at some point \( C \in [x_j, x_i] \).

Therefore, the difference system (15) can be rewritten in the form

\[
X_{ij} = \left[ \int_0^1 DF(\beta x_j + (1-\beta)x_i) d\beta \right] X_{ij} + \sum_{k=1}^{n} \left\{ c_{jk} P X_{jk} - c_{ik} P X_{ik} + d_{jk} L X_{jk} - d_{ik} L X_{ik} \right\}, \quad i, j = 1, ..., n. \] (16)

The first term with the brackets yields instability via the divergence of trajectories within the individual, possibly chaotic oscillators. The second (summation) term, which represents the contribution of the network connections, may overcome the unstable term, provided that the coupling is strong enough.

Notice that the stability of system (16) is redundant as it contains all possible \( (n-1)n/2 \) non-zero differences \( X_{ij} \) along with \( n \) zero differences \( X_{ii} = 0 \) which can be disregarded. At the same time, there are only \( n-1 \) linearly independent differences required to show the convergence between \( n \) variables \( X_{ij} \). However, this redundancy property and the consideration of all non-zero \( X_{ij} \) are a key ingredient of our approach which allows for separating the difference variables later in the stability description, without diagonalizing the connectivity matrices.

We strive to find conditions under which the trivial fixed point \( X_{ij} = 0, \quad i, j = 1, ..., n \) of system (16) is globally stable. This amounts to finding conditions for global stability of synchronization in the network (1).

We introduce the following terms \( A_{ij}X_{ij} \), where \( A \) is a \( 3 \times 3 \) matrix, such that

\[
A_{ij} = \begin{cases} 
  a_x P & \text{if oscillators } i \text{ and } j \text{ belong to } x \text{ layer } C \\
  a_y L & \text{if oscillators } i \text{ and } j \text{ belong to } y \text{ layer } D \\
  K & \text{if oscillators } i \text{ and } j \text{ belong to different layers } C \text{ and } D,
\end{cases}
\] (17)

where constants \( a_x, a_y, \omega_x, \omega_y \) are to be determined.

We add and subtract additional terms \( A_{ij}X_{ij} \) with matrix \( A_{ij} \) defined in (17) from the stability system (16) and obtain

\[
X_{ij} = \left[ \int_0^1 DF(\beta x_j + (1-\beta)x_i) d\beta - A_{ij} \right] X_{ij} + A_{ij}X_{ij} + \sum_{k=1}^{n} \left\{ c_{jk} P X_{jk} - c_{ik} P X_{ik} + d_{jk} L X_{jk} - d_{ik} L X_{ik} \right\}. \] (18)

The introduction of the terms \( A_{ij}X_{ij} \) allows for obtaining stability conditions of the trivial fixed point \( X_{ij} = 0, \quad i, j = 1, ..., n \) in two steps. Note that the matrix \(-A\) contributes to the stability of the fixed point and can compensate for instabilities induced by eigenvalues with
nonnegative real parts of the Jacobian $DF$. This can be achieved by increasing parameters $a_x, a_y, \omega_x,$ and $\omega_y$. At the same time, the instability originated from its positively definite counterpart, matrix $+A$, can be damped by the coupling terms through $c_{ij}$ and $d_{ij}$.

**Step I.** We make the first step by introducing the following auxiliary systems for $i, j = 1, \ldots, n$

$$\dot{X}_{ij} = \left[ \int_0^1 DF(\beta x_j + (1-\beta) x_i) \, d\beta - A_{ij} \right] X_{ij}, \quad (19)$$

This system is identical to system (18) where the coupling terms are removed.

$A_{ij}$ can take three different values, depending on whether oscillators $i$ and $j$ both belong to the $x$ or $y$ graphs, or belong to different graphs, for example, if $i$ belongs to the $x$ graph, and $j$ belongs to the $y$ graph (see (17)). Therefore, we have three types of the auxiliary systems

$$\dot{X}_{ij} = \left[ \int_0^1 DF(\beta x_j + (1-\beta) x_i) \, d\beta - a_x P \right] X_{ij}, \quad (20)$$

if $i$ and $j$ both belong to $x$ layer,

$$\dot{X}_{ij} = \left[ \int_0^1 DF(\beta x_j + (1-\beta) x_i) \, d\beta - a_y L \right] X_{ij}, \quad (21)$$

if $i$ and $j$ both belong to $y$ layer,

$$\dot{X}_{ij} = \left[ \int_0^1 DF(\beta x_j + (1-\beta) x_i) \, d\beta - K \right] X_{ij}, \quad (22)$$

if $i$ and $j$ each belong to different layers.

Remarkably, the auxiliary system (20) coincides with the difference system for the global stability of synchronization in a two-oscillator network (1) with only $x$ coupling, where $a_x$ plays the role of the double coupling strength that guarantees the stability (see [30] for a detailed discussion on this relation).

Similarly, the stability of auxiliary system (21) implies global stability of synchronization in the two-oscillator network (1) with only $y$ coupling, where $a_y$ is the double coupling strength of the $y$ connection. Lastly, the stability of auxiliary system (22) guarantees globally stable synchronization in the two-oscillator network with both $x$ and $y$ coupling, where a combination of constants $\omega_x$ and $\omega_y$, present in $K$, is a combination of the double coupling strengths of $x$ and $y$ connections that is sufficient to induce stable synchronization in the two-oscillator $x$ and $y$ coupled network.

Therefore, our immediate goal is to find upper bounds on the values of $a_x$, $a_y$, and $\omega_x$ and $\omega_y$ that make the origin of the auxiliary systems (20)-(22) stable. This amounts to proving global synchronization in the three $x$-, $y$-, and $(x,y)$-coupled networks that are composed of two oscillators. As only Type I oscillators can be globally synchronized, our approach based on the calculation of $a_x$, $a_y$, is thus limited to this class of oscillators.

The proof of global stability in (20)-(22) and derivation of bounds $a_x$, $a_y$, and $\omega_x$ and $\omega_y$ involves the construction of a Lyapunov function along with the assumption of the eventual dissipateness of the coupled system. Therefore, before advancing with the study of larger networks (1), one has to prove that globally stable synchronization in the simplest $x$-, $y$-, and $(x,y)$-coupled, two-oscillator networks is achievable. The bound $a_x$ for $x$-coupled Lorenz oscillators was given in [30]. Upper bounds for $a_y$, $\omega_x$, and $\omega_y$ can be derived similarly.

Having obtained the bounds $a_x$, $a_y$, and $\omega_x$ and $\omega_y$, and therefore proving the stability of the auxiliary systems (20)-(22), we can now take the second step in analyzing the full stability system (18).

**Step II.** The bounds $a_x$, $a_y$, and $\omega_x$ and $\omega_y$ that stabilize the auxiliary systems (20)-(22) reduce the stability analysis of system (18) to the following equations by excluding the term in brackets

$$\dot{X}_{ij} = A_{ij} X_{ij} + \sum_{k=1}^n \left\{ c_{jk} P X_{jk} - c_{ik} P X_{ik} + d_{jk} L X_{jk} - d_{ik} L X_{ik} \right\}, \quad i, j = 1, \ldots, n. \quad (23)$$

Note that the positive term $A_{ij} X_{ij}$, which contains the upper bounds $a_x$, $a_y$, and $\omega_x$ and $\omega_y$ is destabilizing and must be compensated for by the coupling terms. To study the stability of (23) we introduce a Lyapunov function of the form

$$V = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n X_{ij}^T : I \cdot X_{ij}, \quad (24)$$

where $I$ is an $s \times s$ identity matrix.

Its time derivative with respect to system (23) becomes

$$\dot{V} = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n X_{ij}^T A_{ij} X_{ij} - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \left\{ c_{jk} X_{jk}^T P X_{jk} - c_{ik} X_{ik}^T P X_{ik} \right\} - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \left\{ d_{jk} X_{jk}^T L X_{jk} - d_{ik} X_{ik}^T L X_{ik} \right\}. \quad (25)$$

We need to demonstrate the negative semi-definiteness of the quadratic form $\dot{V}$. As $(X_{ii}^2 = 0, X_{ij}^2 = X_{ji}^2)$, the first sum simplifies to

$$S_1 = \sum_{i=1}^{n-1} \sum_{j=i+1}^n A_{ij} X_{ij}^2. \quad (26)$$

This sum is always positive definite and its contribution must be compensated for by the second and third sums

$$S_2 = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \left\{ c_{jk} X_{jk}^T P X_{jk} - c_{ik} X_{ik}^T P X_{ik} \right\},$$

$$S_3 = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \left\{ d_{jk} X_{jk}^T L X_{jk} - d_{ik} X_{ik}^T L X_{ik} \right\}. \quad (27)$$
Due to the coupling symmetry, the two terms in both $S_2$ and $S_3$ can be made identical by exchanging the indices $i$ by $j$ in the second terms such that

$$S_2 = -\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{jk} X_{jk}^T IP X_{jk},$$
$$S_3 = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} d_{jk} X_{jk}^T ILX_{jk}. \quad (28)$$

Taking into account that $X_{jj} = 0$, we obtain

$$S_2 = -\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{jk} (X_{ji}^T + X_{kj}^T) IP X_{jk},$$
$$S_3 = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} d_{jk} (X_{ji}^T + X_{kj}^T) ILX_{jk}. \quad (29)$$

Again, exchanging $j$ and $k$ in the second terms of $S_2$ and $S_3$ and implying the symmetries of coupling $c_{jk} = c_{kj}$ and $d_{jk} = d_{kj}$, we obtain

$$S_2 = -\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{jk} (X_{ji}^T + X_{kj}^T) IP X_{jk},$$
$$S_3 = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} d_{jk} (X_{ji}^T + X_{kj}^T) ILX_{jk}. \quad (30)$$

Since $X_{ji}^T + X_{kj}^T = [x_i^T - x_j^T + x_k^T - x_i^T] = X_{jk}^T$, we obtain

$$S_2 = -\sum_{i=1}^{n} \sum_{j=1}^{n} n c_{jk} X_{jk}^T IP X_{jk},$$
$$S_3 = -\sum_{i=1}^{n} \sum_{j=1}^{n} n d_{jk} X_{jk}^T ILX_{jk}. \quad (31)$$

Returning to the derivation of the Lyapunov function (23) and combining the sums $S_1$, $S_2$ and $S_3$ yields the condition which guarantees that $\dot{V} \leq 0$:

$$S_1 + S_2 + S_3 = \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij}^T |A_{ij} - n c_{ij} P - n d_{ij} L| X_{ij} < 0. \quad (32)$$

The most remarkable property of this condition is that we are able to eliminate the cross terms and formulate the condition in terms of $X_{ij}$. This is because we chose to consider the redundant system with all possible differences $X_{ij}$, including linearly dependent ones.

The condition (32) finally transforms into

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \left( c_{ij} X_{ij}^T IPX_{ij} + d_{ij} X_{ij}^T ILX_{ij} \right) > \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij}^T IA_{ij} X_{ij}. \quad (33)$$

Notice that the left-hand side (LHS) of this inequality contains only the differences $X_{ij}$ between the oscillators that belong to the edges on the connection graphs $C$ and $D$: the first term on the LHS corresponds to the $x$ layer and the second term is defined by the edges of the $y$ layer. At the same time, the variables on the right-hand side (RHS) of (33) correspond to all possible differences between pairs of oscillators that might or might not be defined by edges of the connection graphs. Hence, to get rid of the presence of the differences $X_{ij}$ and therefore find the conditions explicit in the parameters of the network model (1), we express the differences on the RHS via the differences on the LHS such that we will be able to cancel them.

So far, we have closely followed the steps in the derivation of the connection graph method [30] for single-layer networks. The inequality (33) is similar to that of the connection graph method, except for the presence of the second term on the RHS and a modified matrix $A_{ij}$. A new non-trivial observation, however, is that the total number of oscillators, $n$, in the network (1), composed of two connection layers, appears as a factor in both sums on the RHS, corresponding to the $x$ and $y$ graphs, even though each graph itself may connect fewer oscillators. The stability argument which follows drastically differs from that of the connection graph method.

Denote on the LHS (33), the differences $X_{ij}$ corresponding to edges of the $x$ graph by $X_k$, $k = 1, ..., m$ and the differences $X_{ij}$ corresponding to edges of the $y$ graph by $Y_k$, $k = 1, ..., l$. Recall that $m$ and $l$ are the number of edges on the $x$ and $y$ graphs, respectively. In addition, let $X_k$ be a scalar from the vector $X_k$ which indicates the scalar difference between $x_i$ and $x_j$, corresponding to an edge on the $x$ graph. Similarly, let $Y_k$ be a scalar from the vector $Y_k$, defined by the corresponding $y_i$ and $y_j$. Using these notations, the differences $X_{ij}$ on the RHS will now define the scalars $X_{ij} = x_j - x_i$ and $Y_{ij} = y_j - y_i$. If the inequality (33) is satisfied in terms of $x_i$ and $y_i$, and the via the scalar differences $X_k$ and $Y_k$, then it will also be satisfied for the remaining scalar $z_i$. Recall that $(x_1, y_1, z_1)$ are the scalar coordinates of the individual oscillator, composing the network (1).

Using this notation, we can rewrite (33) as follows

$$n \sum_{k=1}^{m} c_k X_k^2 + n \sum_{k=1}^{l} d_k Y_k^2 > a_x n \sum_{i=1}^{n-1} \sum_{j>i}^{n} X_{ij}^2 + a_y n \sum_{i=1}^{n-1} \sum_{j>i}^{n} Y_{ij}^2 + \sum_{i=1}^{n-1} \sum_{j>i}^{n} (\omega_x X_{ij}^2 + \omega_y Y_{ij}^2), \quad (34)$$

where $c_k = c_{ik,jk}$ and $d_k = d_{ik,jk}$. Here, the RHS of (34) has three terms, obtained by splitting the difference variables into three groups, according to the coefficients of $A_{ij}$ (cf. (33) and (20)-(22)). The first sum on the RHS is composed of the differences that belong to the $x$ graph $G$, the second sum corresponds to the $y$ graph $D$, whereas the third sum identifies the differences between the oscillators which belong to different graphs such that $i \in C$ and $j \in D$ or vice versa.

To recalculate the difference variables of the RHS via the variables $X_k$ and $Y_k$, we should first choose a path
from oscillator $i$ to oscillator $j$ for any pair of oscillators $(i, j)$. We denote this path by $P_{ij}$. Its path length $|P_{ij}|$ is the number of edges comprising the path. The important property of the path $P_{ij}$ is that if, for example, it passes through oscillators with indices 1, 2, 3, and 4, then the corresponding difference $X_{14} = x_4 - x_1 = (x_4 - x_3) + (x_3 - x_2) + (x_2 - x_1) = X_{12} + X_{23} + X_{34}$, where the differences $X_{12}, X_{23}$, and $X_{34}$ correspond to the edges and the path length $|P_{14}| = 3$.

The choice of paths is not unique. We typically choose a shortest path between any pair of $i$ and $j$; however, a different choice of paths can yield closer estimates, as discussed in [34] for single-layer networks.

Once the choice of paths is made, we stick with it and start recalculating the difference variables on the RHS of (34) via

$$X_{ij} = \left(\sum_{k \in P_{ij}} X_k\right)^2 \leq |P_{ij}| \sum_{k \in P_{ij}} X_k^2,$$

$$Y_{ij} = \left(\sum_{k \in P_{ij}} Y_k\right)^2 \leq |P_{ij}| \sum_{k \in P_{ij}} Y_k^2,$$

where once again $|P_{ij}|$ indicates the length of the chosen path from oscillator $i$ to oscillator $j$ along the connection graph, combined of the $x$ and $y$ graphs. At this point, we do not differentiate between paths containing only $x$ or $y$ edges, but we have to consider interlayer paths when necessary.

Applying this idea to each difference variable on the RHS of (34), we obtain the following condition

$$n \sum_{k=1}^{m} c_k X_k^2 + n \sum_{k=1}^{l} d_k Y_k^2 > \sum_{k=1}^{m} \left[ \omega_x b^x_k + \omega_x b^{int}_k \right] X_k^2 + \sum_{k=1}^{l} \left[ \omega_y b^y_k + \omega_y b^{int}_k \right] Y_k^2,$$

where $b^x_k = \sum_{j > i; k \in P_{ij} \in C} |P_{ij}|$ is the sum of the lengths of all chosen paths which belong to the $x$ graph $C$ and go through a given $x$ edge $k$. Similarly, $b^{int}_k = \sum_{j > i; k \in P_{ij} \in D} |P_{ij}|$ is the sum of the lengths of all chosen paths which belong to the $y$ graph $D$ and go through a given $y$ edge $k$. Finally, $b^{int}_k$ is the sum of the lengths of all chosen paths which contain $x$ and $y$ edges and belong to the combined, two-layer $xy$ graph and go through a given edge $k$ which may be an $x$ or $y$ edge.

Note that the two sums on the LHS of (36) correspond to the first sums on the RHS. In the simplest case where both $C$ and $D$ graphs are connected such that each graph couples all $n$ oscillators, $b^{int}_k$ can always be set to 0, since there are always paths between any two nodes that entirely belong to either the $x$ or $y$ graph. As a result, the third and forth sums on the RHS disappear, and we immediately obtain the stability conditions by dropping the summation signs and the difference variables

$$c_k + d_k > \frac{1}{n} \{a_x + b^x_k + a_y + b^y_k\}.$$

In the case of disconnected graphs $C$ and $D$ where all oscillators are coupled through a combination of the two graphs and $b^{int}_k$ is non-zero, the third and forth sums are always present on the RHS. This makes the argument much more complicated but yields a number of surprising implications of the stability method to specific networks discussed in Sections V and VI.

A major stability problem, associated with the third and fourth terms, is rooted in the fact that, for example, the third sum $\sum_{k \in D} [\omega_x b^{int}_k] X_k^2$ contains the difference variables $X_k$ that correspond to the edges of the $y$ graph. As a result, the first sum $n \sum_{k=1}^{m} c_k X_k^2$ on the RHS which contains the variables $X_k$ that correspond to $x$ edges, cannot compensate the third sum on the RHS as they belong to different graphs and therefore cannot be compared. At the same time, the second sum $n \sum_{k=1}^{l} d_k Y_k^2$ on the RHS does belong to the $y$ graph but contains the variables $Y_k$ and not $X_k$ needed to handle the third sum. The same problem relates to the fourth sum $\sum_{k \in C} [\omega_y b^{int}_k] Y_k^2$ which contains $Y_k$ variables, corresponding to the “wrong” graph ($x$ graph).

How can we get around this problem as we simply do not have means on the LHS to compensate for the troublesome sums on the RHS? A solution comes from economics: if you do not have means, borrow them! [But act responsibly]. This remark is added to entertain the reader that might be tired of following the proof up to this point.

In fact, the only place to “borrow” these terms from is the auxiliary stability systems (20) and (21) as they do contain the desired variables $X_k$ and $Y_k$, corresponding to the “right” graphs (the $x$ and $y$ graphs, respectively). Therefore, we need to go back and modify the auxiliary systems (20) and (21) as follows

$$\dot{X}_{ij} = \int_0^1 DF(\beta x_j + (1 - \beta) x_i) d\beta - [a_x + \alpha^x_k] P + \omega_x b^{int}_k L X_{ij} \text{ if } i, j \in x \text{ edge } k,$$

$$\dot{X}_{ij} = \int_0^1 DF(\beta x_j + (1 - \beta) x_i) d\beta - (a_y + \alpha^y_k) L + \omega_y b^{int}_k P X_{ij} \text{ if } i, j \in y \text{ edge } k.$$
The addition of a positive term $\omega_x b^{int}_k LX_{ij}$ to the auxiliary system (38) worsens its stability, therefore we have to introduce an additional parameter $\alpha^x_k$ and make sure that it is sufficiently large to stabilize the new auxiliary system. A very important property is that, in (38), we have to add the positive, destabilizing term $\omega_x b^{int}_k LX_{ij}$ to the second equation for the $(y_j - y_i)$ difference but try to stabilize the system via increasing the additional parameter $\alpha^x_k$ in the first equation for the $(x_j - x_i)$ equation (note the different inner matrices: $P$ versus $L$ in (38)). Depending on the individual oscillator, chosen as the individual unit, this might not be possible, especially when traffic load $b^{int}_k$ on the edge $k$ is high. This property is discussed in detail for the Lorenz and double scroll oscillator examples in Sections V and IV. A similar argument carries over to the auxiliary system (39) where we add the destabilizing term $\omega_y b^{int}_k X_{ij}$ to the $(x_j - x_i)$ equation, but seek to stabilize the system via the additional parameter $\alpha^y_k$ in the $(y_j - y_i)$ equation.

Notice that we only modify the auxiliary systems for $x$ and $y$ edges. All the other auxiliary systems for $X_{ij}$, that do not correspond to edges of either the $x$ or $y$ graphs, remain intact and defined via the original systems (20)-(22).

Thus, the modifications of (38) and (39) make the troublesome sums $\sum_{k \in C} [\omega_x b^{int}_k X^2_k + \sum_{k \in C} \omega_y b^{int}_k Y^2_k]$ in (36) disappear at the expense of worsened stability conditions of the corresponding auxiliary systems, which is reflected by the appearance of additional terms with $\alpha^x_k$ and $\alpha^y_k$.

Therefore, (36) turns into

$$\sum_{k=1}^{n} c_k X^2_k + \sum_{k=1}^{l} d_k Y^2_k > \sum_{k=1}^{n} [\alpha^x_k X^2_k + \sum_{l=1}^{l} [\omega_x b^{int}_k + \alpha^y_k]^2 Y^2_k].$$

Notice the new stabilizing constants $\alpha^x_k$ and $\alpha^y_k$. Depending on the individual oscillator dynamics and traffic load on edge $k$, these constants might have to be very large or even infinite.

Comparing the terms containing $X_k$ and $Y_k$ on the LHS and RHS of (40) and omitting the summation signs, we obtain the following conditions

$$nc_k X^2_k > [\omega_x b^{int}_k + \alpha^x_k]^2 X^2_k, \quad k = 1, \ldots, m$$

$$nd_k Y^2_k > [\omega_y b^{int}_k + \alpha^y_k]^2 Y^2_k, \quad k = 1, \ldots, l.$$  

Finally, we omit the difference variables to obtain the bounds on coupling strengths, $c_k$ for $x$ edges and $d_k$ for $y$ edges, sufficient to make the derivative of the Lyapunov function (25) negative semi-definite, and therefore, ensure global stability of synchronization in the network (1). It follows from (41) that these upper bounds are

$$c_k > \frac{1}{\alpha^x_k} [\omega_x b^{int}_k + \alpha^x_k], \quad k = 1, \ldots, m$$

$$d_k > \frac{1}{\alpha^y_k} [\omega_y b^{int}_k + \alpha^y_k], \quad k = 1, \ldots, l.$$  

This completes the proof of Theorem 1. \qed

[18] Francesco Sorrentino and Edward Ott, “Adaptive syn-


[52] Filippo Radicchi and Ginestra Bianconi, “Redundant in-


