Network Science and Automation

Noname manuscript No.
(will be inserted by the editor)

Network Science and Automation

Lorenzo Zino · Baruch Barzel · Alessandro Rizzo

Abstract From distributed sensing to autonomous vehicles, networks are a crucial component of almost all our automated systems. Indeed, automation requires a coordinated functionality among different, self-driven, autonomous units. For examples, robots that must mobilize in unison, vehicles or drones that must safely share space with each other, and, of course, complex infrastructure networks, such as the Internet, which requires cooperative dynamics among its millions of interdependent routers. At the heart of such multi-component coordination lies a complex network, capturing the patterns of interaction between the autonomous nodes. This network allows the different units to exchange information, influence each other's functionality, and, ultimately, achieve globally synchronous behavior. Here, we layout the mathematical foundations for such emergent large-scale network cooperation. First, analyzing the structural patterns of networks in automation, then showing how these patters contribute to the system's resilient and coordinated functionality. With this toolbox at hand, we discuss common applications, from cyber-resilience to sensor networks and coordinated robotic motion.

Keywords cascading failures · consensus · coupled oscillators · distributed control · distributed sensing · dynamics on networks · networks · pinning control · resilience · synchronization

L. Zino
Faculty of Science and Engineering, University of Groningen, Groningen, The Netherlands
E-mail: lorenzo.zino@rug.nl

B. Barzel
Department of Mathematics, Bar-Ilan University, Ramat-Gan, Israel
The Gonda Interdisciplinary Brain Science Center, Bar-Ilan University, Ramat-Gan, Israel
E-mail: baruchbarzel@gmail.com

A. Rizzo
Department of Electronics and Telecommunications, Politecnico di Torino, Torino, Italy
Office of Innovation, New York University Tandon School of Engineering, Brooklyn NY, US
E-mail: alessandro.rizzo@polito.it

Contents

1 Overview

In its classic form, automation focuses on replacing human labor and, at times, also cognitive skills, with autonomous units. Robots and machines, for example, take the place of manual workers in production lines, and computers enhance our ability to analyze data and conduct elaborate calculations. Yet, modern automation reaches beyond the autonomous *units*, and scales our view up to the autonomous *system*. Indeed, many of our most crucial technological systems cannot be simply reduced to their technical components, but rather their true function and value is rooted in their ability to drive these components towards cooperative large-scale behavior. For example, the Internet, in technical terms, is a critical infrastructure comprising routers, cables and computers. Yet, put together, the Internet is *not* a big computer or router. It is an evolving system that interacts and responds to its human environment, shares, produces and processes information — and, in its essence — exhibits functions the supersede that of its autonomous component.

More broadly, whether it involves drones, autonomous vehicles or sensing units, modern automation is an emergent phenomenon, in which technological units — nodes — cooperate to generate higher level functionality. *Simple building blocks come together to preform complex functions*. Lacking central design, and without specific control over each and every unit, these complex multi-component systems often amaze us with their reliable and resilient functionality. Indeed, we can carelessly send an email, and have no doubt that the Internet's globally distributed routers will find an efficient path for it to reach its destination. Similarly, we turn on electrical units and need not think about the automatic redistribution of loads across all power grid components.

The basis for this seemingly uncoordinated cooperation, lies not at the technology of the units themselves, but rather at their patterns of connectivity — namely the *network*. Each of these systems has an underlying complex network, allowing the components to transfer information, distribute loads and propagate signals, thus enabling a synchronous action among distributed, self-driven, autonomous nodes.

We therefore, now, transition our view point, from the specific technology of the components, to the design principals of their interaction networks. What characteristics of the network allow units to reach a functional consensus? What are the bottlenecks and Achilles' heels of the network that mark its potential vulnerabilities. Conversely, what ensures the network's resilience in the face of failures and attacks? Hence, in this Chapter we seek the fundamental principles that look beyond the *components*, and allow us to mathematically characterize, predict and ultimately control the behavior of their *networks*.

2 Network structure and definitions

Seeking the connectivity patters that drive networks and automation, we first begin by laying the mathematical foundations - indeed, the language, of network science. We focus on the network characteristics that, as we show below, play a key role in the emergent behavior of our interconnected autonomous systems.

2.1 Notation

We gather here the notational conventions used throughout this chapter. Unless otherwise stated, vectors and matrices have dimension n and $n \times n$, respectively. The sets $\mathbb{R}$ and $\mathbb{R}_+$ denote the set of real and real non-negative numbers. The sets $\mathbb{Z}$ and $\mathbb{Z}_+$ denote the set of integer and integer non-negative numbers. The set $\mathbb{C}$ denote the set of complex numbers. Given a complex number $z \in \mathbb{C}$, we denote by $|z|$ its modulus, by $\mathcal{R}(z)$ its real part, and by $\mathcal{I}(z)$ its imaginary part. The imaginary unit is denoted as ι. Given a vector $\boldsymbol{x}$ or matrix $\mathbf{M}$, we denote by $\boldsymbol{x}^\top$ and $\mathbf{M}^\top$ the transpose vector and matrix, respectively. We use $||\boldsymbol{x}||$ to denote $\boldsymbol{x}$'s euclidean norm. The symbol $\mathbf{1}$ denotes a vector of all 1 entries, *i.e.*, $\mathbf{1} = (1, \dots, 1)^\top$; $\mathbf{I}$ denotes the identity matrix, and, more generally, $\mathrm{diag}(\boldsymbol{x})$ represents a matrix with vector $\boldsymbol{x}$ on its diagonal and 0 for all its off-diagonal entries. Given a set $\mathcal{S}$, we denote by $|\mathcal{S}|$ its cardinality.

2.2 Networks as graphs

The most basic mathematical model of a network is given by a graph with n *nodes*, denoted by positive integers, comprising the *node set* $\mathcal{V} = \{1, \dots, n\}$. These nodes are connected through a set of directed *edges* (or links), *i.e.*, the *edge set* $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, such that $(i, j) \in \mathcal{E}$ if and only if node i is connected to node j. An edge that connects node i to itself, (i, i), is called *self-loop*. The pair of sets $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ defines the *network*.

To track the spread of information between nodes in a network we map its pathways, linking potentially distant nodes through indirect connections. A *path* from i to j is a sequence of edges $(i, i_1), (i_1, i_2) \dots, (i_{\ell-1}, j) \in \mathcal{E}$, where ℓ is the path *length*. In case $i = j$ the path is called *cycle*. A path (cycle) is said to be *simple*, if no node is repeated in the sequence of edges (apart from the first and last nodes in a cycle). The length of the shortest path/s between i and j, ℓ_{ij}, provides a measure of distance between them. This, however, is not, formally a metric, as the distance from i to j may, generally, be different than that from j to i. The greatest common divisor of the lengths of all the cycles passing through a node i is denoted as p_i and called the *period* of node i, with the convention that, if no cycles pass through i, then $p_i = \infty$. If $p_i = 1$, we say that node i is *aperiodic*. Note that, if i has a self-loop, then it necessarily is aperiodic. If there is a path from i to j, we say that j is *reachable* from i. If a node i is reachable from any $j \in \mathcal{V}$, we say that i is a *globally reachable node*. If i is reachable from j and j is reachable from i, we say that the two nodes are *connected*. Connectivity is an equivalence relation, which determines a set of equivalence classes, called *strongly-connected component*. If all the nodes belong to the same strongly-connected component, we say that the network is *strongly-connected*. It can be shown that the period of a node and its global reachability are class properties, that is, all the nodes belonging to the same class have the same period and, if one node is globally reachable, then all the nodes share this property. Hence, we will say that a strongly-connected component is globally reachable and has period p.

For strongly connected graphs, paths provide a measure of node "centrality" in the network. Specifically, denoting the set $\mathcal{S}_{ij}$ containing all the *shortest paths* between two nodes i and j, we define the *betweenness centrality* of a node i as the

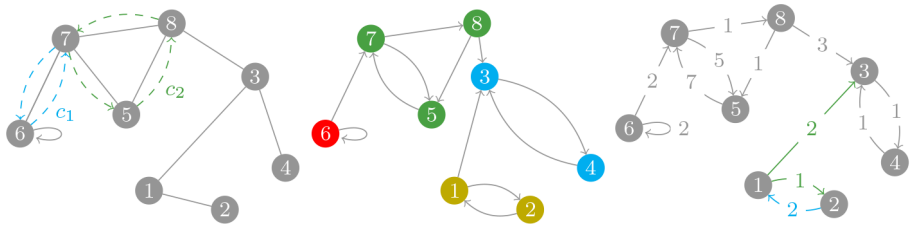

Fig. 1: Three examples of networks: (a) an undirected network, (b) a directed network, and (c) a weighted (directed) network. The network in (a) is strongly connected and aperiodic, that is, $p = 1$. Aperiodicity can be verified by observing that through node 7 we have the cyan cycle $c_1 = (7, 6), (6, 7)$ and the green cycle $c_2 = (7, 5), (5, 8), (8, 7)$, of length equal to 2 and 3, respectively. Node 6 has a self loop and degree $d_6 = 2$. In (b) there are four strongly connected components, highlighted in different colors: $\{6\}$, $\{5, 7, 8\}$, $\{1, 2\}$, and $\{3, 4\}$; the latter (the cyan one) is globally reachable and has period $p_3 = p_4 = 2$. Node 1 has out-degree $d^+ = 2$ and in-degree $d^- = 1$. In (c), we show the network in (b) equipped with a weight matrix. Note that weight $W_{12} = 1$, and that node 1 has weighted out-degree $w_1^+ = 3$ and weighted in-degree $w_1^- = 2$ (out-going and in-going edges are denoted in green and cyan, respectively).

(normalized) number of shortest paths traversing through i, namely

$$\gamma_i \propto \sum_{j,k \in \mathcal{V}} \frac{|\{s \in \mathcal{S}_{jk} : i \in s\}|}{|\mathcal{S}_{jk}|} . \tag{1}$$

As we shall see in Section 4.2, the betweenness centrality plays a key role in determining the vulnerability of a network to cascading failures. Betweenness centrality, we emphasize, is just one of the many centrality measures that can be defined to rank the importance of nodes in a network. For example, in the next section, we present the Bonacich centrality [1].

Given a node $i \in \mathcal{V}$, we denote by

$$\mathcal{N}_i^+ := \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\} \quad \text{and} \quad \mathcal{N}_i^- := \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\} \tag{2}$$

the sets of *out-neighbors* and *in-neighbors*, respectively, namely, the set of nodes that i links to, and the nodes linking to i, respectively. The cardinality of $\mathcal{N}_i^+$ is i's *out-degree* $d_i^+ := |\mathcal{N}_i^+|$, and that of $\mathcal{N}_i^-$ is its *in-degree* $d_i^- := |\mathcal{N}_i^-|$.

A network is said to be *undirected* if for any $(i, j) \in \mathcal{E}$, then $(j, i) \in \mathcal{E}$. For undirected networks, much of the notation presented above simplifies. In fact, if i is reachable from j, then also j is necessarily reachable from i. As a consequence, for undirected graphs, if there is a globally reachable nodes, then the network is necessarily strongly connected. Furthermore, the in- and out-neighbors coincide, and consequently the in-degree is equal to the out-degree. Hence, we can omit the prefix in/out, and refer to the *neighbors* and the *degree* of a node. In an undirected network distance is symmetric and each edge is also a cycle of length 2. See Figure 1 for some illustrative examples of networks and their properties.

2.3 Network matrices and their properties

We can use matrices to represent networks. Given a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we define a non-negative-valued matrix $W \in \mathbb{R}_{+}^{n \times n}$, such that $W_{ij} > 0 \iff (i,j) \in \mathcal{E}$. The resulting matrix $\mathbf{W}$ assigns each edge of the graph (i,j) with a *weight* W_{ij}, providing the network's *weighted adjacency matrix*. Together, this yields the *weighted network*, denoted by the triplet $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$. For each node $i \in \mathcal{V}$, we denote by $w_i^+ := \sum W_{ij}$ and $w_i^- := \sum W_{ji}$ its *weighted out-* and *in-degrees*, respectively. See Figure 1 for an example.

Often the weights W_{ij} capture transition rates between nodes, and hence the weight matrix $\mathbf{W}$ is assumed to be a stochastic matrix, in which all rows sum to 1. For such matrices, the Perron-Frobenius theory can be used to study the spectral properties of $\mathbf{W}$ [2]. An immediate observation is that 1 is an eigenvalue of $\mathbf{W}$, associated with the right-eigenvector $\mathbf{1}$. All the other eigenvalues of $\mathbf{W}$ are not larger than 1 in modulus. Further results can be established, depending on the connectivity properties of the network, which can be directly mapped into properties of the matrix $\mathbf{W}$. For instance, strong connectivity of the network yields irreducibility of $\mathbf{W}$, which guarantees that the eigenvalue 1 is simple, and it admits a left-eigenvector $\boldsymbol{\pi}$ with strictly positive entries. This result can be extended to networks with a unique globally-reachable strongly-connected component, whereby the left-eigenvector $\boldsymbol{\pi}$ has strictly positive entries corresponding to the nodes belonging to the component, whereas the other entries are equal to 0 (see, *e.g.*, [3]). Note that, under these hypotheses we have

$$\lim_{t \to \infty} \mathbf{W}^t \to \mathbf{1}\boldsymbol{\pi}^\top. \tag{3}$$

Such repeated multiplications of $\mathbf{W}$, $t \to \infty$, capture the convergence of a linear process driven by $\mathbf{W}$. Therefore, the left-eigenvector $\boldsymbol{\pi}$, the *Bonacich centrality* of the weighted network, plays a key role in determining the consensus point of linear averaging dynamics, as we will discuss in Section 3.1.

Next we define the weighted *Laplacian matrix* as

$$\mathbf{L} = \mathrm{diag}(\boldsymbol{w}^+) - \mathbf{W}, \tag{4}$$

where $\boldsymbol{w}^+$ is a vector that assembles the out-degrees of all nodes. In simple terms, the diagonal entries of the Laplacian contain the weighted out-degree of the node, while the off-diagonal entries are equal to $-W_{ij}$ the sign inverse of the i,j weight. As opposed to $\mathbf{W}$ that contains the entire information on the network structure, in the Laplacian, the information on the presence and weight of the self-loops is lost. By design, the rows of $\mathbf{L}$ sum to 0. Consequently, the all-1 vector, a right eigenvector of the Laplacian, now has eigenvalue 0.

For stochastic matrices, as all rows sum to unity, the Laplacian reduces to $\mathbf{L} = \mathbf{I} - \mathbf{W}$. Hence, the two matrices share the same eigenvectors $\xi_1, \ldots, \xi_n$, and their eigenvalues are shifted by 1, namely if μ_k is an eigenvalue of $\mathbf{W}$ then $\lambda_k = 1 - \mu_k$ is an eigenvalue of $\mathbf{L}$. Therefore, all the spectral properties described above for stochastic matrices $\mathbf{W}$ apply directly to the study the Laplacian spectrum.

2.4 Real-world networks

Social, biological and technological networks have been extensively studied over the past two decades, uncovering several widespread characteristics, universally observed across these different domains. For example, most real networks exhibit extremely short paths between all connected nodes, with the average shortest path length often scaling as $\bar{\ell} \sim \log n$, a slow logarithmic growth with the network size [1]. Another universal feature, crossing domains of inquiry, is degree-heterogeneity. This is captured by $P^{\pm}(d)$, the probability for a randomly selected node to have an in-degree $d_i^- = d$ (out-degree $d_i^+ = d$). In many real-world networks these probabilities are fat-tailed, describing a majority of low degree nodes, coexisting alongside a small minority of *hubs*, that can have orders of magnitude more neighbors than average. According to real-world empirical observations [4], several real-world degree distributions in technological and biological networks can be accurately approximated by a power-law of the form $P^{\pm}(d) \sim d^{-\gamma}$, describing a collection degree that lacks a typical scale, *i.e.*, *scale-free networks* [1]. The exponent γ plays a key role in shaping the degree distribution, whereby smaller values of γ yield more heterogeneous and disperse distributions. In particular, power-law distributions with exponent $\gamma \in [2, 3)$, which are often observed in real-world networks, exhibit a constant average degree, whereas its variance grows with the network size. Other real-world networks, including many social networks exhibit weaker scale-free properties, which can be captured, for instance, by log-normal distributions [5]. Finally, another key feature of many real-world networks, especially in the social and biological domains, is the tendency of nodes to create clusters, whereby two nodes that are connected have a higher probability of having other connections "in common." Such a property, in combination with the slow logarithmic growth of the shortest paths, characterizes the so told *small-world network* models, which have become popular in network science in the past few decades [6].

3 Main results on dynamics on networks

Up to this point we discussed the characterization of the network's static structure - *who is connected to whom*. However, networks are truly designed to capture dynamic processes, in which nodes spread information and influence each other's activity [7, 8]. Hence, we seek to use the network to map *who is influenced by whom*. We therefore, discuss below dynamics flowing on the networks.

3.1 Consensus problem

Since the second half of the 20th century, the *consensus problem* has started attracting the interest of the systems and control community. The reason for such a growing interest is that the consensus problem, initially proposed by J. French and M.H. De Groot as a model for social influence [9,10], has found a wide range of applications, encompassing opinion dynamics [11,12], distributed sensing [13], and formation control [14]. In its very essence, the problem consists of studying whether a network of dynamical systems is able to reach a common state among its nodes in a distributed fashion via pairwise exchanges of information, lacking

any sort of centralized entity. In the following, we provide a formal definition of the problem and illustrate some of its main results.

Here our nodes represent *units* (also called *agents*) $\mathcal{V} = \{1, \dots, n\}$, modeling, for instance, individuals, sensors, or robots. The agents are connected through a weighted network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$, where $\mathbf{W}$ is a stochastic matrix. Each agent $i \in \mathcal{V}$ is characterized by a state variable $x_i(t) \in \mathbb{R}$, which, at every time-step, is updated to the weighted average of the states of its neighboring agents - hence the *consensus* dynamics. The consensus problem can be formulated either in a discrete-time framework ($t \in \mathbb{Z}_+$), or in continuous time ($t \in \mathbb{R}_+$). In discrete-time, the simplest state update rule is formulated via

$$x_i(t+1) = \sum_{j \in \mathcal{V}} W_{ij} x_j(t), \tag{5}$$

or, in a compact matrix form, as

$$\boldsymbol{x}(t+1) = \mathbf{W}\boldsymbol{x}(t), \tag{6}$$

where $\boldsymbol{x}(t)$ is a n-dimensional vector capturing all the agents' states. In the continuous version, Eqs. (5) and (6) are formulated using the weighted Laplacian matrix as

$$\dot{\boldsymbol{x}}(t) = -\varepsilon \mathbf{L}\boldsymbol{x}(t), \tag{7}$$

where $\varepsilon > 0$ is a positive constant that determined the rate of the consensus dynamics.

One of the main goals of the consensus problem is to study how the topological properties of the network influence the asymptotic behavior of $\boldsymbol{x}(t)$. In particular, the key question is to determine under which conditions the states of the agents converge to a common quantity, reaching a *consensus state* x^*.

Perron-Frobenius theory [2] provides effective tools to study the consensus problem in terms of the spectral properties of the weight matrix $\mathbf{W}$, establishing conditions that can be directly mapped into connectivity patterns of the network. Specifically, it is shown that a consensus state is reached for any initial condition $\boldsymbol{x}(0)$ if and only if the graph described by $\mathbf{W}$ has a unique globally-reachable strongly-connected component. In the case of discrete-time consensus, it is also required that such a strongly-connected component is aperiodic [3]. An extremely important consequence of this observation, especially for applications seeking to drive agents towards consensus, is that such convergence is guaranteed if the network is strongly connected and, for discrete-time consensus, if at least one self-loop is present. This captures rather general conditions to reach consensus.

A second key result obtained through the Perron-Frobenius theory touches on the characterization of the consensus state. Specifically, given the initial conditions $x(0)$ and the (unique) left eigenvector of $\mathbf{W}$ associated with the unit eigenvalue $\boldsymbol{\pi}$, the consensus point coincides with the weighted average

$$x^* = \boldsymbol{\pi}^\top x(0) = \sum_{i \in \mathcal{V}} \pi_i x_i(0). \tag{8}$$

More specifically, each agent contributes to the consensus state proportionally to its Bonacich (eigenvector) centrality. This result has several important consequences. Notably, agents that do not belong to the unique globally-reachable

aperiodic strongly-connected component do not contribute to the formation of the consensus state, as their centrality is equal to 0 [3]. A special case of the consensus problem is when the consensus is driven towards the arithmetical average of the initial conditions. This is observed if and only if the matrix $\mathbf{W}$ is doubly stochastic (a trivial case occurs when the matrix is symmetric). A simulation of a consensus dynamics with $\mathbf{W}$ doubly-stochastic is illustrated in Figure 2b.

Consensus state x^* captures, when indeed reached, the long term state of the system. Next, we seek to estimate the rate of convergence to this consensus. For a review of the main findings, see, *e.g.* [15]. Given the linear nature of the consensus dynamics, the convergence rates are closely related to the eigenvalues of the stochastic matrix $\mathbf{W}$. Intuitively, the speed of convergence of the consensus dynamics depends on the speed of convergence of the matrix $\mathbf{W}^t$ to $\mathbf{1}\boldsymbol{\pi}^\top$, which is determined by the second largest eigenvalue in modulus, denoted by ρ_2 — under the convergence conditions for the discrete-time consensus dynamics, we recall that $\rho_2 < 1$. For symmetric matrices $\mathbf{W}$, all the eigenvalues are real and positive, so ρ_2 is the second largest eigenvalue, and we can directly write the speed of convergence of the discrete-time consensus as

$$||\boldsymbol{x}(t) - \mathbf{1}x^*|| \leq \rho_2^t ||\boldsymbol{x}(0)||\,, \tag{9}$$

following [16]. A similar expression can be obtained for the continuous-time model Eq. (7). For non-symmetric matrices, the result is slightly more complicated, due to the potential presence of non-trivial Jordan blocks. More details and an explicit derivation of the speed of convergence for generic matrices $\mathbf{W}$ can be found in [3].

Building on these seminal results, the systems and control community has extensively studied this problem toward designing algorithms to converge to desired consensus points, and expanding its original formulation to account for a wide range of extensions. These extensions include, but are not limited to, asynchronous and time-varying update rules [17,18], the presence of antagonistic interactions [19], quantized communications [20], and the investigation of different operators than other the weighted average, such as maximum and minimum operators [21].

3.2 Synchronization

The consensus problem captures a cooperative phenomenon, in which agents reach a common state in the absence of centralized control. More broadly, this problem is part of a general class of challenges, of synchronizing a set of coupled dynamical systems in a distributed fashion toward a common trajectory. From a historical perspective, and also due to its applicability to real-world settings [22], the synchronization challenge is often formulated in continuous time. In this vein, synchronization focuses on a state vector, associated with each agent, $\boldsymbol{x}_i \in \mathbb{R}^d$, rather than a scalar state variable. In the absence of a coupling network, these state variables evolve according to a generic non-linear function $F : \mathbb{R}^d \to \mathbb{R}^d$, capturing each individual node's self-dynamics. The couplings via W_{ij} then introduce a distributed, potentially non-linear averaging dynamics, regulated by a function $G : \mathbb{R}^d \to \mathbb{R}^d$. Overall, the state of agent i is governed by the differential equation

$$\dot{\boldsymbol{x}}_i(t) = F(\boldsymbol{x}_i) - \varepsilon \sum_{j \in \mathcal{V}} L_{ij} G(\boldsymbol{x}_j(t)), \tag{10}$$

where $\varepsilon \geq 0$ is a non-negative parameter that quantifies the strength of the coupling between the agents. Setting the function F null and G equal to the identity, Eq. (10) reduces to the consensus dynamics in Eq. (7). Figures 2c and 2d illustrate an example of a set of Rössler oscillators in the absence and in the presence of coupling, respectively, on the network illustrated in Figure 2a.

The goal of the synchronization problem consists of determining the conditions that the non-linear functions F and G and the network — here, represented by the Laplacian matrix $\mathbf{L}$ — have to satisfy in order to guarantee that the vector state of each node $\boldsymbol{x}_i(t)$ converges to a common trajectory $\boldsymbol{x}^*(t)$. Formally, this maps to studying the stability of the synchronization manifold

$$\mathcal{S} := \{\boldsymbol{x}(t) : \boldsymbol{x}_i(t) = \boldsymbol{x}_j(t) = \boldsymbol{x}^*(t), \forall\, i, j \in \mathcal{V}\}. \tag{11}$$

From Eq. (10) and from the zero-sum property of $\mathbf{L}$, it is straightforward to observe that the synchronization manifold $\mathcal{S}$ is invariant, that is, if $\boldsymbol{x}(t^*) \in \mathcal{S}$, for some $t^* \geq 0$, then the system will stay synchronized for all $t \geq t^*$. Hence, the problem of the synchronization of coupled dynamical systems ultimately boils down to determining the asymptotic stability conditions for such a manifold.

The seminal work by [23] introduced an effective and elegant technique to study the local stability of the synchronous state, through the study of modes transverse to it. The approach, based on a master stability function (MSF), decouples the synchronization problem in two independent sub-problems, one focusing on the node's dynamics, the other on the network topology. First, the dynamical system in Eq. (10) is linearized about the synchronous state $\boldsymbol{x}^*(t)$ by defining $\delta\boldsymbol{x}_i(t) = \boldsymbol{x}_i(t) - \boldsymbol{x}^*(t)$, and writing the linearized equations

$$\delta\dot{\boldsymbol{x}}_i = \mathbf{J}_F(\boldsymbol{x}^*)\delta\boldsymbol{x}_i - \varepsilon\mathbf{J}_G(\boldsymbol{x}^*)\sum_{j\in\mathcal{V}} L_{ij}\delta\boldsymbol{x}_j, \tag{12}$$

with $i \in \mathcal{V}$ and where $\mathbf{J}_F \in \mathbb{R}^{d\times d}$ and $\mathbf{J}_G \in \mathbb{R}^{d\times d}$ are the Jacobian matrices of F and G, respectively. The system of equations in Eq. (12) can be further decomposed into n decoupled equations by projecting them along the eigenvectors of the Laplacian $\mathbf{L}$, obtaining

$$\dot{\xi}_k = [\mathbf{J}_F(\boldsymbol{x}^*) - \varepsilon\lambda_k\mathbf{J}_G(\boldsymbol{x}^*)]\xi_k, \tag{13}$$

where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of $\mathbf{L}$, in ascending order in modulus. Assuming $\mathbf{W}$ to be strongly connected, we have $\lambda_1 = 0$ and $\lambda_2 \neq 0$. Hence, the first term $k = 1$ of Eq. (13) coincides with the perturbation parallel to the synchronization manifold, and all the other $n-1$ equations correspond to its transverse directions. Therefore, the synchronization manifold is stable if all the equations $k = 2, \ldots, n$ in Eq. (13) damp out. We define the MSF $\lambda_{max}(\alpha)$ that associates each complex number $\alpha \in \mathbb{C}$ with the maximum Lyapunov exponent of $\xi = [\mathbf{J}_F(\boldsymbol{x}^*) + \alpha\mathbf{J}_G(\boldsymbol{x}^*)]\xi$. Hence, if $\lambda_{max}(\sigma\lambda_k) < 0$, for all $k = 2, \ldots, n$, the synchronization manifold is locally stable. Interestingly, the MSF allows us to decouple the effects of the dynamics and the role of the network. The nodal dynamics, captured via $\mathbf{J}_F$ and $\mathbf{J}_G$, fully determine the critical region of the complex plane in which $\lambda_{max}(\alpha) < 0$; the network topology, in contrast, determines the eigenvalues λ_k of $\mathbf{L}$. Synchronization is attained if all the eigenvalues belong to the critical region. Hence, the MSF can be effectively utilized to design a network structure to synchronize a general set of dynamical systems in a distributed fashion.

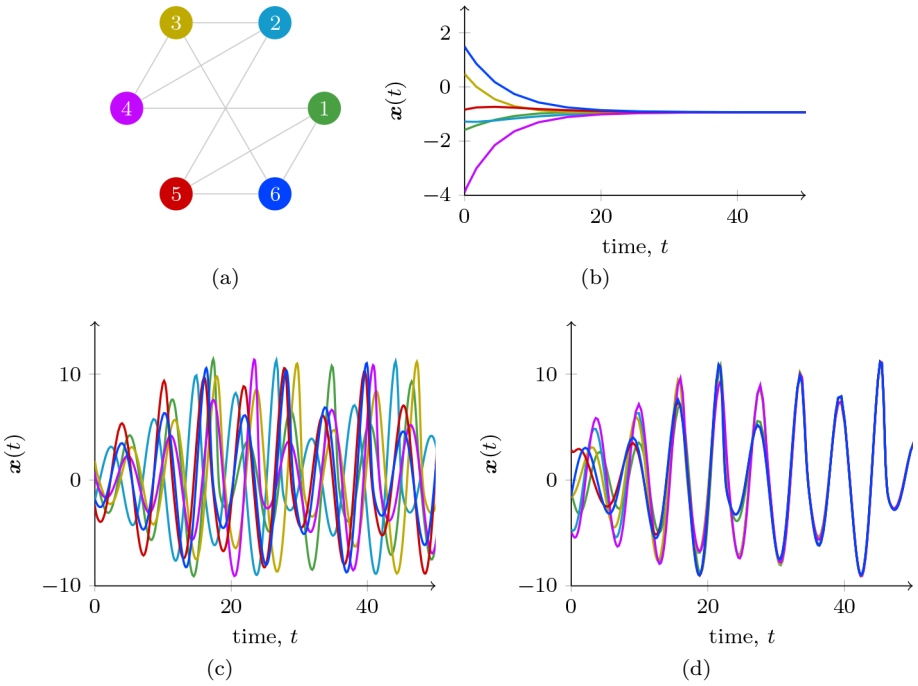

Fig. 2: Consensus dynamics and synchronization. We consider six dynamical systems connected through the 3-regular network illustrated in (a). All the weights are $W_{ij} = 1/3$, as in Eq. (20). In (b), we show the trajectories of a continuous-time consensus dynamics, which converge to the average of the initial conditions. In (c) and (d), we illustrate the trajectories of six Rössler oscillators, with equations $\dot{x}_i = -y_i$, $\dot{y}_i = x_1 + 0.2y_i$, $\dot{z}_i = 0.2 + z_i(x_i - 0.2)$ (for the sake of readability, we plot only the first variable x_i for each of the dynamical systems, a similar behavior is observed for the other variables). In (c), the oscillators are not coupled and the trajectories are not synchronized. In (d), the dynamical systems are coupled with $\varepsilon = 0.3$ and $G(\boldsymbol{x}_j(t)) = [x_j(t), 0, 0]^\top$, leading the trajectories to the synchronization manifold.

Generalizations of the MSF approach have been proposed to deal with many different scenarios, including moving agents [24] time-varying and stochastic couplings [25], and more complex network structures such as multi-layer networks [26, 27] and simplicial complexes [28]. The main limitations for the use of the MSF are that, in general, the computation of the largest Lyapunov exponent can be performed only numerically, thus limiting the analytical capability of the method; and that, using the MSF, only local stability results can be established. Different approaches to study the stability of the synchronization manifolds have been developed in the last two decades by means of Lyapunov methods [29] and contraction theory [30]. While these approaches have allowed specific rigorous, global convergence results, they typically provide sufficient conditions for synchronization that, in general, are more conservative than those established via MSF-based approaches.

3.3 Perturbative analysis

In case Eq. (10) exhibits a stable fixed-point $\boldsymbol{x}^*$, the challenge is to understand its fixed-point dynamics [7,8,31]. This can be observed by following its response to small perturbations, either structural — removing nodes/links or altering weights, or dynamic — small activity perturbations $x^* \to x+\delta x(t)$. Such perturbation instigates a *signal* S, which then propagates through all network pathways to impact the state of all other nodes. First, we wish to track the signal's spatiotemporal propagation, namely how much time will it take for the signal to travel from a *source* node i to a *target* node j. This can be done by extracting the specific response time τ_i of all nodes to the incoming signal, then summing over all subsequent responses along the path from i to j. The resulting propagation patterns uncover a complex interplay between the weighted network structure W_{ij} and the non-linear functions F and G, resulting in three dynamic classes [32]: *Ultra-fast.* for certain dynamics the highly-connected hubs respond rapidly (τ_i small), and hence, as most pathways traverse trough them, they significantly expedite the signal propagation through the network. *Fast.* In other dynamics, the degrees play no role, the system exhibits a typical response time τ_i for all nodes, and the signal propagation is limited by the shortest path length of the network. *Ultra-slow.* In the last class hubs are bottlenecks (τ_i large), causing signals to spread extremely slowly.

The long-term response of the system to the signal S is observed once the propagation is completed and all nodes have reaches their final, perturbed state $x_i^* + \delta x_i(S)$. This defines the *cascade* $C = \{i \in \mathcal{V} : \delta x_i(S) > \text{Th}\}$, the group of all nodes, whose response exceeded a pre-defined threshold Th. Often, the cascade sizes $|C|$ are broadly distributed — most perturbations have an insignificant impact, while a selected few may cause a major disruption [8].

A sufficiently large perturbation can lead to instability, resulting in an abrupt transition from one fixed-point, the desired state, to another, potentially undesired [33]. This can capture, for instance, a major blackout, or an Internet failure, which can be caused by infrastructure damage, *i.e.*, structural perturbations, or by overloads, capturing activity perturbations.

3.4 Control

Consensus and synchronization are *emergent phenomena*, that is, their onset is spontaneously achieved over time by the network nodes. Similarly, perturbation analysis captures the network's response to naturally occurring disturbances. With this in mind, the natural next step is to seek controlled interventions to drive the network towards a desired dynamic behavior, *e.g.*, an equilibrium point, a limit cycle, an attractor, etc. Typically, we wish to achieve such control with limited resources, *i.e.*, acting upon a subset of the network nodes, links, or using a limited amount of energy. Network control can be achieved via *off-line* strategies, such as controlling the nodal dynamic or their coupling, or *on-line* strategies, through adaptive rewiring of the network structure [34]. Here, to outline the principles of control, we focus on a representative problem, where one seeks to synchronize a network system, whose uncontrolled dynamics in Eq. (10) does not naturally converge to a synchronous solution [35].

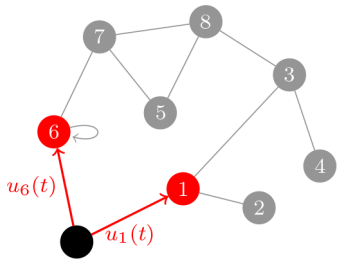

Fig. 3: Schematic of a pinning control scheme. The master node (in black) is connected to the two pinned nodes (in red), namely 1 and 6.

One of the first and most successful methods proposed to synchronize a network is *pinning control*, originally proposed in [36], where one controls the entire network by driving just a small number of nodes. First we designate a *master node*, whose state is fully under our control, namely we can freely determine its dynamics by designing a suitable control input signal. The master node is then linked to a small number of *pinned nodes* $1, \dots, m$, influenced indirectly, via their links to the master node, by our input signal, as illustrated in Figure 3. These pinned nodes follow

$$\dot{x}_i(t) = F(x_i) - \varepsilon \sum_{j \in \mathcal{V}} L_{ij} G(x_j(t)) + u_i(t), \tag{14}$$

where $u_i(t)$ represents the control exerted by the master node on the pinned node $i \in \{1, \dots, m\}$. The remaining $n-m$ nodes continue to evolve according to Eq. (10).

The fundamental question is *how many and which nodes should be pinned in order to control the network?* Clearly, a general answer to this problem does not exist. Indeed, such an answer inevitably depends on the properties of the non-linear functions F and G, on the weighted network topology W_{ij}, and on the form of the control inputs u_i. Therefore, the first approach to tackle this challenge is under linear dynamics, mapping Eq. (14) to

$$\dot{\boldsymbol{x}}(t) = \mathbf{W}\boldsymbol{x} + \mathbf{B}\boldsymbol{u} \tag{15}$$

where is the weighted network matrix, as in the consensus problem Eq. (7), and $\mathbf{B} \in \mathbb{R}^{n \times m}$ is a rectangular matrix, routing the control inputs $\boldsymbol{u}(t)$ to the m pinned nodes. The linear Eq. (15) reduce the control problem to a structural one, seeking the conditions of W_{ij} that guarantee controllability [37].

Under a single pinned node $m = 1$ the controllability of the network is determined by the presence of a subgraph of $\mathcal{G}$ that is a *cactus*, that is, such that it spans all the nodes of the network and in which every edge belongs to at most one simple cycle [37]. Other results based on structural controllability are summarized in [38]. For the general case with $m \geq 1$, standard control-theoretic tools may be leveraged, specifically Kalman's controllability rank condition, showing that Eq. (15) is controllable if the matrix $\mathbf{C} := [\mathbf{B}, \mathbf{WB}, \mathbf{W}^2\mathbf{B}, \dots \mathbf{W}^{n-1}\mathbf{B}]$ has full rank, *i.e.*, $\text{Rank}(\mathbf{C}) = n$. The challenge is to verify Kalman's condition, which becomes unfeasible for large-scale real-world network. To address this, it is shown that the this condition can be mapped to the more scalable graph-theoretic concept of maximal matching [39], where one seeks the largest set of edges with different start and

end nodes. Linking the size of the maximal matching to the number of nodes that should be pinned to control the network, this criterion helps predict Eq. (15))'s controllability. Controllability is found to be primarily driven by the network's degree distribution $P^{\pm}(d)$, independent of the weights W_{ij}, and, counter-intuitively, often driven by the low-degree nodes, especially for directed networks.

Next, if indeed the network is controllable, we wish to design the control inputs $u_i(t)$ to drive it to the desired dynamic state. For a detailed discussion, we refer to the review papers [40,41]. A common approach employs a feedback strategy, setting [36]

$$u_i(t) = k_i G(x^*(t) - x_i(t)), \tag{16}$$

where $x^*(t)$ is the desired dynamic state and $k_i \geq 0$ are the control gains. For example, such feedback control can steer the system to the stationary state $x^*(t) = \bar{x}$, achieving consensus toward a desired equilibrium via control rather than to spontaneous emergence as in Eq. (7). In undirected scale-free networks this strategy shows that controlling the hubs is most effective for reaching synchronization, as it requires a smaller number of nodes to be pinned [36].

The specific structure of the feedback control scheme in Eq. (16) has enabled the scientific community to establish several important results on network synchronizability, even though a complete theory is still lacking. In fact, the expression for $u_i(t)$ in Eq. (16) allows us to write Eq. (14) in a form similar to Eq. (10), with an additional term that accounts for the master node. The master stability function approach used to study synchronization for the uncontrolled problem can then been extended to tackle this scenario. Specifically, in [42], the authors propose to add an $(n+1)$-th equation for the master node, that is, simply $\dot{x}_{n+1} = F(x_{n+1})$. Consequently, all the $n+1$ equations can be written in the form of Eq. (10), where the Laplacian matrix $\mathbf{L}$ is substituted by an augmented Laplacian matrix $\mathbf{M}$, in which a row and a column corresponding to the master node are added. All the remaining entries in this added row are equal to zero, while the column contains the control gains. Furthermore, the corresponding control gains are subtracted from the diagonal of the augmented Laplacian. The MSF-approach can then be applied to the augmented system on $n+1$ equations to determine the stability, depending on $k_1, \ldots, k_m$.

One of the main limitations of the approaches described so far is that they rely on the assumption that all the dynamical systems are identical, which ensures stability of the synchronous trajectory. Unfortunately, in many real-world applications, we face the problem of synchronizing networks of heterogeneous dynamical systems, often in the presence of noise and disturbances. To overcome these problem, in [43,44], the authors proposed to use a coupling term that includes a distributed integral action. Building on this intuition, in [45], a distributed proportional-integral-derivative (PID) control protocol was proposed to reach consensus in a network or linear heterogeneous systems in the presence of disturbances.

In its simplest formulation, we can assume that each node of the network is a unidimensional dynamical system with linear dynamics. Hence, the proposed model can be formulated as follow:

$$\dot{x}_i(t) = a_i x_i(t) + b_i + u_i(t)\,, \tag{17}$$

for all $i \in \mathcal{V}$, where $a_i \in \mathbb{R}$ is a constant that determines the uncoupled dynamic of the system, $b_i \in \mathcal{R}$ is a constant disturbances (*e.g.* an input/output in the

dynamical system i, and $u_i(t)$ is the control action applied to node i at time t. In the standard coupling discussed presented so far, the control action coincides with the continuous-time consensus dynamics, that is,

$$u_i(t) = -\varepsilon \sum_{j \in \mathcal{V}} L_{ij} x_j(t) \,. \tag{18}$$

However, the control action in Eq. (18) is not able to guarantee convergence to consensus in many scenarios, when the parameters a_i and b_i are heterogeneous. In [45], a controller in the following form has been proposed:

$$u_i(t) = -\sum_{j \in \mathcal{V}} L_{ij} \left(\alpha x_j(t) + \beta \int_0^t x_j(s) ds + \gamma \dot{x}_j \right) , \tag{19}$$

where $\alpha, \beta, \gamma \in \mathbb{R}_+$ are constant parameters that weight the three terms in the controller. Note that, besides utilizing the information from the state of the neighboring nodes $x_j(t)$, the proposed controller uses also their integral and derivatives. The analysis of Eq. (19) has allowed the authors of [45] to explicitly derive conditions for the parameters α, β, and γ to control the system toward a consensus. These conditions, besides depending on the disturbances b_i, depend on the network structure through the second largest eigenvalue in modulus of $\mathbf{W}$.

Recently, several works have started to address the control problem also from an energy point of view, namely, toward the development of a minimum energy theory for the control of network systems. Several results have been established for linear systems in the form of Eq. (15). In [46], the authors establish upper and lower bounds on the control cost — defined in terms of the smallest eigenvalue of the Gramian matrix— as scaling laws of the time required to reach synchronization. Further development in this directions have been proposed in [47], where rigorous bounds on the smallest eigenvalue of the Gramian matrix are derived as functions of the eigenvalues of the matrix $\mathbf{W}$ and on the number of pinned nodes m, establishing non-trivial trade-off between the number of pinned nodes and the control cost. Recently, an interesting discussion on minimum energy control is presented in [48]. In this work, using an approach based on standard control-theoretic tools, the authors express the control energy as a function of the real part of the eigenvalues of $\mathbf{W}$. Therein, an interesting heuristic has been proposed for generic (directed) networks, according to which the cost for controlling a network is reduced if the pinned nodes are chosen as those with large (weighted) out-degree (w_i^+) but small in-degree (w_i^-).

4 Applications

Many of our modern technological applications rely on networks. Such systems, *e.g.*, the Internet or the power grid, behave almost as natural phenomena, often lacking a blue-print or central control, hence functioning based on the emergent cooperation between their components. Therefore, the network principles of consensus, synchronization and control play a key role in an array of modern day applications, from distributed sensing to cyber-resilience and coordinated robotic motion.

4.1 Distributed sensing

Sensor networks are a key application of the theoretical findings discussed in the Section 3. Since the 1990s, the use of wired of wireless networks of sensors able to exchange data has become widespread in technological and industrial systems [49]. The Internet of Things has pushed this technology further, providing the market with a plethora of inexpensive devices able to perform sensing, computation, and communication at a negligible cost [49]. One of the most common applications of sensor networks is distributed estimation, that is, the joint estimation of the value of a quantity of interest, with the advantage of reducing the measurement uncertainty, without requiring a centralized processor to gather all the sensors' measurements and centrally perform the desired estimation. There are several advantages to synchronize distributed estimations, from high robustness against noise, errors, and malicious attacks, to lower costs by relinquishing the need for a central processor. Most important, a network of redundant sensors performs under distributed operations avoids the risk of a single point leading to systemic failure.

Formally, in the distributed sensing problem, each of the n sensors is represented by a node, and the communication patterns between sensors by links, leading to the sensor network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Each sensor $i \in \mathcal{V}$ performs a measurement of a common quantity of interest x, whose result we denote by $\hat{x}_i$. The distributed sensing problem consists of designing an algorithm to estimate the actual value of the quantity in a distributed fashion, using local exchanges of information only between the linked sensors, namely sensor can only share information with its out-neighborhood $\mathcal{N}_i^+$. The goal of such estimation is to converge, asymptotically or in finite time, to a common estimate x^* of the quantity x, reducing the uncertainty associated with each local measurement $\hat{x}_i$.

Assuming that the sensors are all identical and unbiased, that is, the expected value of their measurements coincides with x and they all have the same variance σ^2, the best estimate of the quantity of interest can be obtained by computing the arithmetic average of the sensors' measurements, whose variance, following the central limit theorem, approaches σ^2/n [50]. Recalling the consensus problem formalized and discusses in Section 3.1, the arithmetic average of a set of measurements can be computed in a distributed fashion, by initializing the state of each sensor to the measured value, *i.e.*, setting $x_i(0) = \hat{x}_i$, and then updating the state of the sensors according the algorithm in Eq. (5), with a suitable weight matrix $\mathbf{W}$ adapted to the sensor network, in order to ensure convergence to the average. Assumption that the network of sensors is strongly connected and that each sensor can access its own state (all the nodes have a self-loop), we know from Section 3.1 that the state of each node converges to a common quantity, precisely equal to the weighted average of the initial conditions, with weights determined by each sensors' Bonacich centrality, $x_i(t) \to x^* = \boldsymbol{\pi}^\top \hat{\boldsymbol{x}}$. Hence, to estimate the arithmetic average of the initial measurements, one has to design the matrix $\mathbf{W}$ such that its Bonacich centrality is uniform among all nodes. This can be satisfied by designing $\mathbf{W}$ to be doubly stochastic, having both its rows and columns sum to 1.

Hence, the problem of designing an algorithm for distributed sensing ultimately boils down to designing a doubly stochastic matrix $\mathbf{W}$ adapted to the graph representing the sensor network. Several methods have been proposed to successfully address this problem, depending on the properties of the network. The simplest method it to select symmetric weights $\mathbf{W} = \mathbf{W}^\top$, in which case stochasticity di-

rectly implies double-stochasticity. If the network is undirected and regular, namely that all nodes have the same degree $d_i = d$ for all $i \in \mathcal{V}$, this amounts to setting $\mathbf{W}$ with uniform weights as

$$W_{ij} = \frac{1}{d}, \tag{20}$$

for all $(i, j) \in \mathcal{E}$. In this case, at each time-step, each sensor will average its own state with those of all its neighbors. For non-regular networks, where degrees are potentially heterogeneous, the practical solution is to use Metropolis weights setting

$$W_{ij} = \frac{1}{\max\{d_i, d_j\}}, \tag{21}$$

for all $(i, j) \in \mathcal{E}$ such that $j \neq i$, and

$$W_{ii} = 1 - \sum_{j \in \mathcal{V} \setminus \{i\}} W_{ij}. \tag{22}$$

Finally, if, in addition, the network is directed, double-stochasticity requires using Birkhoff's theorem for doubly-stochastic matrices [51]. Specifically, selecting a set of simple cycles $\mathcal{C} = \{c_1, \ldots, c_m\}$ that span all the edges of the network, we let $\mathcal{V}_h$ to be the nodes through which cycle c_h passes and we define the permutation matrix $\mathbf{W}^{(h)}$ associated with the cycle c_h as

$$W_{ij}^{(h)} = \begin{cases} 1 & \text{if } (i,j) \in c_h, \text{ or } i = i \text{ and } i \notin \mathcal{V}_h, \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

Note that each matrix $\mathbf{W}^{(h)}$ is doubly stochastic, but in general, they are not adapted to the network, since only a subset of the edges of the network are associated with non-zero entries of a permutation matrix (precisely, those associated with the corresponding cycle). Then, we construct the doubly stochastic matrix adapted to the network as

$$\mathbf{W} = \frac{1}{m+1} \left(\mathbf{I} + \sum_{h=1}^{m} \mathbf{W}^{(h)} \right). \tag{24}$$

With the appropriate construction it is now verified that the consensus dynamics will solve the distributed sensing problem, leading us to investigate the system's performance. First, we ask how fast does the algorithm converge and, consequently, how to optimally interconnect a set of sensors to guarantee rapid convergence. This problem ultimately boils down to estimating the speed of convergence of the consensus algorithm and, hence, to the computation of the second largest eigenvalue of $\mathbf{W}$. In addition to speed we wish to evaluate accuracy, asking how noise and errors in each sensor's measurement propagate to the distributed estimate x^*. This, we show below problem, is also strictly related to the spectral properties of the weight matrix $\mathbf{W}$.

The problem of understanding how to interconnect a set of nodes to have fast convergence to consensus is a problem of paramount importance in the design of networks of sensors and, more in general, of networked systems. The rate of convergence of the state of the nodes to the desired estimate can be bounded using Eq. (9), that is, as a function of the second largest eigenvalue of $\mathbf{W}$ in modulus

ρ_2. Another important metric that measures the convergence performance of the distributed sensing algorithm is

$$J(t) := \frac{1}{n} \sum_{t \geq 0} ||\boldsymbol{x}(t) - \mathbf{1}x^*||^2 , \tag{25}$$

which can be associated to the cost of a linear-quadratic regulator problem. Also this metric can be related to the spectrum of matrix $\mathbf{W}$. Specifically, in [52], the authors show that for a set of homogeneous unbiased sensors with variance σ^2 in the measurements, it holds

$$J(t) = \frac{\sigma^2}{n} \sum_{i=1}^{n} \frac{1}{1 - |\mu_i|^2}. \tag{26}$$

Clearly, considering both metrics, the optimal solution would be to utilize a fully-connected (complete) network, where convergence is reached in one step, since each sensor can access the information of all the others and compute the average (in fact, for complete graphs it holds $\mu_2 = \cdots = \mu_n = 0$). However, complete network structures may be unfeasible in large-scale systems because they require too many connections, while often sensors may be able to process only a limited number of inputs. The problem of identifying families of graphs with bounded degree that yield fast consensus has been extensively studied in the literature. In [52], the family of de Bruijn graphs has been identified as a good candidate to this goal, ensuring not only much faster convergence rate than other topologies with bounded degree such as lattices [53], but also finite-time convergence. Other important network for which fat convergence results have been established include several realistic models of complex networks, including small-world networks [54, 55].

So far, we have focus our discussion on the performance of a distributed sensing algorithm in terms of its speed of convergence to consensus. Another important performance metric of this family of algorithms, consists of understanding how the error and the noise in the initial sensor's measurements propagate to the distributed estimation. If the sensors are all equal and unbiased, we can assume that each measurement $\hat{x}_i = x + \eta_i$, where $\eta_i \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian distributed random variable with mean equal to 0 and variance σ_i^2. If $\mathbf{W}$ is symmetric (*e.g.* using $\mathbf{W}$ constructed according to Eq. (20) or Eq. (21)), then we can compute the error at the tth iteration of the algorithm as

$$\frac{1}{n}\mathbb{E}[||\boldsymbol{x}(t) - \mathbf{1}x||^2] = \frac{1}{n}\mathbb{E}[||\mathbf{W}^t\boldsymbol{x}(0) - \mathbf{1}x||^2] = \frac{1}{n}\mathbb{E}[||\mathbf{W}^t\boldsymbol{\eta}||^2] = \frac{\sigma^2}{n}\sum_{i=1}^{n}|\mu_i|^{2t}. \tag{27}$$

Note that the estimation error converges to the asymptotic value of $\frac{\sigma^2}{n}$, which coincides with the error of a centralized estimator. The speed of convergence is again associated with the other eigenvalues $\mathbf{W}$, and in particular, with the second largest in modulus, which is the slowest one to converge to 0.

In a scenario of unbiased heterogeneous sensors, the arithmetic average of the sensors' measurements may not be the best estimate of the desired quantity. In fact, assuming that the measurement of sensor $i \in \mathcal{V}$ is a Gaussian random variable with mean equal to x and variance equal to σ_i^2, that is, $\hat{x}_i \sim \mathcal{N}(x, \sigma^2)$ the arithmetic

average of the measurements x^* would lead to an estimate of the desired quantity such that

$$x^* \sim \mathcal{N}\left(x, \frac{1}{n^2}\sum_{i\in\mathcal{V}}\sigma_i^2\right), \tag{28}$$

being a the average of independent Gaussian random variables [50]. instead, the minimum-variance estimator x_{best} is obtained through a weighted average, where the weight given to the measurement of sensor i is proportional to the inverse of its variance [50]. In fact, the estimator obtained with these weights has the following distribution

$$x_{best} \sim \mathcal{N}\left(x, \frac{1}{\sum_{i\in\mathcal{V}}\frac{1}{\sigma_i^2}},\right) \tag{29}$$

whose variance is always less than or equal to the one of the arithmetic mean, with equality holding only if all the variances are equal (in that case, the two averages coincide). Such an estimator can be easily implemented in a distributed fashion, without the need of re-designing the weight matrix $\mathbf{W}$. In fact, the minimum-variance estimation can be computed by running two consensus dynamics in parallel, with a doubly-stochastic matrix $\mathbf{W}$. Specifically, each sensor $i \in \mathcal{V}$ has a two-dimensional state, initialized as

$$y_i(0) = \frac{\hat{x}_i}{\sigma_i^2} \qquad \text{and} \qquad z_i(0) = \frac{1}{\sigma_i^2}, \tag{30}$$

where $\hat{x}_i$ is the measurement of sensor i. Being $\mathbf{W}$ doubly-stochastic, the consensus point of the two states will be equal to

$$y^* = \frac{\sum_{i\in\mathcal{V}}\frac{\hat{x}_i}{\sigma_i^2}}{n} \qquad \text{and} \qquad z^* = \frac{\sum_{i\in\mathcal{V}}\frac{1}{\sigma_i^2}}{n}, \tag{31}$$

which implies that each sensor can calculate the minimum-variance estimator by simply computing the ratio $x_i(t) = y_i(t)/z_i(t)$, which converges to x_{best}. A comparison between a distributed sensing algorithm based on the computation of the average of the measurements and the one proposed in Eq. (31) is illustrated in Figure 4.

The problem of understanding how noisy measurements affect the distributed estimation process becomes crucial when one adopts the distributed estimation process to reconstruct a the state of a dynamical system from a set of repeated measurements obtained from a network of sensors. Such a problem has been extensively studied in the literature, and a two stage strategy to tackle it has been originally proposed and demonstrated in [56,57]. This strategy entails the use of a Kalman filter, which does not require any information from the other sensors, and then p updates, performed according to a consensus dynamic, as illustrated in the schematic in Figure 5. In the simplest implementation, denoted by $y_i(t)$ the measurement of sensor i at time t, the estimate $x_i(t)$ is updated as

$$\begin{cases} z_i(t) = (1-\ell)x_i(t) + \ell y_i(t), \\ x_i(t+1) = \sum_{j\in\mathcal{V}}(W^p)_{ij} z_j(t), \end{cases} \tag{32}$$

where in the first step, a Kalman filter is applied, where $\ell \in (0,1)$ is the (common) Kalman gain, producing a (local) estimate $z_i(t)$; then, the sensors average their

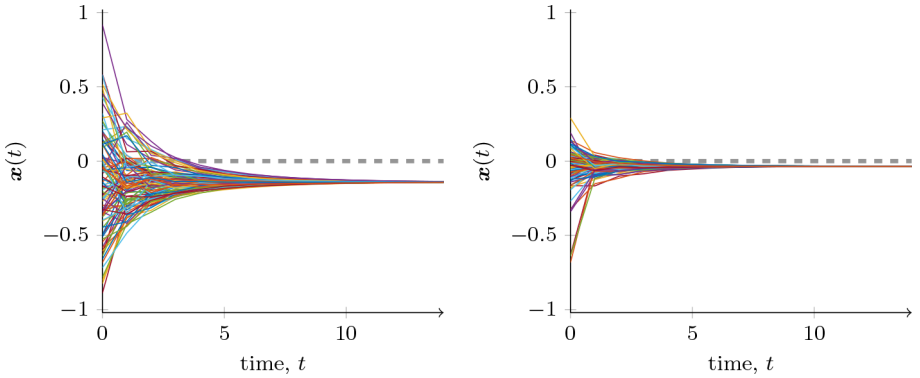

Fig. 4: Comparison between two different methods to perform distributed sensing. In both simulations, 100 sensors make a noisy measurement of a quantity, which is equal to 0. The first 20 sensors have variance $\sigma_i^2 = 0.01$, the other 80 sensors have variance $\sigma_i^2 = 1$. Sensors are connected through a random regular graph with degree 6, generated with a configuration model [1], and $\mathbf{W}$ is defined according to Eq. (20). In (a), we use a standard consensus dynamics; in (b), we use the algorithm proposed in Eq. (31). We observe that the latter sensibly reduces the error in the distributed estimation.

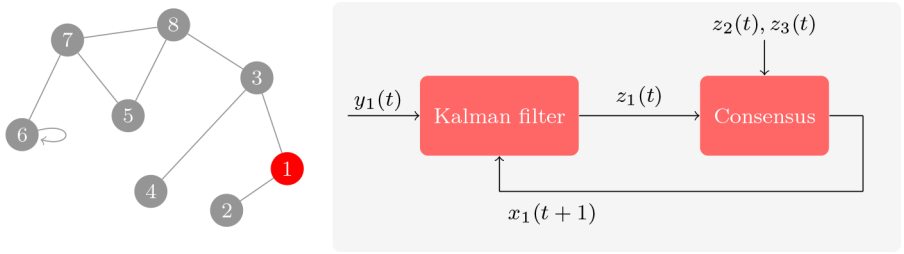

Fig. 5: Schematic of a distributed Kalman filter. The scheme on the right-had-side refers to node 1 (highlighted in red).

estimates according to p consecutive steps of a consensus dynamics, using the local estimates of the neighboring sensors z_j, producing thus a distributed estimate $x_i(t)$. Subsequent studies have allowed the systems and control community to derive rigorous estimations on the error of this estimation process, relating it to the eigenvalues of $\mathbf{W}$, the number of consensus steps m, and the Kalman gain ℓ, and formalizing optimization problems to optimally design the estimation process [53, 58]. Several extensions of this problem has been proposed, for instance, to address heterogeneity among the sensors [59], or to applications such that distributed manipulation strategies to infer mechanical parameters in unknown payloads [60].

4.2 Infrastructure models and cyberphysical systems

The opportunity of mathematically modeling a set of possibly heterogeneous dynamical systems, evolving and interacting over a topology of any complexity, make networks an intuitive modeling paradigm for cyberphysical systems (CPS) and infrastructure models. CPSs are complex interconnected systems of different nature (mechanical, electrical, chemical, etc.), governed by software components, where the behavior of hardware and software is usually deeply intertwined. They usually operate over multiple spatial and temporal scales and require transdisciplinary knowledge to be studied and managed. Typical examples of CPSs are industrial control systems, robotics systems, autonomous vehicles, avionics systems, medical monitoring, tele-surgery, the internet of things, to name a few [61,62]. Infrastructure models, on the other hand, are complex connections of units of the same nature, interconnected or interacting over a spatially-extended network. Typical examples are power grids (see Figure 6), water grids, transportation networks, computer networks, and telecommunication networks [63–68]. It is evident that single infrastructure systems are heavily interdependent: for example, a shortage in the power grid for a prolonged time interval would cut the mobile communication network off. These critical interdependencies have led scholars to gather together single infrastructure models in more complex, interconnected ones, leading to the concept of *critical infrastructure* models. Modeling tools for such systems are interdependent paradigms, such as systems-of-systems, networks-of-networks, or multilayer networks [69].

The most important problem in all these systems concerns the effect of isolated failures on the global functioning of the system, that is, how the performance of the system degrades in spite of isolated or multiple failures [70–74]. In critical infrastructures, moreover, a problem of interest is to study how failures propagate both *over* and *across* the single infrastructures [61,62]. The study of this problem is key toward understanding the mechanisms that trigger cascading failures, for instance, causing massive blackouts in power grids [63], which in turn may damage the functionality of the telecommunication networks [61]. Hence, the results of these can be applied to detect the vulnerability of CPSs and infrastructure systems, preventing the emergence of these disastrous cascading phenomena.

In terms of the topological properties of a network, the vulnerability of a network system is strictly related to the network resilience, that is, the ability of the network to maintains its connectivity when nodes and/or edges are removed. Such a problem has been extensively studied in statistical physics by means of percolation theory [75–77]. In these works, the robustness of different network topologies has been analytically studied, establishing important results that relate the vulnerability of a network to the degree distribution of the nodes and the correlation between the degrees. In [75], the authors use percolation theory to study resilience of different types of random networks, depending on their degree distribution $P(d)$. Specifically, they assume that a fraction p of the nodes is removed randomly, and they study whether there is a critical threshold for p such that, below the threshold, the network has a giant strongly-connected component, while above the threshold such a giant component vanishes. Interestingly, the authors find that such a threshold depends on the ratio between the second moment of the degree distribution and its average. Specifically, the threshold tends to 1 as such a ratio grows. Hence, networks with power-law degree distribution $P(d) \propto d^{-\alpha}$, with

$\alpha \leq 3$ are extremely resilient, as the threshold $p \to 1$ for this types of networks, which are good proxies of several real-world networks, including social networks and internet [1].

However, this "static" concept of network resilience *per se* is often not sufficient to accurately determine the presence of vulnerabilities in a network of dynamical systems, since it is not able to capture the complex interplay between the structural properties of the network and the dynamical aspects of the processes unfolding on the network [33]. In fact, in real-world infrastructures and cyberphysical systems, the global performance of the network may be only mildly affected by the lost of connectivity of marginal nodes, whereas serious damages to the system and its performance can occur even if the network remains connected (*e.g.* due to overloads). This requires us to use the tools of our perturbative analysis of Section 3.3, seeking to predict the onset and propagation of dynamic cascades in the network.

The theory of dynamic vulnerability has started being developed from the concept of cascading failure, that is, understanding when the failure of a node in a network of dynamical systems can trigger a large-scale cascade of failures, causing the collapse of the entire network or of a substantial part of it. In [70], the authors propose a simple but effective model for studying cascading failures caused by overloads in complex networks, such as power grids or cyberphysical systems. In their model, it is assumed that each pair of nodes of a strongly connected network exchanges one unit of quantity (*e.g.* information or energy) through the shortest path between them. According to this model, the load of a node $b_i(t)$ is equal to the number of shortest paths passing through it, and thus proportional to the betweenness centrality γ_i. The authors assume that the capacity of each node is proportional to its initial load, that is,

$$C_i = (1+\alpha) b_i(0) \propto \gamma_i, \tag{33}$$

for some $\alpha \geq 0$, and that nodes whose load is greater than their capacity ($b_i(t) > C_i$) fail and are thus removed from the network. Clearly, each removal will change the network structure, and this may change the loads on the nodes, potentially triggering a cascade of failures. In their work, the authors have shown that the effect of the failure (removal) of a node depends on its betweenness centrality. Specifically, the failure of a node i with small γ_i has a small impact on the network, while the failure of a node i with large γ_i would likely trigger a cascade, especially when the betweenness centrality of the nodes is highly heterogeneous. Interestingly, the authors show that for scale-free networks the failure of the node with higher betweenness centrality may cause the failure of more than 60% of the nodes in a network with $5,000$ nodes. In [71], it has been shown that the immediate removal of some nodes and edges after the initial failure, but before the propagation of the cascade (specifically, nodes with low centrality and edges on which the initial failure would strongly increase the load) can drastically reduce the size of the cascade.

In [72], a similar model for cascading failures has been proposed for weighted networks, where the weight of each edge W_{ij} represents the efficiency of the communication along that edge. In this model, overloaded nodes are not removed, but the communication through edges insisting on them is degraded and thus the

corresponding efficiency is reduced, according to the following formula:

$$W_{ij}(t+1) = \begin{cases} W_{ij} & \text{if } b_i(t) \leq C_i\,, \\ W_{ij}\,\frac{C_i}{b_i(t)} & \text{if } b_i(t) > C_i\,. \end{cases} \tag{34}$$

The authors utilize this model to study the effect of failures on the network performance, evaluated as the average link efficiency, while different performance measures have been discussed in [78,73]. Even for this model, the effect of the failure of a node is related to the (weighted) betweenness centrality of the nodes that fail. Building on these seminal works, a huge body of literature has been developed toward including more real-world features in the modeling frameworks. Specifically, accurate models of power grids have been proposed [74], which have allowed the researchers to accurately identify critical properties of the network structure [64], sets of vulnerable nodes [65], and vulnerable edges [66], by leveraging network-theoretic tools. Besides power grids, these models for cascading failures have been tailored to study the vulnerability of other complex systems, including water supply networks [67] and production networks [68].

These models, while assuming a dynamic exchange of energy between all nodes, are in there essence, still structural, mapping the loads on the nodes to the network structure via γ_i or updating the weights according to Eq. (34). To truly capture the role of the dynamics we must include the non-linear mechanisms F, G driving the system, *a là* Eq. (10). First, one maps the system's *resilience function*, that captures its potential fixed-points x^* — desirable vs. undesirable. For random networks with arbitrary degree sequence $d^{\pm} = (d_1^{\pm}, \ldots, d_n^{\pm})^{\top}$, it is shown that the control parameter, governing the transitions between these states is determined by W_{ij} via [33]

$$\beta = \frac{\mathbf{1}^{\top}\mathbf{W}\boldsymbol{d}^{-}}{\mathbf{1}^{\top}\mathbf{W}\mathbf{1}}. \tag{35}$$

Hence, the network's dynamic response to any form of perturbation — changing weights, removing/adding nodes or links — can be reduced to the perturbation's subsequent impact on the macroscopic parameter β. This parameter, in turn, fully determines the perturbation outcome, specifically whether the system will transition to the undesired x^*.

These vulnerability results have shown the richness and complexity of the emergent behavior of infrastructure networks, and have allowed to shed light on the phenomenon of cascading failures on a network. However, in the real world, power grids, water supply systems, production networks, transportation and communication networks, and many other infrastructure networks are not isolated, but they are often mutually dependent and interdependent, whereby the failure of one of these layers may cause severe consequences on the other layers. Such complex systems of interconnected complex systems determines the so told *critical infrastructure* of a country, whose modeling and vulnerability analysis is key for many applications in our complex, hyper-connected world. In [61], the authors have proposed an approach based on the functional interconnection of different networks, to study how failures in one network affect the others. In the proposed approach, the interdependency between different systems is captured by a matrix function called *interdependence matrix*. In the example proposed in that paper, the authors investigate the interconnection of the Italian electric grid and the telecommunication network.

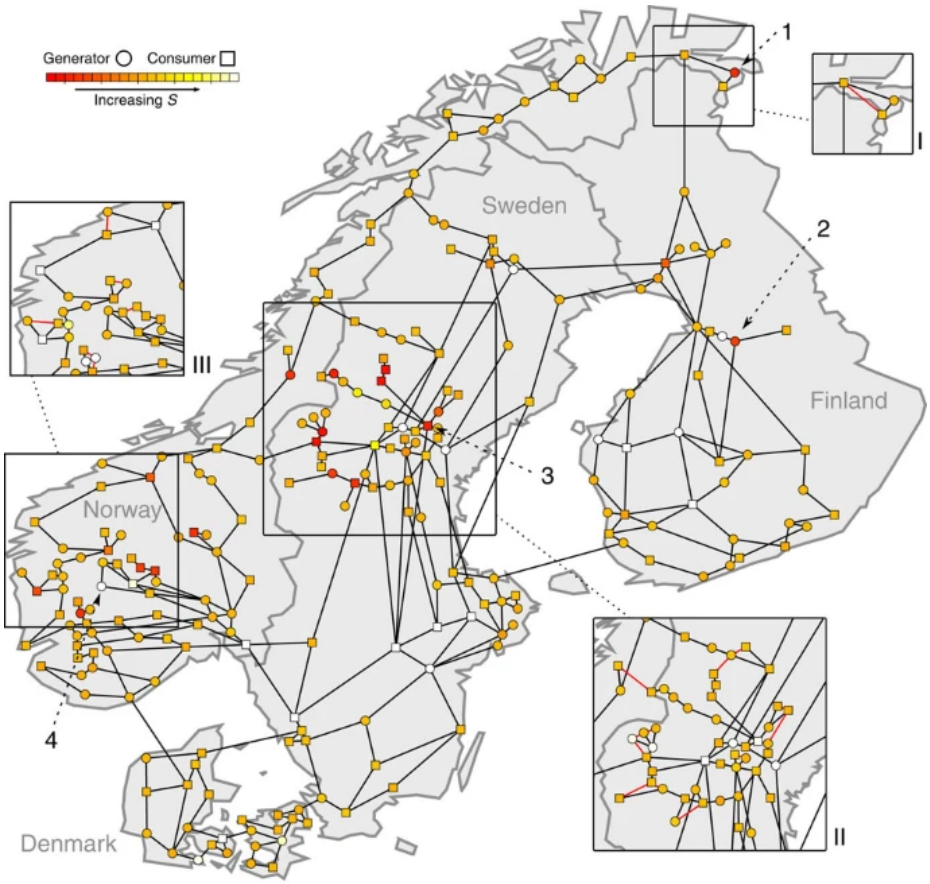

Fig. 6: Northern European power grid, composed by 236 nodes and 320 edges (reprinted from [64]).

In [62], a simple but effective model to study cascade of failures in interdependent networks has been proposed and analyzed. In the simplest scenario, two networks $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$ are considered. Each network $\mathcal{G}_i$ is characterized by its degree distribution $P_i(d)$, which associate to each number d, the probability that a node of $\mathcal{G}_i$ has degree d. The interdependence between the two networks is modeled as follows. Each node $i \in \mathcal{V}_1$ is associated with a node $\bar{i} \in \mathcal{V}_2$ through a *functional dependence*, whereby the functionality of node i depends on the one of $\bar{i}$. This represent, for instance, the fact that $\bar{i}$ supplies some critical resources to node i. Similar, each node $i \in \mathcal{V}_2$ is associated with a node in $\mathcal{V}_1$ through a functional dependence. Then, when a node fails, as a consequence all the nodes that have a functional dependence on it fail too, possibly triggering a cascade. The model can be easily generalized to multiple dependencies and networks. The authors utilize an analytical approach based on branching processes to study a system where some random nodes fail on one of the two networks, and consider different network structures, and they reach some non-trivial findings. Interest-

ingly, network structures with broad degree distribution, which exhibit a high resilience when the network is isolated, display instead high vulnerability when interconnected. Recent developments of the theory of interconnected networks are extensively discussed in the review paper [69].

4.3 Motion coordination

Motion coordination is a very important phenomenon in many biological systems, being key to the emergence of collective behaviors such as fish schooling and birds flocking. Remarkably, even though the single members of a fish school or of a bird flock are not aware of the entire state of the system, they are able to achieve global coordination by means of local interactions with other members [79]. The observation and study of these biological system have inspired the design of distributed protocols to reproduce these coordination behaviors in group of robots. Motion coordination for groups of robots finds a wide range of applications, spanning from the coordination of drones, to platooning of autonomous vehicles, and coverage problems in surveillance [38, 80, 81]. As we shall see, network science is an important tool for the analysis and the design of these systems, where global coordination emerges as an effect of local interactions.

In its simplest formulation, the problem of motion coordination can be summarized as follows. We consider a group of n robots, each one characterized by a position $x_i(t) \in \mathbb{R}^d$ and a velocity $v_i(t) \in \mathbb{R}^d$, where d is the dimension of the space in which the robots are moving. For the sake of simplicity, we introduce the scenario $d = 1$, in which the robots are moving in a uni-dimensional space. Similar to biological systems, robots coordinate their motion in a distributed fashion, that is, the system of robots is connected by the presence of a network structure $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that determines how robots can communicate, and the control action implemented on robot $i \in \mathcal{V}$ is a function of the information exchanged with its neighbors, without the intervention of a central unit. Control actions can apply a force to the robot, and thus they determine the acceleration of the robot. Hence, in its simplest form, the dynamics of robot $i \in \mathcal{V}$ is described by the following system of equations:

$$\begin{cases} \dot{x}_i(t) = v_i(t)\,, \\ \dot{v}_i(t) = u_i(t)\,, \end{cases} \tag{36}$$

where $u_i(t)$ is the control exerted on node i and is a function of $x_j(t)$ and $v_j(t)$, for $j \in \mathcal{N}_i^+$. Even though the model in Eq. (36) is quite simplistic, many robot dynamics can be simplified to this form by applying suitable non-linear feedback linearization techniques, based on the inverse dynamics of the system [82].

In [80], the authors consider the *rendezvous* problem, where a group of robots governed by Eq. (36) must coordinate in a decentralized fashion toward reaching a consensus in which $x_i(t) = x_j(t)$ and $v_i(t) = v_j(t)$, for all $i, j \in \mathcal{V}$. Note that the second task (that is, convergence of the velocities) can be easily achieved by defining a control $u_i(t)$ according to a standard consensus dynamics, that is, in the form $u_i(t) = -\sum_{j\in\mathcal{V}} L_{ij} v_i(t)$, for some matrix $\mathbf{W}$ adapted to the graph $\mathcal{G}$. However, the convergence of the velocities does not in general implies that all the positions converge. Hence, to address this problem, the authors have proposed a

second-order linear consensus protocol, in which

$$u_i(t) = -\sum_{j\in\mathcal{V}} L_{ij}(\alpha x_i(t) + \beta v_i(t))\,, \tag{37}$$

where $\mathbf{W}$ is a weight matrix adapted to $\mathcal{G}$. In plain words, the control action applied to robot i is a linear combination of a weighted average of the position and of the velocity of the neighbor robots. The case $\alpha = 0$ reduces to the consensus algorithm on the velocities, whose limitations were previously discussed. In [80], the authors establish some conditions for the dynamics in Eq. (36) to coordinate the network under the second-order linear consensus protocol in Eq. (37). Specifically, they show that the presence of a globally reachable node in $\mathcal{G}$ is a necessary condition to reach reach coordination. A sufficient condition is also proved, for the special case $\alpha = 1$. Further studies establish a necessary and sufficient condition for reaching coordination [83]. Specifically, besides the presence of a globally reachable node, the authors found that Eq. (37) succeeds in controlling the system if

$$\frac{\beta^2}{\alpha} > \max_{i=2,\dots,n} \frac{\mathcal{I}(\lambda_i)}{|\lambda_i|^2\mathcal{R}(\lambda_i)}\,, \tag{38}$$

where $\lambda_2, \dots \lambda_n$ are the non-zero eigenvalues of $\mathbf{L}$, and $\mathcal{R}(x)$ and $\mathcal{I}(x)$ denote the real and imaginary part of a complex number $x \in \mathbb{C}$, respectively. Note that also the convergence of the second-order consensus dynamics is related to the spectral properties of the network.

Rendezvous is a very important and ambitious coordination task, which is not always required. Simpler coordination tasks are of great interest for the robotics and control community. *Flocking,* for instance, requires that robots reach a coordination state in which they have the same velocities, but can maintain different positions (often with some requirements on the distances) [84]; *formation control* problems, on the other hand, require the robots to displace according to some specific shape and follow a (possibly predefined) trajectory as a rigid body [14]. In the previous discussion, we have observed that the coordination of the velocities, which is a common goal of these problems, can be directly reduced to a consensus problem in one-dimensional space. However, such a task may become non-trivial on a plane or on the 3-dimensional space, in particular when some additional constraints on the robots' positions are posed.

Here, we illustrate a simple formulation of such types of problems in a two-dimensional plane [85]. Particles with coupled oscillator dynamics constitute a valuable and successful framework to model and study collective motion on the plane. In its simplest formulation, we assume that a set of particles can move on a circle. In this setting, each node (representing a member of the group) is characterized by an oscillator, which models the position of the node as a point on the complex plane. Specifically, given the Cartesian coordinates $(x_i(t), y_i(t))$ of the position of node i at time t, such a position is encoded as the complex number $z_i(t) = x_i(t) + \imath y_i(t) = e^{\imath\theta_i(t)}$, where $\theta_i(t)$ is the phase of node i at time t, as illustrated in Figure 7. The state of each node is thus characterized by its phase $\theta_i(t)$, which evolves on the one-dimensional torus (that is, the quotient space $\mathbb{R}$ with the equivalence relation $0 = 2\pi$). Similar to the uni-dimensional model described above, the control action can apply a force to the robot, affecting its (angular)

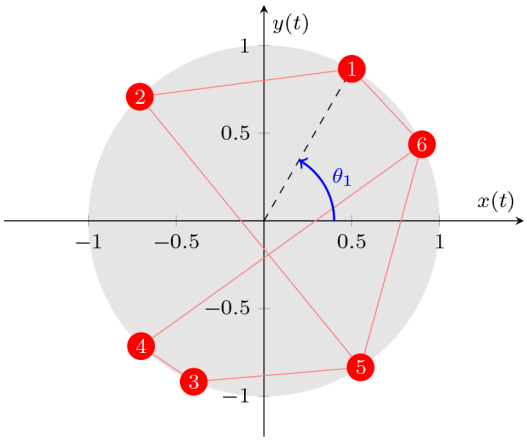

Fig. 7: Example of six coupled oscillators moving on the unit cycle. In the figure, we highlight the phase of node 1, denoted by θ_1.

velocity. Hence, the state of node i evolves according to the following system of equations:

$$\begin{cases} \dot{z}_i(t) = e^{\iota\theta_i(t)}\,, \\ \dot{\theta}_i(t) = u_i(t)\,, \end{cases} \tag{39}$$

where $u_i(t)$ is a control action that depends on $r_j(t)$ and $x_j(t)$, with $j \in \mathcal{N}_i^+$. The classical approach [85] consists of writing the control action as the sum of three contribution: a constant term ω_i, termed *natural oscillation*, a term that depends only on the phase of the other nodes $u_i^{or}(t)$, called *orientation control* and a term that depends also on the others' position $u_i^{sp}(t)$, termed *spacing control*, as

$$u_i(t) = \omega_i + u_i^{or}(t) + u_i^{sp}(t)\,. \tag{40}$$

When the goal of the controller is to coordinate the phases of the nodes, one can set the spacing control $u_i^{sp}(t) = 0$, and focus on the system

$$\dot{\theta}_i(t) = u_i(t) = \omega_i + u_i^{or}(\theta(t))\,, \tag{41}$$

which evolves on an n-dimensional torus.

Several analyses of the system in Eq. (39) have been pursued, for different shapes of the control action $u_i^{or}(\theta_i(t))$ [85]. Of particular interest is the Kuramoto model, in which it is assumed that nodes can interact through a weighted network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$, and it is set

$$u_i^{or}(\theta(t)) = K \sum_{j \in \mathcal{V}} W_{ij} \sin\big(\theta_j(t) - \theta_i(t)\big)\,. \tag{42}$$

for some constant $K > 0$ that represent the coupling strength. Most of the studies of the model have been proposed for complete networks with uniform weights, that is, $W_{ij} = 1/N$ for all $i, j \in \mathcal{V}$. See, *e.g.* this recent review for more details [86]. These results shed light on the critical role played by the coupling strength K on the emergence of synchronization, with the observation of interesting phenomena

such as explosive synchronization and the emergence of Chimera states. However, several interesting results have been established for scenarios in which the communication between nodes is constrained by the presence of network structures, relating the emergence of synchronization to the coupling strength K and to network properties such as its connectivity and its degree distribution. For more details see, for instance, [87].

So far, we have assumed that the network of interactions is not affected by the motion of the robots. However, in many real-world applications, the interactions between robots depend on the position of the robots, whereby robots often have a limited range of interaction. In these scenarios, it has been shown in [38] that the application of a standard linear consensus dynamics as a control action may cause the lost of the necessary connectivity properties, hindering thus convergence. Hence, in many applications, one of the main problem is that the network $\mathcal{G}$ is not fixed, but is evolves together with the dynamical process, and more sophisticated non-linear control actions have to be designed to guarantee that the network has always some connectivity properties (*e.g.* the presence of a globally reachable node) that guarantees convergence [38]. Several extensions of the standard theory of consensus and synchronization dynamics have been proposed to deal with time-varying networks [3].

5 Ongoing research and future challenges

Network science, an emerging field, is continuously evolving, helping us uncover both the universal, as well as the system-specific connectivity patterns of real-world complex systems. While we currently have a rather strong grip on the structure of many relevant networks, we continue to seek progress on several challenges. For example, networks that host different types of nodes, or ones whose links vary with time. Another important challenge is to systematically translate our advances on network structure into predictions on their actual observed dynamic behavior. Together, we wish to exploit the powerful toolbox of network science towards practical modern day applications, ultimately, aiming to *understand, predict and control out most crucial complex technological systems.*

5.1 Networks with adversarial and malicious nodes

A common feature of all the processes described in this chapter is the presence of cooperative dynamics between the nodes. In the consensus dynamics described in Section 3.1, nodes cooperate to reach a common state, and such a modeling framework is utilized to design distributed sensing algorithms (Section 4.1). In the synchronization problem (in Section 3.2), the presence of a cooperative coupling between a group of dynamical systems lead to the emergence of collective network behaviors, as also discussed in the applications illustrated in Section 4.3. However, many real-world applications witness the presence of not only cooperative interactions, but also of antagonistic interactions. These antagonistic interactions model, for instance, malicious attacks to distributed sensing systems, or the presence of non-cooperative robots in multi-agent systems. Signed networks have emerged as a powerful tool to represent and study the presence of cooperative and antagonistic

interactions. In a signed network, the edge set $\mathcal{E}$ is partitioned into two complementary subsets: the *positive edge set* $\mathcal{E}^+$, which models cooperative interactions, and the *negative edge set* $\mathcal{E}^-$, which models antagonistic interactions. The use of signed networks has started becoming popular in the 1940s, within the social psychology literature, with the definition of the important concept of *structural balance* [88]. A signed network is structurally balanced if and only if its nodes can be partitioned into two sets, with all positive edges connecting the nodes within each set and negative edges connecting the nodes between the two sets.

In the last decade, the study of dynamical processes on signed networks has started growing in popularity, in particular for its several insightful applications to biological and social systems [89]. The consensus dynamics on signed graphs has been originally proposed and analyzed in a seminal paper by C. Altafini [19]. In this paper, the consensus on signed networks is defined in a continuous-time framework by means of the signed Laplacian matrix of a graph. Specifically, given a stochastic weight matrix $W \in \mathbb{R}_+^{n\times n}$ (which represent the strength of the interactions between nodes), and the positive and negative edge sets $\mathcal{E}^+$ and $\mathcal{E}^-$ (which represents the type of such interactions — either cooperative or antagonistic), the signed Laplacian matrix $\bar{\mathbf{L}}$ is defined as

$$\bar{L}_{ij} = \begin{cases} -W_{ij} & \text{if } i \neq j \text{ and } (i,j) \in \mathcal{E}^+ \,, \\ W_{ij} & \text{if } i \neq j \text{ and } (i,j) \in \mathcal{E}^- \,, \\ \sum_{j\neq i} W_{ij} & \text{if } i = j \,, \\ 0 & \text{if } i \neq j \text{ and } (i,j) \notin \mathcal{E} \,. \end{cases} \tag{43}$$

The consensus dynamics is then defined as

$$\dot{\boldsymbol{x}}(t) = -\bar{\mathbf{L}}\boldsymbol{x}(t) \,. \tag{44}$$

Note that, different from the standard Laplacian matrix, the terms corresponding to negative edges appear in the Laplacian with a positive sign. Hence, nodes instead of averaging their state with the ones of their neighbors, they move away from those with whom they are connected through a negative edge. The theoretical analysis of the consensus on signed networks, initially performed in [19] for undirected networks via a Lyapunov argument, and then extended to more general cases, including directed networks [90], has highlighted the importance of the graph-theoretic concept of structural balance. In fact, for strongly connected networks, it has been shown that, if the signed network is structurally balanced, then the state of the nodes converge to a *bipartite consensus* (see Figures 8a and 8b), in which the state of all nodes converge to the same quantity in absolute value, but with different sign. Otherwise, if the network is not structurally balanced, the state of all the nodes converge to the consensus point $x^* = 0$ (see Figures 8c and 8d). Extensions of these results, accounting for time- varying network topologies and for more complex dynamics including synchronization of dynamical systems and stochastic dynamics have been proposed and analyzed, see, *e.g.* the review paper [89]. These results have allowed to understand the mechanisms of many important phenomena that emerge whenever cooperating and antagonistic interactions present in a system, including polarization phenomena in social systems.

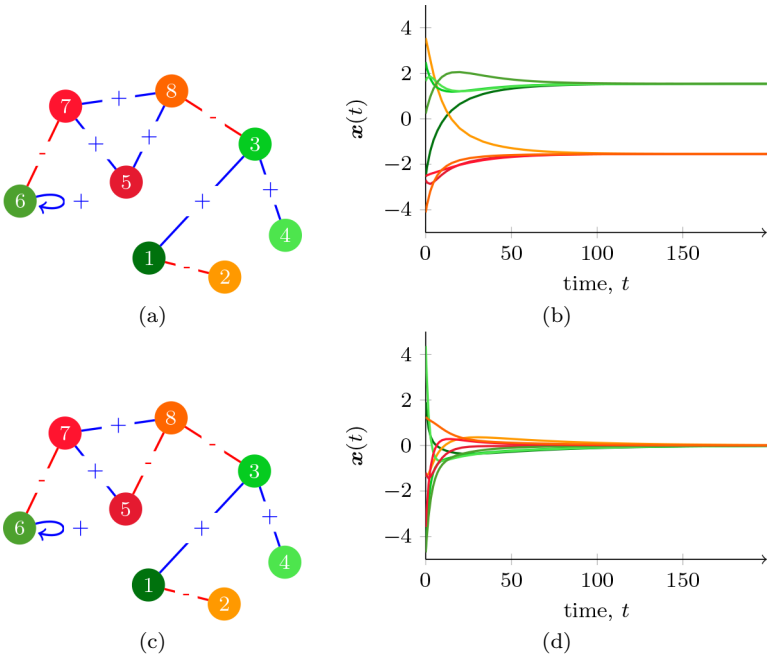

Fig. 8: Consensus dynamics on signed networks. In the networks in (a) and (c), positive edges are denoted in blue, negative edges are denoted in red. The network in (a) is structurally balanced (the two sets are $\mathcal{V}_1 = \{1, 3, 4, 6\}$ and $\mathcal{V}_2 = \{2, 5, 7, 8\}$). Hence, the corresponding trajectories of the consensus dynamics in (b) converge to a polarized equilibrium. The network in (c) is not structurally balanced and the corresponding trajectories in (d) converge to the consensus point $x^* = 0$.

5.2 Dynamics on time-varying and adaptive networks

In this chapter, we have focused our discussion on group of interacting dynamical systems whose pattern of interaction does not change in time, and thus it can be represented by a time-invariant network. However, such an assumption may be simplistic in many real-world scenarios. In motion coordination, we have already mentioned that agents may have a limited range of interactions, and thus the position of the agents may influence the network structure [38]. In sensor networks and to perform distributed estimations, we have assumed that all the sensors perform their tasks (namely, exchanging information and averaging the measurements) in a synchronized fashion, that is, according to a centralized clock. However, in many applications the sensors may perform their tasks asynchronously and in the presence of disturbances and communication errors, yielding thus a time-varying pattern of information exchange [91–94]. Furthermore, the use of adaptive networks may be a strategy to enhance the control of networks of dynamical systems and improve the performance [95,96]. These three real-world problems are just three examples of the several motivations that have called for an extension of the theory of interacting dynamical systems to dynamical network structures [97,98].

For all these reasons, the last 15 years have witnessed a growing interest in the study of dynamics evolving on time-varying networks of interactions. A time-varying network can be described by a time-invariant node set $\mathcal{V} = \{1, \dots, n\}$, a time-varying set of edges $\mathcal{E}(t) \subseteq \mathcal{V} \times \mathcal{V}$, and (possibly) a time-varying weight matrix $\mathbf{W}(t) \in \mathbb{R}_+^{n \times n}$, such that $W_{ij}(t) > 0 \iff (i,j) \in \mathcal{E}(t)$, where the network can evolve in discrete time $t \in \mathbb{Z}_+$ or in continuous time $t \in \mathbb{R}_+$. Such a growing body of literature has started developing from bunch of seminal papers written between at the beginning of the 2000s by different groups of researchers from the systems and control community [91–94]. In these papers, the consensus dynamics on time-varying networks is formalized in both the discrete- and continuous-time frameworks, and necessary and sufficient conditions on the network structure for convergence of the states of all nodes to a consensus state are derived. These results typically require that the network obtained by combining all the edge sets over a sufficiently long time time-horizon have some connectivity properties, such as being strongly connected for undirected graphs [91] or having a globally reachable node [93]. Further conditions may be required to reached desired consensus states, such as the average of the initial conditions [92]. These seminal works have laid the foundation of several lines of research, including the analysis of the performance and of the convergence times of consensus dynamics on time-varying networks, scenarios with communication noise or disturbances, and the extension of these results to more complex dynamical systems such as synchronization and motion coordination problems. Further details can be found in several surveys and books, for instance, see [98, 3].

Among these several extensions, of particular interest is the scenario in which the network of interaction evolves according to a stochastic process. In fact, the stochasticity in the network formation process may hinder the direct application of the convergence results established in the seminal papers on consensus on time-varying networks [91–94] and their more recent extensions to other dynamics. Hence, dynamics on stochastically switching networks have called for the development of a different set of tools for their analysis, grounded in the theory of stochastic dynamical systems. The main difficulty of the analysis of such problems is that the network and the nodal dynamics co-evolve, and thus the analytical approaches should be tailored to the specific properties of both dynamics and to the corresponding time-scales, hindering the possibility of developing a general theory for these dynamics. In particular, several interesting results have been established in scenarios in which the network formation dynamics is much faster than the nodal dynamics, that often goes under the name of *blinking networks*. In these scenarios, under some conditions, the stochastic network dynamics behaves like a deterministic system determined by the expectation of the stochastic variables. For these scenarios, the results for deterministic consensus and synchronization problems have been adapted and extended [99, 100]. In a regime in which the dynamics of the networks and the nodal dynamics co-evolve at comparable time scales, the aforementioned averaging techniques cannot be applied. Rigorous convergence results for the consensus problem has initially been established for simple time-varying random network structure, where each edges are present or absent with a fixed probability, independently of the others [101]. Extensions have then been developed to incorporate non-trivial correlations between edges, for network of homogeneous nodes [102]. In the last few years, activity-driven networks have been proposed as a valuable framework to perform rigorous analyses of

dynamics on time-varying networks with heterogeneous nodes [103–105]. In [106], the consensus dynamics on activity-driven networks is studied by means of a perturbation technique, establishing closed-form results on the speed of convergence that highlight that heterogeneity hinders distributed coordination.

In all these works, it is assumed that the dynamics of the network is not directly influenced by the nodal dynamics. However, in many realistic scenarios, the state of the nodes influence the network formation process. Besides the already mentioned problems related to motion coordination, in which the network connectivity is passively affected by the position of the agents in the space [38, 24, 107], an interesting scenario is that of *adaptive topologies* [108, 95], in which the state of the nodes actively affects the network formation process, whereby nodes adapt their contact in order to facilitate the coordination with others. In particular, we mention the *edge-snapping* model, proposed in [95], in which nodes implement local decentralized rules to activate and deactivate edges by means of a second-order dynamical process. In [95], the effectiveness of such an adaptive topology in synchronizing networks of dynamical systems has been proved utilizing a MSF-based approach. Based on this seminal paper, adaptive topologies have then been proposed to enhance pinning control schemes to facilitate the synchronization process [96].

5.3 Controllability of brain networks

The recent discoveries in neuroscience have allowed the scientific community to increase our understanding on the structure and the functioning of the brain [109]. The brain is a highly-complex system, formed by a large number of neurons, spanning from the 302 neurons of the *C. Elegans* — a nematode often used as a model organism by researchers— to the tens of billions for human beings. Each neuron has a non-trivial internal dynamics, while they also interact with other neurons. Such a structure has led an increasing number of researchers to investigate the brain functioning from the perspective of networks of coupled dynamical systems, starting from a seminal paper by D. S. Bassett and E. Bullmore [110]. In these works, network are used to represent the patterns of interactions within brains, where the nodes may be used to represent either single neurons or brain units that perform a specific function [109].

Among the growing body of literature on brain networks, we would like to mention the work by S. Gu *et al.* [111]. In this paper, the authors investigate the controllability of structural brain networks, utilizing an approach based on the computation of the smallest eigenvalue of the controllability Gramian, as proposed in [47] (see Section 3.4 for more details). Specifically, the authors study the controllability of a network of coupled linear dynamical systems, connected according to the structure of the brain, where each node represent an area of the brain, and its state represents the magnitude of the neurophysiological activity of the corresponding area. The results of this study provides some interesting insights into the functioning of the brain; distinct areas of the brain (with different connectivity characteristics) have been identified as critical to control the network, depending on the state that the network has to reach.

While linear dynamics may be used to study the functioning of the brain at a coarse level of brain areas, the neuronal activity is highly non-linear, calling for

the use of networks of coupled non-linear dynamical systems. Network of coupled oscillators — as the Kuramoto oscillators presented in Section 4.3— have emerged as a suitable modeling framework to study brain networks at the level of the single neurons or small groups of neurons. Of particular interest is the analysis of partial (or clustered) synchronization [112]. In fact, in a healthy functioning brain, we typically observe the emergence of cluster synchronization, where the neurons belonging to the same area of the brain have a synchronized activity, but such an activity is not synchronized with the one of neurons in other areas, while global synchronization is often observed in pathological brains of epileptic patients. Recent results on the stability of clustered synchronization for coupled non-linear dynamical systems [113] and Kuramoto oscillators [114] have help shedding light on this important phenomenon, establishing quantitative conditions on the network structure and on the coupling between the dynamical systems that guarantees the stability of cluster synchronized states, specifically highlighting the role of symmetries on the emergent behavior of cluster synchronization and, potentially, on its controllability [115,116]. These control-theoretic results might be used to help design stimulation techniques to prevent the emergence of epileptic seizures.

6 Further reading

- A. Barrat, M. Barthélemy, and A. Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, Cambridge, UK, 2008.
- M. E. J. Newman. *Networks: an introduction*. Oxford University Press, Oxford, UK, 2010.
- A.-L. Barabási. *Network Science*. Cambridge University Press, Cambridge, UK, 2016.
- V. Latora, V. Nicosia, and G. Russo. *Complex Networks: Principles, Methods and Applications*. Cambridge, UK, 2017.

Acknowledgements

The authors are indebted to Claudio Altafini, Francesco Bullo, Mario Di Bernardo, Mattia Frasca, and Sandro Zampieri for precious discussion and advice.

References

1. S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, 424(4):175–308, 2006.
2. E. Seneta. *Non-negative Matrices and Markov Chains*. Springer, New York NY, US, 1981.
3. F. Fagnani and P. Frasca. *Introduction to Averaging Dynamics over Networks*. Springer, Cham, Switzerland, 2018.
4. A. Clauset, C. R. Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703, 2009.
5. A. D. Broido and A. Clauset. Scale-free networks are rare. *Nature Communications*, 10(1):1017, Mar 2019.
6. D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

7. B. Barzel and O. Biham. Quantifying the connectivity of a network: The network correlation function method. *Physical Review E*, 80(4):046104–13, 2009.
8. B. Barzel and A.-L. Barabási. Universality in network dynamics. *Nature Physics*, 9:673–681, 2013.
9. J. R. P. French. A formal theory of social power. *Psychological Review*, 63(3):181–194, 1956.
10. M. H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.
11. P. Jia, A. MirTabatabaei, N. E. Friedkin, and F. Bullo. Opinion dynamics and the evolution of social power in influence networks. *SIAM Review*, 57(3):367–397, 2015.
12. B. D. Anderson and M. Ye. Recent Advances in the Modelling and Analysis of Opinion Dynamics on Influence Networks. *International Journal of Automation and Computing*, 16(2):129–149, 2019.
13. W. Ren and R. W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Springer, London, UK, 1 edition, 2003.
14. K. K. Oh, M. C. Park, and H. S. Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, 2015.
15. A. Olshevsky and J. N. Tsitsiklis. Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48(1):33–55, 2009.
16. L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65 – 78, 2004.
17. J. Tsitsiklis, D. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, 1986.
18. L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, 2005.
19. C. Altafini. Consensus Problems on Networks with Antagonistic Interactions. *IEEE Transactions on Automatic Control*, 58(4):935–946, 2013.
20. P. Frasca, R. Carli, F. Fagnani, and S. Zampieri. Average consensus on networks with quantized communication. *International Journal of Robust and Nonlinear Control*, 19(16):1787–1816, 2009.
21. S. Giannini, A. Petitti, D. Di Paola, and A. Rizzo. Asynchronous max-consensus protocol with time delays: Convergence results and applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(2):256–264, 2016.
22. A. Pikovsky, M. G. Rosenblum, and J. Kurths. *Synchronization, A Universal Concept in Nonlinear Sciences*. Cambridge University Press, Cambridge, UK, 2001.
23. L. M. Pecora and T. L. Carroll. Master stability functions for synchronized coupled systems. *Physical Review Letters*, 80:2109–2112, 1998.
24. M. Frasca, A. Buscarino, A. Rizzo, L. Fortuna, and S. Boccaletti. Synchronization of moving chaotic agents. *Physical Review Letters*, 100:044102, 2008.
25. M. Porfiri. A master stability function for stochastically coupled chaotic maps. *EPL (Europhysics Letters)*, 96(4):40014, 2011.
26. C. I. Del Genio, J. Gómez-Gardeñes, I. Bonamassa, and S. Boccaletti. Synchronization in networks with multiple interaction layers. *Science Advances*, 2(11):e1601679, 2016.
27. L. Tang, X. Wu, J. Lü, J.-a. Lu, and R. M. D'Souza. Master stability functions for complete, intralayer, and interlayer synchronization in multiplex networks of coupled rössler oscillators. *Physical Review E*, 99(1):012304, 2019.
28. L. V. Gambuzza, F. Di Patti, L. Gallo, S. Lepri, M. Romance, R. Criado, M. Frasca, V. Latora, and S. Boccaletti. Stability of synchronization in simplicial complexes. *Nature Communications*, 12(1):1255, 2021.
29. P. DeLellis, M. di Bernardo, and G. Russo. On quad, lipschitz, and contracting vector fields for consensus and synchronization of networks. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 58(3):576–583, 2011.
30. W. Wang and J.-J. E. Slotine. On partial contraction analysis for coupled nonlinear oscillators. *Biological Cybernetics*, 92(1):38–53, 2004.
31. U. Harush and B. Barzel. Dynamic patterns of information flow in complex networks. *Nature Communications*, 8:1–11, 2016.
32. C. Hens, U. Harush, S. Haber, R. Cohen, and B. Barzel. Spatiotemporal signal propagation in complex networks. *Nature Physics*, 15:403–412, 2019.
33. J. Gao, B. Barzel, and A.-L. Barabási. Universal resilience patterns in complex networks. *Nature*, 530:307–312, 2016.

34. P. DeLellis, M. di Bernardo, T. E. Gorochowski, and G. Russo. Synchronization and control of complex networks via contraction, adaptation and evolution. *IEEE Circuits and Systems Magazine*, 10(3):64–82, 2010.
35. Y. Tang, F. Qian, H. Gao, and J. Kurths. Synchronization in complex networks and its application – a survey of recent advances and challenges. *Annual Reviews in Control*, 38(2):184 – 198, 2014.
36. X. F. Wang and G. Chen. Pinning control of scale-free dynamical networks. *Physica A: Statistical Mechanics and its Applications*, 310(3):521 – 531, 2002.
37. C.-T. Lin. Structural controllability. *IEEE Transactions on Automatic Control*, 19(3):201–208, 1974.
38. M. Mesbahi and M. Egerstedt. *Graph theoretic methods in multiagent networks*. Princeton University Press, Princeton NJ, US, 1 edition, 2010.
39. Y. Y. Liu, J. J. Slotine, and A. L. Barabási. Controllability of complex networks. *Nature*, 473(7346):167–173, 2011.
40. X. Wang and H. Su. Pinning control of complex networked systems: A decade after and beyond. *Annual Reviews in Control*, 38(1):103 – 111, 2014.
41. W. Xing, P. Shi, R. K. Agarwal, and Y. Zhao. A survey on global pinning synchronization of complex networks. *Journal of the Franklin Institute*, 356(6):3590 – 3611, 2019.
42. F. Sorrentino, M. di Bernardo, F. Garofalo, and G. Chen. Controllability of complex networks via pinning. *Physical Review E*, 75:046103, 2007.
43. R. Carli, E. D'Elia, and S. Zampieri. A pi controller based on asymmetric gossip communications for clocks synchronization in wireless sensors networks. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 7512–7517, 2011.
44. M. Andreasson, D. V. Dimarogonas, H. Sandberg, and K. H. Johansson. Distributed control of networked dynamical systems: Static feedback, integral action and consensus. *IEEE Transactions on Automatic Control*, 59(7):1750–1764, 2014.
45. D. A. Burbano Lombana and M. di Bernardo. Distributed PID control for consensus of homogeneous and heterogeneous networks. *IEEE Transactions on Control of Network Systems*, 2(2):154–163, 2015.
46. G. Yan, J. Ren, Y.-C. Lai, C.-H. Lai, and B. Li. Controlling complex networks: How much energy is needed? *Physical Review Letters*, 108:218703, 2012.
47. F. Pasqualetti, S. Zampieri, and F. Bullo. Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, 1(1):40–52, 2014.
48. G. Lindmark and C. Altafini. Minimum energy control for complex networks. *Scientific Reports*, 8(1):1–14, 2018.
49. I. F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, 2002.
50. S. M. Ross. *Introduction to Probability Models*. Academic Press, Cambridge MA, US, 11 edition, 2014.
51. D. S. Bernstein. *Matrix Mathematics: Theory, Facts, and Formulas*. Princeton University Press, Princeton, NJ, US, 2 edition, 2011.
52. J.-C. Delvenne, R. Carli, and S. Zampieri. Optimal strategies in the average consensus problem. *Systems & Control Letters*, 58(10):759–765, 2009.
53. R. Carli, F. Fagnani, A. Speranzon, and S. Zampieri. Communication constraints in the average consensus problem. *Automatica*, 44(3):671–684, 2008.
54. R. Olfati-Saber. Ultrafast consensus in small-world networks. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 2371–2378 vol. 4, 2005.
55. A. Tahbaz-Salehi and A. Jadbabaie. Small world phenomenon, rapidly mixing markov chains, and average consensus algorithms. In *2007 46th IEEE Conference on Decision and Control*, pages 276–281, 2007.
56. R. Olfati-Saber and J. S. Shamma. Consensus filters for sensor networks and distributed sensor fusion. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 6698–6703, 2005.
57. D. P. Spanos, R. Olfati-Saber, and R. M. Murray. Approximate distributed kalman filtering in sensor networks with quantifiable performance. In *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.*, pages 133–139, 2005.
58. F. S. Cattivelli and A. H. Sayed. Diffusion strategies for distributed Kalman filtering and smoothing. *IEEE Transactions on Automatic Control*, 55(9):2069–2084, 2010.

59. D. Di Paola, A. Petitti, and A. Rizzo. Distributed kalman filtering via node selection in heterogeneous sensor networks. *International Journal of Systems Science*, 46(14):2572–2583, 2015.
60. A. Franchi, A. Petitti, and A. Rizzo. Distributed estimation of state and parameters in multiagent cooperative load manipulation. *IEEE Transactions on Control of Network Systems*, 6(2):690–701, 2018.
61. V. Rosato, L. Issacharoff, F. Tiriticco, S. Meloni, S. D. Porcellinis, and R. Setola. Modelling interdependent infrastructures using interacting dynamical models. *International Journal of Critical Infrastructures*, 4(1/2):63, 2008.
62. S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin. Catastrophic cascade of failures in interdependent networks. *Nature*, 464(7291):1025–1028, 2010.
63. A. Atputharajah and T. K. Saha. Power system blackouts - literature review. In *2009 International Conference on Industrial and Information Systems (ICIIS)*, pages 460–465, 2009.
64. P. J. Menck, J. Heitzig, J. Kurths, and H. J. Schellnhuber. How dead ends undermine power grid stability. *Nature Communications*, 5(1), 2014.
65. Y. Yang, T. Nishikawa, and A. E. Motter. Small vulnerable sets determine large network cascades in power grids. *Science*, 358(6365), 2017.
66. B. Schäfer, D. Witthaut, M. Timme, and V. Latora. Dynamically induced cascading failures in power grids. *Nature Communications*, 9(1), 2018.
67. H. Zhong and S. Y. Nof. The dynamic lines of collaboration model: Collaborative disruption response in cyber–physical systems. *Computers & Industrial Engineering*, 87:370–382, 2015.
68. D. R. Baqaee. Cascading failures in production networks. *Econometrica*, 86(5):1819–1838, 2018.
69. L. D. Valdez, L. Shekhtman, C. E. La Rocca, X. Zhang, S. V. Buldyrev, P. A. Trunfio, L. A. Braunstein, and S. Havlin. Cascading failures in complex networks. *Journal of Complex Networks*, 8(2), 2020.
70. A. E. Motter and Y.-C. Lai. Cascade-based attacks on complex networks. *Physical Review E*, 66:065102, 2002.
71. A. E. Motter. Cascade control and defense in complex networks. *Physical Review Letters*, 93(9):1–4, 2004.
72. P. Crucitti, V. Latora, and M. Marchiori. Model for cascading failures in complex networks. *Physical Review E*, 69(4):4, 2004.
73. S. Arianos, E. Bompard, A. Carbone, and F. Xue. Power grid vulnerability: A complex network approach. *Chaos*, 19(1), 2009.
74. E. Bompard, E. Pons, and D. Wu. Extended topological metrics for the analysis of power grid vulnerability. *IEEE Systems Journal*, 6(3):481–487, 2012.
75. D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters*, 85:5468–5471, 2000.
76. R. Cohen, K. Erez, D. ben Avraham, and S. Havlin. Resilience of the internet to random breakdowns. *Physical Review Letters*, 85:4626–4628, 2000.
77. R. Cohen, K. Erez, D. ben Avraham, and S. Havlin. Breakdown of the internet under intentional attack. *Physical Review Letters*, 86:3682–3685, 2001.
78. E. Bompard, R. Napoli, and F. Xue. Analysis of structural vulnerabilities in power transmission grids. *International Journal of Critical Infrastructure Protection*, 2(1-2):5–12, 2009.
79. D. J. T. Sumpter. *Collective Animal Behavior*. Princeton University Press, Princeton NJ, US, 2011.
80. W. Ren and E. Atkins. Distributed multi-vehicle coordinated control via local information exchange. *International Journal of Robust and Nonlinear Control*, 17(10-11):1002–1033, 2007.
81. F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press, Princeton NJ, US, 2009.
82. A. Isidori. *Nonlinear Control Systems*. Springer, London, UK, 3 edition, 1995.
83. W. Yu, G. Chen, and M. Cao. Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems. *Automatica*, 46(6):1089–1095, 2010.
84. R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

85. D. A. Paley, N. E. Leonard, R. Sepulchre, D. Grunbaum, and J. K. Parrish. Oscillator models and collective motion. *IEEE Control Systems Magazine*, 27(4):89–105, 2007.
86. J. Wu and X. Li. Collective synchronization of kuramoto-oscillator networks. *IEEE Circuits and Systems Magazine*, 20(3):46–67, 2020.
87. F. A. Rodrigues, T. K. D. Peron, P. Ji, and J. Kurths. The Kuramoto model in complex networks. *Physics Reports*, 610:1–98, 2016.
88. F. Heider. Attitudes and cognitive organization. *The Journal of Psychology*, 21(1):107–112, 1946. PMID: 21010780.
89. G. Shi, C. Altafini, and J. S. Baras. Dynamics over Signed Networks. *SIAM Review*, 61(2):229–257, 2019.
90. W. Xia, M. Cao, and K. H. Johansson. Structural balance and opinion separation in trust–mistrust social networks. *IEEE Transactions on Control of Network Systems*, 3(1):46–56, 2016.
91. A. Jadbabaie, Jie Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.
92. R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.
93. W. Ren and R. W. Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, 50(5):655–661, 2005.
94. L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, 2005.
95. P. DeLellis, M. diBernardo, F. Garofalo, and M. Porfiri. Evolution of complex networks via edge snapping. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 57(8):2132–2143, 2010.
96. P. DeLellis, M. di Bernardo, and M. Porfiri. Pinning control of complex networks via edge snapping. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(3):033119, 2011.
97. I. Belykh, M. di Bernardo, J. Kurths, and M. Porfiri. Evolving dynamical networks. *Physica D: Nonlinear Phenomena*, 267:1–6, 2014. Evolving Dynamical Networks.
98. S. S. Kia, B. V. Scoy, J. Cortes, R. A. Freeman, K. M. Lynch, and S. Martinez. Tutorial on dynamic average consensus: The problem, its applications, and the algorithms. *IEEE Control Systems*, 39(3):40–72, 2019.
99. I. V. Belykh, V. N. Belykh, and M. Hasler. Blinking model and synchronization in small-world networks with a time-varying coupling. *Physica D: Nonlinear Phenomena*, 195(1):188–206, 2004.
100. M. Porfiri. Stochastic synchronization in blinking networks of chaotic maps. *Physical Review E*, 85:056114, 2012.
101. Y. Hatano and M. Mesbahi. Agreement over random networks. *IEEE Transactions on Automatic Control*, 50(11):1867–1872, 2005.
102. N. Abaid, I. Igel, and M. Porfiri. On the consensus protocol of conspecific agents. *Linear Algebra and its Applications*, 437(1):221–235, 2012.
103. N. Perra, B. Gonçalves, R. Pastor-Satorras, and A. Vespignani. Activity driven modeling of time varying networks. *Scientific Reports*, 2(1):469, 2012.
104. L. Zino, A. Rizzo, and M. Porfiri. Continuous-time discrete-distribution theory for activity-driven networks. *Physical Review Letters*, 117:228302, 2016.
105. L. Zino, A. Rizzo, and M. Porfiri. An analytical framework for the study of epidemic models on activity driven networks. *Journal of Complex Networks*, 5(6):924–952, 2017.
106. L. Zino, A. Rizzo, and M. Porfiri. Consensus over activity-driven networks. *IEEE Transactions on Control of Network Systems*, 7(2):866–877, 2020.
107. M. Frasca, A. Buscarino, A. Rizzo, and L. Fortuna. Spatial pinning control. *Physical review letters*, 108(20):204102, 2012.
108. C. Zhou and J. Kurths. Dynamical weights and enhanced synchronization in adaptive complex networks. *Physical Review Letters*, 96:164102, 2006.
109. E. Tang and D. S. Bassett. Colloquium: Control of dynamics in brain networks. *Reviews of Modern Physics*, 90:031003, 2018.
110. D. S. Bassett and E. Bullmore. Small-world brain networks. *The Neuroscientist*, 12(6):512–523, 2006.

111. S. Gu, F. Pasqualetti, M. Cieslak, Q. K. Telesford, A. B. Yu, A. E. Kahn, J. D. Medaglia, J. M. Vettel, M. B. Miller, S. T. Grafton, and D. S. Bassett. Controllability of structural brain networks. *Nature Communications*, 6(1), 2015.
112. L. M. Pecora, F. Sorrentino, A. M. Hagerstrom, T. E. Murphy, and R. Roy. Cluster synchronization and isolated desynchronization in complex networks with symmetries. *Nature Communications*, 5(1), 2014.
113. L. V. Gambuzza, M. Frasca, and V. Latora. Distributed control of synchronization of a group of network nodes. *IEEE Transactions on Automatic Control*, 64(1):365–372, 2018.
114. T. Menara, G. Baggio, D. S. Bassett, and F. Pasqualetti. Stability conditions for cluster synchronization in networks of heterogeneous kuramoto oscillators. *IEEE Transactions on Control of Network Systems*, 7(1):302–314, 2020.
115. L. Gambuzza, M. Frasca, F. Sorrentino, L. Pecora, and S. Boccaletti. Controlling symmetries and clustered dynamics of complex networks. *IEEE Transactions on Network Science and Engineering*, 2020.
116. F. D. Rossa, L. Pecora, K. Blaha, A. Shirin, I. Klickstein, and F. Sorrentino. Symmetries and cluster synchronization in multilayer networks. *Nature Communications*, 11(1), 2020.

Author biographies

Lorenzo Zino is a Postdoctoral Researcher with the University of Groningen, The Netherlands, since 2019. He received his Ph.D. in Pure and Applied Mathematics (with honors) from Politecnico di Torino and Università di Torino (joint doctorate program), in 2018. His research interests encompass the modeling, the analysis, and the control aspects of dynamical processes over networks, applied probability, network modeling and analysis, and game theory.

Alessandro Rizzo is an Associate Professor at Politecnico di Torino, Italy, and a Visiting Professor at the NYU Tandon School of Engineering, Brooklyn NY, USA. He received his Ph.D. in Electronics and Automation Engineering and the Laurea in Computer Engineering from University of Catania, Italy, in 2000 and 1996, respectively. During his tenure in Torino, he has established the Complex Systems Laboratory, where he conducts and supervises research on complex networks and systems, dynamical systems, and robotics (https://csl.polito.it)

Baruch Barzel is an Associate Professor at Bar-Ilan University, Israel. He received his Ph.D. in physics from the Hebrew University of Jerusalem, Israel, in 2010, and went on to pursued his postdoctoral training at the Network Science Institute of Northeastern University and at the Channing Division of Network Medicine in Harvard Medical School, Boston. Today Baruch directs the Network Science Center at Bar-Ilan University, and his lab's goal is to develop the statistical physics of network dynamics (https://www.barzellab.com)