

---

# ADVANCES IN NETWORK CLUSTERING AND BLOCKMODELING

---

**Patrick Doreian**

University of Ljubljana, University of Pittsburgh

**Vladimir Batagelj**

University of Ljubljana, IMFM Ljubljana, IAM UP Koper, NRU HSE Moscow

**Anuška Ferligoj**

University of Ljubljana, NRU HSE Moscow



A JOHN WILEY & SONS, INC., PUBLICATION

## CHAPTER 12

---

# STRUCTURED NETWORKS AND COARSE-GRAINED DESCRIPTIONS: A DYNAMICAL PERSPECTIVE

---

MICHAEL T. SCHAUB<sup>1,2</sup>, JEAN-CHARLES DELVENNE<sup>3</sup>, RENAUD LAMBIOTTE<sup>4,5</sup>,  
AND MAURICIO BARAHONA<sup>6,7</sup>

<sup>1</sup> Institute for Data, Systems and Society, Massachusetts Institute of Technology

<sup>2</sup> Department of Engineering Sciences, University of Oxford

<sup>3</sup> ICTEAM and CORE, Université catholique de Louvain

<sup>4</sup> Mathematical Institute, University of Oxford

<sup>5</sup> naXys, University of Namur

<sup>6</sup> Department of Mathematics, Imperial College London

<sup>7</sup> EPSRC Centre for Mathematics of Precision Healthcare, Imperial College London

### 12.1 Introduction

The language of networks and graphs has become a ubiquitous tool to formalise and analyse systems and relational data across scientific disciplines, from biology to physics, from computer science to sociology [50]. Accordingly, scholars from a variety of areas have investigated such networks from different angles, developing diverse computational and mathematical toolboxes in order to analyse and ascribe meaning to the different patterns found in specific networks of interest. Modular structures are one of the most commonly studied features of networks in this context [67, 24, 74, 58, 25]. Yet, as highlighted by the lack of a common terminology (modules, partitions, blocks, communities, and clusters are but a few terms commonly found to denote various notions of modular structure in the literature), *why* scholars are interested in modular structures and *how* these structures are construed can be broadly different. Hence the perspective adopted when studying the modular structure in networks must depend on the context and specific application in mind [74, 72]. In the following, we focus on one particular motivation: namely, the rich interplay between network structure and a dynamics acting on top of the network as a means of identifying modules in the network or describing the effect that modules can have on the dynamical behaviour of a system.

**Why a dynamical perspective?** One of the main motivations for identifying modular structures in networks is that they provide a simplified, coarse-grained description of the system structure. Think of a social network, in which we might be able to decompose the system into (overlapping) groups of people such as circles of friends. We may then represent the system in terms of the interactions between these different groups, thereby reducing the complexity of the description. The hope is not only to arrive at a more compact structural description but also that the obtained modules can be interpreted as ‘building blocks’ with a functional meaning.

For instance, consider the well-known Karate Club network studied by Zachary [85], representing the social interactions between members of a Karate club that eventually split into two factions after a dispute. An interesting feature of this network is the fact that the split of the club is commensurate with the graph structure: if we apply graph partitioning methods to this network, the partition into two groups found is commonly well aligned with the split that occurred in reality. While the example of the Karate Club is by no means to be taken as a general indication of the relationship between structure and function, or between network structure and any other type of external data [55], it highlights the ultimate rationale for the detection of modules is often to gain insight into the system behaviour. For instance, we might be interested in how rumours spread in a social network, or opinions are formed. To understand such processes, we need to take into account the system structure but we also need an understanding of the *dynamics* that acts on top of this structure, since the system behaviour is the result of the interplay between the *structure* of a network *and* the *dynamics* acting on top of it. We thus aim to gain a reduced description of the system that takes into account both its structure and dynamics.

**Dynamics on networks or dynamics of networks?** We should make a distinction here between the dynamics of the network structure itself, which we call *structural dynamics* in the sequel, and dynamical processes that happen on top of a *fixed* network structure.

On the one hand, a social network can be subject to a structural dynamics over time as people become acquainted or start to dislike each other so that links and nodes appear, disappear or change weight (e.g., if we see who follows whom on twitter, who declares to be friends on Facebook, etc). The study of how these structures vary over time can be of central importance, e.g., for the spread of pathogens that can spread faster or slower depending on contact patterns. See Ref. [32] for an overview and further references on these topics.

On the other hand, data may often be naturally interpreted as a dynamics evolving and supported on a latent, unobserved fixed network. For instance, communication patterns between different people (e.g., on an online social network, an email or a mobile phone call network) may be thought as a type of point process that activates latent links at particular times [87]. The sequence of activation patterns may not be completely random at each step, but have a certain type of path dependence or memory (e.g., travellers traversing a network of flight connections from one to another city [64]). Hence, while the information recorded is temporal, the underlying network itself may be interpreted as a quasi-static object on which a path-dependent dynamics occurs.

There are of course other systems in which the dynamics on the network and the structural dynamics of the network influence each other leading to an evolution of the network structure that reflects the prevalent dynamic patterns on it. For instance, neuronal networks are known to have high plasticity and adjust their weight structure (links) based on the activity of their nodes (neurons), a feature that is commonly associated with learning.

Whether one should focus on structural dynamics, dynamics on top of a network, or both is therefore dependent on what the network representation aims to capture. In reality, all of these viewpoints are ultimately abstractions and thus attempts to capture different aspects of real world systems which hopefully provide additional insight into their behaviour.

**Network dynamics — the scope of this chapter** Our focus here will be on dynamical processes acting on top of networks. We thereby assume that the underlying (latent) network structure is known and approximately constant over the time scales of the observed dynamics. Hence, we largely omit the issue of structural dynamics, even though this may not be justified in certain applications. In the sequel, we will show that this approach is fruitful in many contexts, yielding insights that go beyond purely structural network analysis. While clearly important, the joint treatment of structural dynamics in conjunction with dynamics on networks has received less attention in the literature and requires a more elaborate mathematical machinery that goes beyond the scope of this chapter. Furthermore, we do not consider here the question of how and why the networks have arisen in the first place. (A reader interested in these questions may refer to some of the other chapters in this book.) We will therefore assume that the observed network is well defined, i.e., we treat it as an empirical reality with low uncertainty. The dynamical perspective adopted here is especially useful in such cases: the network is specified, but the emergent behaviour (our object of interest) might be hard to grasp due to the complexity of the system.

More explicitly, think again of the Karate club example. From a statistical perspective, one might want to answer the question of why the structure of the network is as observed. We may adopt a generative model (e.g., a stochastic blockmodel) and assume that the observed network is a random realisation from this model. We could then attempt to find a classification of the nodes such that the observed link probabilities between blocks of nodes reflect the observed block structure, hence explaining parsimoniously the main features in the data [76]. Using this perspective, we assume that if we could repeat the ‘experiment’ that created the network multiple times, the realisation of the network would be different each time, and we want our model to correspond to the simplest generative process consistent with those observations. In many circumstances this is a hypothetical question, however, as we only have access to a single observed network, and thus need to assume that our class of models (e.g., stochastic blockmodels) provides a suitable approximate depiction of all important features of the network.

Here we ask a complementary question: given the particular network we observe and an endowed dynamics taken place on it, are there partitions aligned with this process? For the Karate club this could give an indication of whether the split of the club was facilitated by how its particular network structure influenced the opinion formation process in this social network. Irrespective of the network’s genesis, these types of questions are of interest in many areas and underpin our perspective in this chapter.

**Outline of this chapter** We divide this chapter into three parts. In the first part, we introduce the general mathematical setup for the types of dynamics we consider throughout the chapter. We provide two guiding examples, namely consensus dynamics and diffusion processes (random walks), motivating their connection to social network analysis, and provide a brief discussion on the general dynamical framework and its possible extensions.

In the second part, we focus on the influence of graph structure on the dynamics taking place on the network, focussing on three concepts that allow us to gain insight into this notion. First, we describe how time scale separation can appear in the dynamics on a network as a consequence of graph structure. Second, we discuss how the presence of particular symmetries in the network give rise to invariant dynamical subspaces that can be

precisely described by graph partitions. Third, we show how this dynamical viewpoint can be extended to study dynamics on networks with signed edges, which allow us to discuss connections to concepts in social network analysis, such as structural balance.

In the third part, we discuss how to use dynamical processes unfolding on the network to detect meaningful network substructures. We then show how different such measures can be related to seemingly different methods for community detection and coarse-graining proposed in the literature. We conclude with a brief summary and highlight interesting open future directions.

Our account is geared towards conveying intuition rather than covering technical details. We provide pointers to additional literature with detailed results throughout the text.

**Notation** For simplicity, in the following we consider mainly undirected, connected graphs with  $n$  nodes (vertices) and  $m$  links (edges). Our ideas extend to directed graphs, however, and we provide appropriate references to the literature for the interested reader as we go along. The topology of a graph is encoded in the weighted adjacency matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$ , where  $A_{ij}$  is the weight of the link between node  $i$  and node  $j$ . Clearly, for an undirected graph  $\mathbf{A} = \mathbf{A}^\top$ . Typically, most graphs are unsigned (i.e.,  $A_{ij} \geq 0, \forall i, j$ ). The weighted out-degrees (or strengths) of the nodes are given by the vector  $\text{outdeg} = \mathbf{d} = \mathbf{A}\mathbf{1}$ , where  $\mathbf{1}$  is the  $n \times 1$  vector of ones. For a given vector  $\mathbf{v}$ , we will sometimes define the associated diagonal matrix  $\text{diag}(\mathbf{v})$  with elements  $v_i$  on the diagonal and zero elsewhere. For instance, we define the diagonal matrix of degrees  $\mathbf{D} = \text{diag}(\mathbf{d})$  and denote the total weight of the edges by  $w = \mathbf{1}^\top \mathbf{D} \mathbf{1} / 2 = \mathbf{1}^\top \mathbf{d} / 2$ .

The combinatorial graph Laplacian is defined as  $\mathbf{L} = \mathbf{D} - \mathbf{A}$ , while the normalised graph Laplacian is defined as  $\mathbf{L}_N = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}$ . Both these Laplacians are symmetric positive semi-definite, with a simple zero eigenvalue when the graph is connected [16, 28]. When describing diffusion processes on graphs, it is also useful to define the (asymmetric) random walk Laplacian  $\mathbf{L}_{RW} = \mathbf{D}^{-1} \mathbf{L}$ , which is isospectral with the normalised Laplacian for undirected graphs.

We will also consider *signed graphs*, where the weights  $A_{ij}$  can be positive or negative. In the case of signed graphs, we define the vector of absolute degrees  $\mathbf{d}_S = |\mathbf{A}| \mathbf{1}$ , where the absolute value is taken element-wise, with the corresponding absolute degree matrix  $\mathbf{D}_S = \text{diag}(\mathbf{d}_S)$ . For signed networks, we will define the signed Laplacian  $\mathbf{L}_S = \mathbf{D}_S - \mathbf{A}$ , which is also positive semi-definite. The signed Laplacian reduces to the combinatorial Laplacian in the case of an unsigned graph.

A (hard) partition  $\mathcal{P}$  of a graph of  $n$  nodes into  $k$  cells  $\{\mathcal{C}_i\}_{i=1}^k$  can be encoded by an indicator matrix  $\mathbf{C} \in \{0, 1\}^{n \times k}$ , with entries  $C_{ij} = 1$  if node  $i$  is part of cell  $\mathcal{C}_j$  and  $C_{ij} = 0$  otherwise. Hence the columns of  $\mathbf{C}$  are the indicator vectors  $\mathbf{c}^{(i)}$  of the cells:

$$\mathbf{C} := [\mathbf{c}^{(1)}, \dots, \mathbf{c}^{(k)}]. \quad (12.1)$$

## 12.2 Part I – Dynamics on and of networks

### 12.2.1 General setup

In its most general form, we are interested in dynamical systems of the form:

$$\dot{\mathbf{x}}(t) = \mathcal{A}(\mathbf{x}(t), t) \mathbf{x}(t) + \mathcal{B}(t) \mathbf{u}(t) \quad \mathcal{A} \in \mathbb{R}^{\ell \times \ell}, \quad \mathcal{B} \in \mathbb{R}^{\ell \times p} \quad (12.2a)$$

$$\mathbf{y}(t) = \mathcal{D}(t) \mathbf{x}(t) \quad \mathcal{D} \in \mathbb{R}^{c \times \ell}, \quad (12.2b)$$

where  $\mathbf{x} \in \mathbb{R}^\ell$ ,  $\mathbf{y} \in \mathbb{R}^c$ ,  $\mathbf{u} \in \mathbb{R}^p$  are the state, the observed state, and the input vectors of the system, respectively. Discrete time versions are also of interest [19, 20], but we will stick to the continuous time version henceforth.

In the context of networked systems, the system of ODEs Equation 12.2 arises by endowing each node with one or more state variables, whose union corresponds to the state vector  $\mathbf{x}$ . In general, the matrix  $\mathcal{A}$  is linked to the ‘network’: a time-varying, state-dependent coupling between the state variables of the agents (nodes). A set of exogenous inputs, described by the vector  $\mathbf{u}$  acts on the state variables through the input matrix  $\mathcal{B}$ . In such a system, we may not be able to observe and measure all the system states. This is captured by the fact that the output  $\mathbf{y}$  is a linear transformation of  $\mathbf{x}$ . This framework can naturally account for weighted, signed or other types of interactions. Furthermore, the fact that each node can be endowed with several state variables allows for the modelling of higher order dynamics (e.g., higher order Markov processes) [64, 65, 30, 21]. Note that this form also allows for the inclusion of exogenous inputs, a factor usually neglected in standard network analyses, although it has recently gained prominence for the problem of controlling networks [44].

The system Equation 12.2 formally describes the full coupled dynamics *of and on* a network, since the ‘network’ (encapsulated in the matrix  $\mathcal{A}(\mathbf{x}(t), t)$ ) is both state and time-dependent. However, such systems are difficult to analyze in general. When the coupling  $\mathcal{A}(t)$  is only time-dependent, the system describes the dynamics on a time-varying network. Such linear time-varying models have a long history in systems and control theory, and there is a rich literature pertaining to their analysis [35, 12] at the expense of more advanced mathematical machinery. Although a growing literature in *dynamical social network analysis* melding such concepts from control and dynamical systems with social network analysis (e.g., for opinion formation [59]) has recently emerged, such models have been comparably less studied within the scope of network theory.

**Dynamics on fixed networks** To simplify our exposition, we will here assume that the (latent) coupling remains constant over time, i.e.,  $\mathcal{A}, \mathcal{B}, \mathcal{D}$  have no explicit time dependence. This is what we have termed *dynamics on a (fixed) network*. It is important to remark that this assumption does not imply that each link is constantly activated over time, but that it is available for a potential interaction [64, 65, 30, 21]. We will also assume that  $\ell = p = c = n$ , which implies that there is only one state variable per node:

$$\dot{\mathbf{x}} = \mathcal{A} \mathbf{x} + \mathcal{B} \mathbf{u}(t) \quad \mathcal{A} \in \mathbb{R}^{n \times n}, \quad \mathcal{B} \in \mathbb{R}^{n \times n} \quad (12.3a)$$

$$\mathbf{y} = \mathcal{D} \mathbf{x} \quad \mathcal{D} \in \mathbb{R}^{n \times n}. \quad (12.3b)$$

In the following, we consider examples of this simpler form, which provide rich insights into problems of interest in practical applications. Specifically, we first consider consensus dynamics and its variants (motivated by opinion formation), followed by diffusion processes and random walk dynamics (motivated by information propagation).

### 12.2.2 Consensus dynamics

Consensus is one of the most popular and well studied dynamics on networks [33, 52, 61, 53, 84, 62], and can be thought of as a linear version of synchronisation [9, 34]. The attractiveness of consensus lies in its analytic tractability and simplicity, which nevertheless provides a good first description of some fundamental behaviours. For instance, in the socio-economic domain, consensus provides a model for opinion formation in a society of

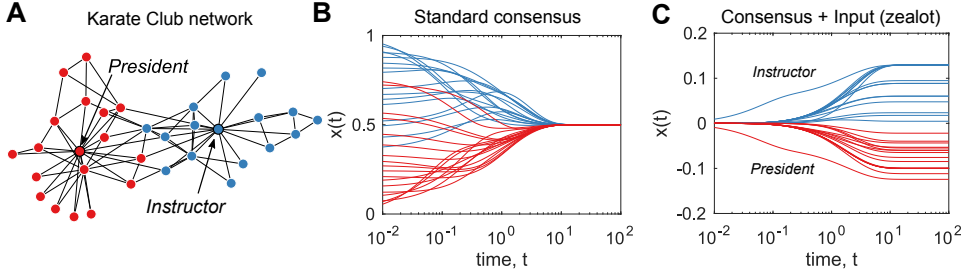


Figure 12.1: **Consensus dynamics on the Karate Club network.** **A** The Karate Club network originally analysed by Zachary [85] with nodes coloured according to the split that occurred in the real case. **B** Consensus dynamics on the Karate club network starting from a random initial condition. As time progresses, the states of the individual nodes become more aligned and eventually reach the consensus value equal to the arithmetic average of the initial condition. Note that above the time scale given by the eigenvalue  $1/\lambda_2(\mathbf{L}) \approx 1/0.47$ , the agents converge into two groups that reflect the observed split before converging to global consensus (see Section 12.3.1). **C** If an external input is applied to the system (see text), the opinion dynamics will in general not converge to a single value but lead to a dispersed set of final opinions, which still reflect the split observed in reality.

individuals, whereas in engineering systems, consensus constitutes a basic building block for efficient distributed computation of global functions in networks of sensors, robots, or other agents [33]. For a recent survey of consensus processes with a particular focus on opinion formation, we refer the reader to Proskurnikov et al. [59].

To define the standard consensus dynamics, consider a given connected network of  $n$  nodes with adjacency matrix  $\mathbf{A}$ . Let us endow each node with a scalar state variable  $x_i \in \mathbb{R}$ . The consensus dynamics on such a network is defined as:

$$\dot{\mathbf{x}} = -\mathbf{L}\mathbf{x}, \quad (\text{consensus dynamics}) \quad (12.4)$$

where  $\mathbf{L}$  is the graph Laplacian. Clearly, the consensus dynamics amounts to

$$\dot{x}_i = -D_{ii}x_i + \sum_j A_{ij}x_j = -\sum_j A_{ij}(x_i - x_j), \forall i,$$

i.e., each node adjusts its state such that the difference to its neighbours is reduced. The name of the dynamics derives from the fact that for any given initial state  $\mathbf{x}_0 = \mathbf{x}(0)$ , the differential equation Equation 12.4 drives the state to a global ‘consensus state’, where the state variables of all nodes are equal to the arithmetic average of the initial node states:  $x_i = x_* \forall i$ , where  $x_* = \mathbf{1}^\top \mathbf{x}_0 / n$  as  $t \rightarrow \infty$ . Relative to our framework Equation 12.3, the standard consensus dynamics Equation 12.4 corresponds to  $\mathcal{A} = -\mathbf{L}$ ,  $\mathcal{D} = \mathbf{I}$ ,  $\mathbf{u} = \mathbf{0}$ .

Intuitively, this dynamics may be interpreted as an opinion formation process on a network of agents who, in the absence of further inputs, will eventually agree on the same value of their state (‘opinion’), namely, the average opinion of their initial states. Figure 12.1B shows an example of the consensus dynamics on the Karate Club network starting from a random initial condition for the agents and converging asymptotically towards the common, final ‘opinion’. Yet the network structure plays a role in the form in which this opinion is approached: the opinions of each of the two factions (as recorded by their eventual split in real life) converge earlier towards a ‘group opinion’ with higher cohesion.

While in the absence of external inputs the standard consensus dynamics converges to a fixed point, the framework Equation 12.3 allows us to explore the influence of inputs over time  $\mathbf{u}(t) \neq \mathbf{0}$ , e.g., by external agents, media, etc. In that case, the asymptotic convergence of the dynamics to an eventual consensus is not guaranteed. For instance, some agents may behave like ‘zealots’, who do not update their opinion as described above, but give more weight to their own opinion [47, 60, 1]. Let us consider the Karate Club network with a constant external input:

$$\dot{\mathbf{x}} = -\mathbf{L}\mathbf{x} + \mathbf{u}, \quad (12.5)$$

with  $u_{\text{president}} = -1$  for the president,  $u_{\text{instructor}} = +1$  for the instructor, and all other nodes have no input  $u_i = 0$ . This can be thought of as a simplified model of a zealot-like behaviour of these two agents. In this case, there is no final consensus reached within the system: the final opinion of each of the agents is dispersed between the extreme positions taken by the instructor and the president (Figure 12.1C). Importantly, the final opinions of the agents are well aligned with the split that eventually occurred in the Karate club, in which half of the members joined the instructor to form a new club and the other half stayed with the president.

These results highlight how the graph properties (encapsulated by the graph Laplacian  $\mathbf{L}$ ) can shape and constrain the dynamics on the network, and thus influence the observed behaviour of the system.

**Discussion: More detailed consensus models** The consensus dynamics studied here is chosen for its simplicity. Of course, in real world systems the process of opinion formation is much more complex. For instance, opinions can be interlinked and part of a belief system [59, 26]; update and gossiping processes may be nonlinear or asynchronous [34, 53, 84]; and noisy external inputs may influence the process [83]. All these factors lead to a much more complex dynamics. In particular, opinions may not converge to a single value, or might stabilise to different values in different parts of the network. See [59] and references therein for a discussion on the so-called social cleavage problem.

### 12.2.3 Diffusion processes and random walks

Random walks are another important dynamical process which can naturally evolve on graphs. Random walks are often taken as a (simple) proxy for diffusive processes and, like consensus processes, these type of models have found applications in various domains, including information diffusion in social networks. Other applications of such processes include searching on networks, web browsing, dimensionality reduction via diffusion maps, respondent driven sampling, and, indeed, community detection [2, 46]. Perhaps the most popular example is the celebrated PageRank algorithm, which is used for the ranking of webpages, and can be seen as an application of random walks on directed networks.

The dynamics of a continuous-time unbiased random walk on a network with combinatorial Laplacian  $\mathbf{L}$  is governed by the Kolmogorov forward equation:

$$\dot{\mathbf{p}}^T = -\mathbf{p}^T \mathbf{L}_{\text{RW}} \quad (\text{random walk}) \quad (12.6)$$

where  $\mathbf{L}_{\text{RW}} = \mathbf{D}^{-1}\mathbf{L}$  is the random walk Laplacian. This equation describes the diffusion of a random particle on the network; specifically, the time evolution of the probability mass function  $\mathbf{p}(t)$  of an  $n$ -dimensional random vector  $\mathbf{x}(t)$  with components  $x_i(t) = 1$  if the particle is at node  $i$  at time  $t$  and zero otherwise. For a connected, undirected network, this dynamics will converge to the stationary distribution  $\boldsymbol{\pi} = \mathbf{d}/2w$ . For the general case of a directed graph, see [20, 40, 41].

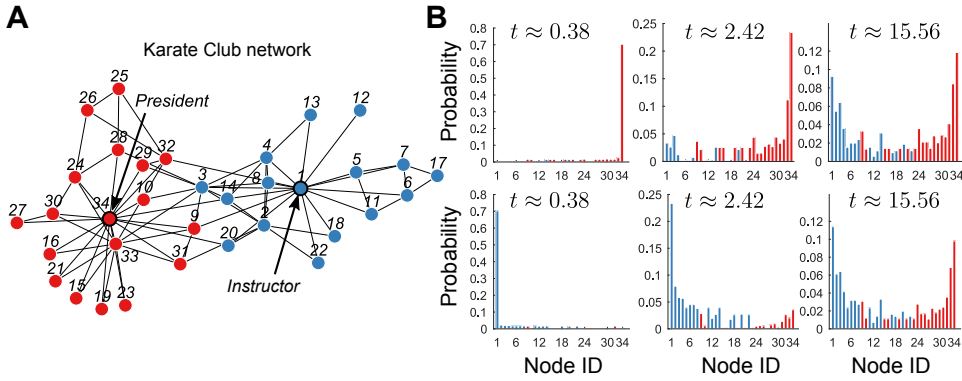


Figure 12.2: **Illustration of the evolution of a random walk dynamics on the Karate Club network.** **A** The Karate Club network with labelled nodes. The factions of the ultimate split observed in reality are indicated by color (grayscale). **B** The evolution of the probability distribution of the random walk over time exemplified at 3 time snapshots from two different initial conditions: the random walker starts at time  $t = 0$  at the ‘president’ node 34 (upper three panels), or at the ‘instructor’ node 1 (lower three panels). As time evolves, the probability of the walker to be found on the other nodes becomes more spread out on the graph and eventually reaches the stationary distribution  $\pi = \mathbf{d}/2w$ . Note that for short times, the probability is spread mostly within the corresponding ‘factions’ (i.e., president for the top panels; instructor for the bottom panels). Beyond the slowest time scale in this dynamics given by  $1/\lambda(\mathbf{L}_{RW}) \approx 1/0.13$ , the random walk becomes well mixed and the information spreads across factions (see Section 12.3.1 for more details).

An illustration of the time evolution of such a random walk on the Karate club network is displayed in Figure 12.2. As time progresses, the process converges towards the distribution  $\pi$ , which is proportional to the degree vector. It is known that the degree is also a simple centrality measure for the nodes of a graph [41], and this observed behaviour highlights how the centrality of the instructor and the president (the highest degree nodes) is also of dynamical importance. Figure 12.2B also illustrates the notion of information propagation: two random walks started one from the instructor or alternatively from the president spread initially mostly within their natural groups before eventually spreading across the whole network.

**Discussion: Random walks and consensus as dual processes** It is worth remarking that the random walk dynamics Equation 12.6 may be seen as the dual of a non-standard consensus dynamics of the form  $\dot{\mathbf{x}} = -\mathbf{D}^{-1}\mathbf{L}\mathbf{x}$ , which (unlike the standard consensus Equation 12.4) converges to a different final value: a *degree-weighted* average of the initial node-states. Conversely, a combinatorial Laplacian dynamics of the form  $\dot{\mathbf{p}}^T = -\mathbf{p}^T\mathbf{L}$  can also be seen as describing a different random walk, which is the dual of the standard consensus Equation 12.4.

### 12.3 Part II – The influence of graph structure on network dynamics

#### 12.3.1 Time scale separation in partitioned networks

**Standard time scale separation** The classic concept of time scale separation in a dynamical system follows from the simple system:

$$\frac{dx}{dt} = f(x, y), \quad (12.7a)$$

$$\varepsilon^{-1} \frac{dy}{dt} = g(x, y), \quad (12.7b)$$

where  $f, g$  are bounded functions and  $\varepsilon$  is a small constant relative to those bounds. By a simple rescaling we may assume that  $f, g$  are of order 1 and  $\varepsilon \ll 1$ . In this system,  $x(t)$  changes much more rapidly than  $y(t)$  since  $dy/dt = \varepsilon g(x, y)$ , which is small by construction. An alternative, equivalent statement follows from defining a ‘slow’ time variable  $\tau := \varepsilon t$ , hence Equation 12.7b can be rewritten as  $dy/d\tau = g(x, y)$ . From this rewriting, it follows that there is a *separation of time scales* in the dynamics, where  $y$  evolves according to the slow time scale  $\tau$ , and  $x$  evolves according to the faster  $t$ .

When a time scale separation is present in a system, it can be exploited to simplify its analysis in various ways. If we are interested in the short term behaviour of the system, we may effectively treat  $y$  as a fixed parameter and ignore its time evolution, leading to an effective one-dimensional system description based on the *fast* time scale, for instance describing the (fast) convergence of  $x$  to the fixed point  $x_*(y)$ . On the other hand, if we are interested in the long term behaviour of the system, then it is  $y$  we are most interested in. Since the dynamics of  $x$  is faster than that of  $y$ , we may assume that  $x \simeq x_*(y)$  at all times beyond an initial transient, leading to a one-dimensional system for the evolution of  $y(t)$ . Using these simplifications will, of course, lead to errors when comparing the approximation to the actual time-evolution. However, the error can be bounded through time scale separation theory.

In summary, when there are two well separated time scales in the system ( $t$  and  $\tau = \varepsilon t$ ) the dynamics ‘almost decouples’ into two different regimes: for the fast behaviour, we may simply concentrate on  $x$ , whereas for the slow, long-term behaviour we may focus on  $y$  and forget about the detailed dynamics of  $x$ . When several *distinct* time scales are present, we can similarly approximate the dynamics over particular time scales by reduced dynamics that can be obtained by finding quasi-invariant subspaces in the original system. These concepts emerge naturally in the study of networked dynamics, as we discuss below.

**Time scales in consensus dynamics on networks** Time scale separation can appear naturally in the context of dynamics on networks. For simplicity, we will describe the results here in the context of consensus dynamics, though translating these ideas to diffusion processes is straightforward.

Given an initial condition  $\mathbf{x}_0$ , linear systems theory tells us that the solution to the consensus dynamics Equation 12.4 is given by

$$\mathbf{x}(t) = \exp(-\mathbf{L}t) \mathbf{x}_0,$$

where  $\exp(\cdot)$  denotes the matrix exponential. Writing the full solution in this manner obscures the time scales present in the evolution of  $\mathbf{x}(t)$ , and how they get mixed via the network interactions represented by the Laplacian  $\mathbf{L}$ . To reveal the characteristic time

scales, we use the spectral decomposition of the Laplacian (if the graph is undirected), or a slightly more careful, yet related, treatment for directed graphs [41].

For simplicity of exposition, let us assume that the graph is undirected (i.e.,  $\mathbf{L} = \mathbf{L}^\top$ ). Let us denote the eigenvectors of  $\mathbf{L}$  by  $\mathbf{v}_i$  with associated eigenvalues  $\lambda_i$  in increasing order  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . The spectral decomposition of the Laplacian is then:

$$\mathbf{L} = \sum_{i=1}^n \lambda_i \mathbf{v}_i \mathbf{v}_i^\top.$$

Accordingly, the solution of the consensus dynamics can be written in the spectral basis as

$$\mathbf{x}(t) = \sum_{i=1}^n e^{-\lambda_i t} \mathbf{v}_i \mathbf{v}_i^\top \mathbf{x}_0 = \sum_{i=1}^n \left( \mathbf{v}_i^\top \mathbf{x}_0 \right) e^{-\lambda_i t} \mathbf{v}_i.$$

In this rewriting, the time scales of the process become apparent: they are dictated by the eigenvalues of the Laplacian matrix, with each eigenvector (eigenmode) decaying with a characteristic time scale  $\tau_i = 1/\lambda_i$ . Hence, if there are large differences (gaps) between eigenvalues, the system will have time scale separation. For instance, if the  $k$  smallest eigenvalues  $\{\lambda_1 = 0, \dots, \lambda_k\}$  are well separated from the remaining eigenvalues such that  $\lambda_k \ll \lambda_{k+1}$ , the eigenmodes associated with  $\{\lambda_{k+1}, \dots, \lambda_n\}$  become negligible for  $t > 1/\lambda_{k+1}$  and it follows that the system can be effectively described by the  $k$  smallest eigenmodes for  $t > 1/\lambda_{k+1}$ . More technically, we say that the first  $k$  eigenvectors form a dominant invariant subspace of the dynamics and there exists an associated lower dimensional ( $k < n$ ) description of the dynamics on the network which is valid after the time scale  $1/\lambda_{k+1}$ . A natural question is: how is the time scale separation that emerges from the spectral properties of the Laplacian related to the network structure? In the case of networks, time scale separation is typically associated with a lower dimensional description of the dynamics which is aligned with *localised substructures* in the graph, as we illustrate through the following example.

**Example: A modular partitioned network structure induces time scale separation**

To illustrate the discussion above, let us consider a network composed of  $k$  modules with strong in-block coupling and weaker inter-block coupling, as given by the adjacency matrix

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & & & \\ & \mathbf{A}_2 & & \\ & & \ddots & \\ & & & \mathbf{A}_k \end{pmatrix} + \mathbf{A}_{\text{random}} := \mathbf{A}_{\text{structure}} + \mathbf{A}_{\text{random}}. \quad (12.8)$$

Here  $\mathbf{A}_{\text{random}}$  can be interpreted as a realisation of an Erdős-Rényi (ER) random graph with unstructured, sparse connectivity, whereas the  $\mathbf{A}_i$  are the adjacency sub-matrices of the individual modules that have higher random connectivity inside. Here, we model these modules also as ER graphs that possess a higher connectivity probability than  $\mathbf{A}_{\text{random}}$  (Figure 12.3). How does the structure present in the graph affect the spectrum and eigenvectors of the corresponding Laplacian,  $\mathbf{L} = \mathbf{L}_{\text{structure}} + \mathbf{L}_{\text{random}}$ ?

Let us first consider the case where  $\mathbf{A}_{\text{random}} = 0$ , i.e., the graph consists of  $k$  disconnected components. In that case, it is easy to see that we have a repeated eigenvalue  $\lambda = 0$  with multiplicity  $k$  and the associated  $k$ -dimensional eigenspace can be spanned by the  $k$  indicator vectors  $\mathbf{c}^{(j)} \in \mathbb{R}^n$  defined in Equation 12.1, which are localised on the blocks in the graph.

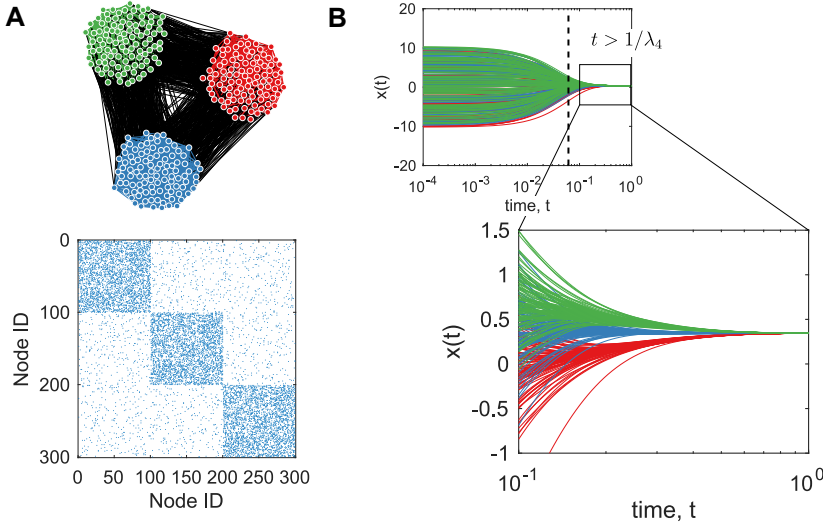


Figure 12.3: **Consensus dynamics on a structured network.** **A** Visualisation of the network and adjacency matrix of an unweighted structured network with 3 groups of the form Equation 12.8. **B** The consensus dynamics on this network displays time scale separation: after  $t \approx 1/\lambda_4 = 0.06$ , approximate consensus is reached within each group (groups indicated by color/grayscale) followed by global consensus across the whole network. A similar effect can be observed in the consensus dynamics and random walk on the Karate club network in Figures 12.1 and 12.2

To gain insight into the case where  $\mathbf{A}_{\text{random}} \neq 0$ , we invoke matrix perturbation theory and random matrix theory [82, 78, 10, 81]. For a network of the form Equation 12.8, a form of the Davis-Kahan theorem [63, 8, 42, 81] provides bounds on the angular distance between the subspace  $\mathcal{S}$  spanned by the block-vectors  $\{\mathbf{c}^{(j)}\}_{j=1}^k$  and the subspace  $\mathcal{S}'$  spanned by the eigenvectors of  $\mathbf{L}$  associated with the  $k$  smallest eigenvalues.

Intuitively, Davis-Kahan states that if the random component is small, then  $\mathcal{S} \approx \mathcal{S}'$ . The implication is that the dominant invariant subspaces will be commensurate with the structural decomposition of the network in terms of the block-vectors. Hence the long-term dynamics will directly reflect the structural decomposition of the network. In other words, the time scale separation in such a networked system takes an intuitive meaning: quasi-consensus is reached more quickly within each block, while global consensus is only reached on a longer time scale.

These points are illustrated numerically in Figure 12.3, where we show how the consensus dynamics evolves on a network of the form Equation 12.8 consisting of 3 groups with 100 nodes each. As shown in Figure 12.3B, the dynamics becomes effectively low dimensional after around  $t \simeq 1/\lambda_4$ , beyond which it is well approximated by the 3 dominant eigenmodes aligned with the intrinsic blocks in the network.

**Discussion: Time scale separation beyond homogeneous block structures** It is important to emphasise that time scale separation may also be induced by network structures that are not block-homogeneous. Many networks contain natural ‘non-clique like’ substructures (e.g., ring-shaped), which may act effectively as a dynamical substructure over

a particular time scale [69, 70]. The presence of such substructures will too affect the observed consensus dynamics on the network.

Furthermore, our discussion is not limited to the case where the structure of the network is block-diagonal, but can be extended seamlessly to networks consisting of a low-rank structure plus a random ‘noise’ component [10, 49, 56, 86]. Such networks encompass stochastic blockmodels [63, 42], although the spectral properties of a realisation of a stochastic blockmodel may not be concentrated around their expectation when the network is very sparse [48, 18, 37]. For more details on stochastic blockmodels, see [76, 6, 31] and some of the other chapters in this book.

### 12.3.2 Strictly invariant subspaces of the network dynamics and external equitable partitions

Let us now consider another type of network structure that induces a specific form of exact coarse-graining of the dynamical process acting on the network [51, 71]: the so-called *external equitable partition* (EEP).

In order to introduce EEPs, we first recall the well known graph-theoretic notion of equitable partition [28]. An equitable partition splits the graph into a set  $\{\mathcal{C}_i\}$  of non-overlapping groups of nodes called cells, that fulfil the following condition: *for each node  $v \in \mathcal{C}_i$ , the number of connections to nodes in cell  $\mathcal{C}_j$  is only dependent on  $i, j$* . Stated differently, the nodes inside each cell of an equitable partition have the same out-degree pattern with respect to every cell. The EEP is a relaxed version of the equitable partition: the requirement on equal out-degree need only hold for the number of connections between *different* cells  $\mathcal{C}_i, \mathcal{C}_j$ , where  $i \neq j$ . EPs and EEPs are closely related to so-called orbit partitions and to symmetry properties of a graph, and may be detected using tools from computational group theory [77, 54, 66]. An example of a graph with an EEP is shown in Figure 12.4A.

The presence of an EEP in a network has important consequences for a dynamics taking place on it. To see this, we consider the algebraic characterisation [13, 22, 15] of an EEP of a graph of  $n$  nodes into  $k$  cells encoded by the  $n \times k$  indicator matrix  $\mathbf{C}$  Equation 12.1. Associated with the EEP there is a quotient graph—a coarse-grained version of the original graph—such that each cell becomes a node and the weights between these new nodes are the total out-degrees between the cells in the original graph (Fig. 12.4A). It can then be verified that

$$\mathbf{L}\mathbf{C} = \mathbf{C}\mathbf{L}^\pi, \quad (12.9)$$

where  $\mathbf{L}^\pi$  is the  $k \times k$  Laplacian of the quotient graph induced by  $\mathbf{C}$ :

$$\mathbf{L}^\pi = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top \mathbf{L} \mathbf{C} = \mathbf{C}^+ \mathbf{L} \mathbf{C}. \quad (12.10)$$

Here the  $k \times n$  matrix  $\mathbf{C}^+$  is the (left) Moore-Penrose pseudoinverse of  $\mathbf{C}$ . Observe that multiplying a vector  $\mathbf{x} \in \mathbb{R}^n$  by  $\mathbf{C}^\top$  from the left sums up the components within each cell, and that  $\mathbf{C}^\top \mathbf{C}$  is a diagonal matrix with the number of nodes per cell on the diagonal. Hence,  $\mathbf{C}^+ = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top$  can be simply interpreted as a *cell averaging operator* [51]. After straightforward algebraic manipulations it is easy to show that:

$$\mathbf{C}^+ \mathbf{L} = \mathbf{L}^\pi \mathbf{C}^+, \quad (12.11)$$

which summarises the relationship between the cell averaging operator  $\mathbf{C}^+$  and the Laplacians of the original and quotient graphs. Note that although the Laplacian of the original (undirected) graph  $\mathbf{L}$  is symmetric, the Laplacian of the quotient graph  $\mathbf{L}^\pi$  will be asymmetric in general.

**Dynamical implications of EEPs** The definition of the EEP Equation 12.9 implies an invariance of the partition encoded by  $\mathbf{C}$  with respect to the Laplacian  $\mathbf{L}$ . Specifically, if we apply the Laplacian to the indicator matrix  $\mathbf{C}$  we obtain a linearly rescaled (by  $\mathbf{L}^\pi$ ) version of  $\mathbf{C}$ . A similar invariance of the cell averaging operator  $\mathbf{C}^+$  with respect to  $\mathbf{L}$  underpins Equation 12.11.

Let us expand on some of the consequences of this invariance. Equation Equation 12.9 implies that the columns of  $\mathbf{C}$  span an invariant subspace of  $\mathbf{L}$ . As the invariant subspaces of  $\mathbf{L}$  are expressible in terms of its eigenvectors, it follows that there exists a set of eigenvectors of  $\mathbf{L}$  whose components are constant on each cell of the partition. Furthermore, it can be shown that the eigenvalues associated with the eigenvectors spanning the invariant subspace are shared with  $\mathbf{L}^\pi$  [51]. If  $\mathbf{L}$  has degenerate eigenvalues, an eigenbasis can be chosen such that it is localised on the cells of the partition [51].

These algebraic properties of an EEP have implications for the dynamics dictated by  $\mathbf{L}$ , as we illustrate now for consensus dynamics [51]. First, an EEP is consistent with *partial consensus* such that the agreement within a cell (if present) is preserved. Specifically, let the initial state vector be given by  $\mathbf{x} = \mathbf{C}\mathbf{y}$  for some arbitrary  $\mathbf{y}$ , so that every node within cell  $\mathcal{C}_i$  has the same initial value  $y_i$  of their ‘opinion’ variable. Then the nodes inside each cell remain identical *for all times* and the dynamics of the cell variables is governed by the quotient graph:

$$\dot{\mathbf{x}} = \mathbf{C}\dot{\mathbf{y}} \quad \text{with} \quad \dot{\mathbf{y}} = -\mathbf{L}^\pi \mathbf{y}, \quad (12.12)$$

which follows directly from Equation 12.9. In words, if consensus has been reached within each EEP cell, then the lower dimensional Laplacian matrix  $\mathbf{L}^\pi$  (with dimensionality equal to the number of cells  $k$ ) describes the full dynamics of the system (Fig. 12.4B). This dynamical invariance thus provides a simpler model of the system.

A second consequence of the presence of an EEP is that the dynamics of the cell-averaged states  $\langle \mathbf{x} \rangle_{\mathcal{C}_i}$  is exactly described by the quotient graph:

$$\frac{d\langle \mathbf{x} \rangle_{\mathcal{C}_i}}{dt} = -\mathbf{L}^\pi \langle \mathbf{x} \rangle_{\mathcal{C}_i} \quad \text{where} \quad \langle \mathbf{x} \rangle_{\mathcal{C}_i} := \mathbf{C}^+ \mathbf{x}. \quad (12.13)$$

This dynamical coarse-graining follows directly from Equation 12.11. Hence the cell-averaged dynamics is also governed by the lower dimensional quotient Laplacian (Fig. 12.4C), and if we are only interested in such cell averages we can reduce our model significantly.

Finally, a third implication of the EEP structure relates to the system with inputs. It can be shown [51] that all the results for the autonomous consensus dynamics with no inputs can be equivalently rephrased for the system with inputs:

$$\dot{\mathbf{x}} = -\mathbf{L}\mathbf{x} + \mathbf{u}(t), \quad (12.14)$$

when the input  $\mathbf{u}(t) = \mathbf{C}\mathbf{v}(t)$ ,  $\mathbf{v}(t) \in \mathbb{R}^c$  is consistent with the cells of an EEP. In that case, the nodes inside each cell remain identical for all times, as in Equation 12.12.

**Remark:** While we have focused here on the implications of an EEP for linear consensus dynamics, invariant partitions like the EEP play a similar role for other linear and *nonlinear* dynamics (e.g., Kuramoto synchronisation). See Ref. [71] for an extended discussion including synchronisation dynamics, as well as dynamics on signed networks.

#### **Discussion: Differences and relationships between EEPs and time scale separation**

Let us discuss briefly the difference between the presence of an EEP in a network, and time

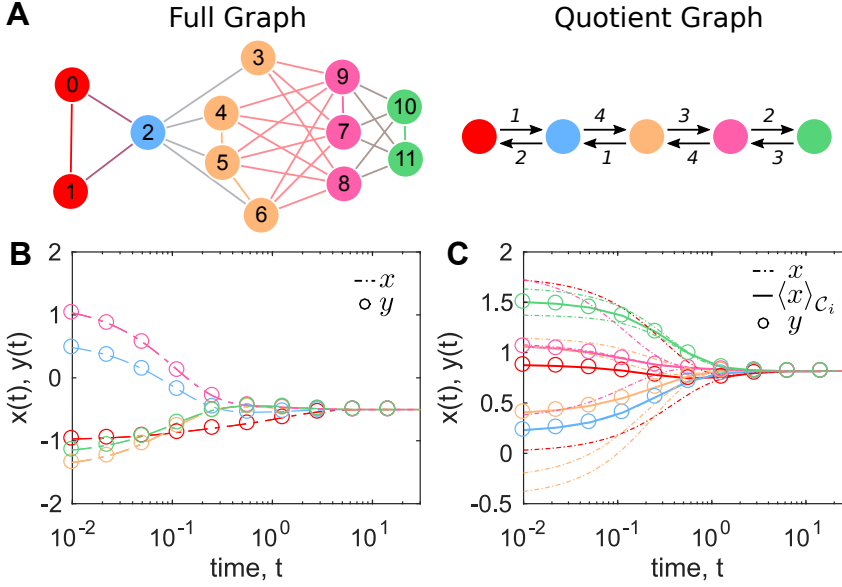


Figure 12.4: **The external equitable partition and its dynamical implications.** **A** A graph with  $n = 12$  nodes (left) with an external equitable partition into five cells (indicated with colors) and its associated quotient graph according to the EEP (right). **B** Invariance of the EEP: The consensus dynamics on the full graph Equation 12.4 from an initial condition  $\mathbf{x} = \mathbf{C}\mathbf{y}$  is shown with dash-dotted lines, whereas the associated quotient dynamics Equation 12.12 governing  $\mathbf{y}$  is shown with circles. If all states within each cell are equal (i.e., cluster-synchronised), the dynamics will always remain cluster-synchronised and are described by the dynamics of the quotient graph for all times. **C** Cell-averaging dynamics of the EEP: For consensus dynamics, the quotient graph dynamics (circles) also describes the cell-averaged dynamics (solid lines) of the unsynchronised full graph dynamics (dash-dotted lines), as given by Equation 12.13.

scale separation in the system. In our context, both concepts can be related to the notion of (strictly or almost) invariant subspaces in the dynamics. However, the link between structure and dynamics that each of them represents can be very different.

The presence of an EEP is related to symmetries in the graph, which translate into the fact that a set of Laplacian eigenvectors have components that are constant on each cell in the graph. However, these eigenvectors can be associated to any eigenvalue of the graph, whether fast or slow. In broad terms, for the EEP the ‘shape’ of the eigenvectors with respect to the cells is important, but the eigenvalues themselves are not relevant.

This notion is therefore different to the time scale separation discussed in Section 12.3.1, where the defining criterion focusses on the eigenvalues—more precisely, on the existence of gaps between eigenvalues that separate them into groups associated with different time scales. In our particular example of a planted partition model in Figure 12.3, the associated eigenvectors were indeed approximately constant on each cell (i.e., on each block of nodes) and would tend to align with the cells as the random part decreases. Hence in this case both the (approximate) EEP structure and the time scale separation are well aligned. However, this may not always be the case. We may indeed have an EEP in which the set of eigenvectors corresponding to the cells are precisely the slowest eigenmodes, but this is not necessary.

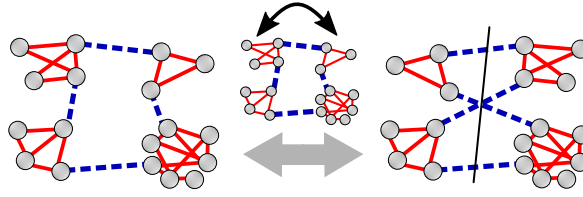


Figure 12.5: **Example of a structurally balanced graph.** A structurally balanced graph (left) with positive edges (solid, red), and negative edges (dashed, blue) can be redrawn in a different way (right) so that it can be split into two groups such that the negative edges connect nodes of different groups and links between nodes in the same group are positive.

Conversely, the eigenvectors corresponding to the slowest time scales do not have to be exactly constant on every cell, or may not be block-structured in general [69, 70]. Therefore the notions of EEP and time scale separation are distinct but not mutually exclusive.

### 12.3.3 Structural balance: consensus on signed networks and polarised opinion dynamics

**Signed networks and structural balance** In social networks, relationships can be friendly or hostile, or reflect either trust or distrust between individuals. The positive or negative character of links between agents is a central concept associated with the emergence of conflict and tension in social psychology, and has been studied within the classic literature in social network theory [29, 14]. More recently, the study of networks with *signed interactions* has gained popularity in the context of online social networks [43] and online cooperation [79]. More broadly, networks with signed edges are also essential to model biological systems and their dynamics. Examples include genes that promote or repress the expression of other genes in genetic regulatory networks [17], or neurons that can excite or inhibit the firing of other neurons in neuronal networks, thereby shaping the global dynamics of the system [27, 70].

Following Cartwright [14], a network is *structurally balanced* if the product of the signs along any closed path in the network is positive. This implies that only consistent social relationships are allowed in triangles of three nodes: either all interactions are positive, or there are exactly two negative links, which may be interpreted in the sense that “the enemy of my enemy is my friend” [29]. Equivalently, a network is structurally balanced if it can be split into two factions, where each faction contains only positive interactions internally, while the connections between the two factions are purely antagonistic [4], as illustrated in Figure 12.5. It has been shown that many social networks are empirically close to structural balance [23], suggesting that there might be a structural dynamics *driving* social networks towards structural balance [45, 80].

**Consensus dynamics on signed networks and polarisation** There are several mathematical formulations that incorporate positive and negative network interactions. Here we consider systems where the interactions are mediated through a signed Laplacian matrix defined as [4, 39]:

$$\mathbf{L}_S = \mathbf{D}_S - \mathbf{A}, \quad (12.15)$$

where  $\mathbf{D}_S = \text{diag}(|\mathbf{A}| \mathbf{1})$  is the diagonal absolute degree matrix, and the adjacency matrix  $\mathbf{A}$  can contain both positive and negative weights. Like the standard Laplacian, the signed

Laplacian  $\mathbf{L}_S$  is positive semidefinite and its spectrum contains a zero eigenvalue when the graph is connected and structurally balanced [4, 39]. To see this, note that the signed Laplacian can be decomposed as:

$$\mathbf{L}_S = \mathbf{B}\mathbf{W}_{\text{abs}}\mathbf{B}^\top, \quad (12.16)$$

where  $\mathbf{W}_{\text{abs}} = \text{diag}(|\mathbf{w}_e|)$  is the diagonal matrix containing the absolute edge weights and  $\mathbf{B} \in \mathbb{R}^{n \times m}$  is the node-to-edge incidence matrix:

$$B_{ie} = \begin{cases} 1 & \text{if } i \text{ is the tail of edge } e, \\ -\text{sign}(e) & \text{if } i \text{ is the head of edge } e. \end{cases}$$

When the graph is connected but not structurally balanced,  $\mathbf{L}_S$  is positive definite.

The following interesting alternative characterisation of a structurally balanced graph was highlighted by Altafini [4, 5]: a network is structurally balanced if there exists a  $\pm 1$  diagonal matrix  $\mathbf{\Sigma}$  such that the matrix

$$\mathbf{L}_\Sigma = \mathbf{\Sigma}\mathbf{L}_S\mathbf{\Sigma} \quad (12.17)$$

contains only negative elements on the off-diagonal. In other words,  $\mathbf{L}_\Sigma$  is a standard Laplacian matrix for another (associated) graph with only positive weights. The matrix  $\mathbf{\Sigma}$  is called switching equivalence, signature similarly, or gauge transformation in the literature [5]. Choosing  $\mathbf{\Sigma} = \mathbf{I}$  makes it clear that a network with only positive weights is itself balanced, which emphasises that  $\mathbf{L}_S$  is a proper generalisation of the standard graph Laplacian. Using this characterisation, one can efficiently determine whether a network is structurally balanced [23] and obtain the corresponding switching equivalence matrix  $\mathbf{\Sigma}$ .

We are now in a position to study the opinion dynamics on a signed network given by:

$$\dot{\mathbf{x}} = -\mathbf{L}_S\mathbf{x}. \quad (12.18)$$

The above equation is often called ‘signed consensus’ dynamics and we will adopt this name here, but it is important to note that this dynamics does not have to lead to a single consensus for all agents [4, 3, 60]. Intuitively, this should be clear: since the agents can be repelled by the opinions of their neighbours, the eventual dynamics need not converge to a unique consensus, even in the absence of external inputs. This is in contrast to the standard consensus case shown in Figure 12.1.

A remarkable feature of structurally balanced networks is that the signed consensus dynamics Equation 12.18 converges asymptotically to a *polarised state*, i.e., a state where the final value for all nodes is the same in magnitude, but the nodes are divided into two sets with opposite sign (Figure 12.5). More precisely, there is an eigenvector of  $\mathbf{L}_S$  associated with eigenvalue 0 of the form  $\boldsymbol{\sigma} = [\sigma_1, \dots, \sigma_N]^T$  where all  $\sigma_i \in \{-1, +1\}$  such that the dynamics Equation 12.18 converges to the final state

$$\lim_{t \rightarrow \infty} \mathbf{x}(t) = \frac{\boldsymbol{\sigma}^T \mathbf{x}_0}{n} \boldsymbol{\sigma}. \quad (12.19)$$

The sign pattern of  $\boldsymbol{\sigma}$  corresponds precisely to the switching equivalence transformation  $\mathbf{\Sigma} = \text{diag}(\boldsymbol{\sigma})$  and  $\sigma_i$  is denoted the polarisation of node  $i$ . Note that the overall sign of  $\boldsymbol{\sigma}$  is arbitrary, so that the polarisation of a node is only meaningful relative to that of other nodes. An example of this behaviour is shown in Figure 12.6.

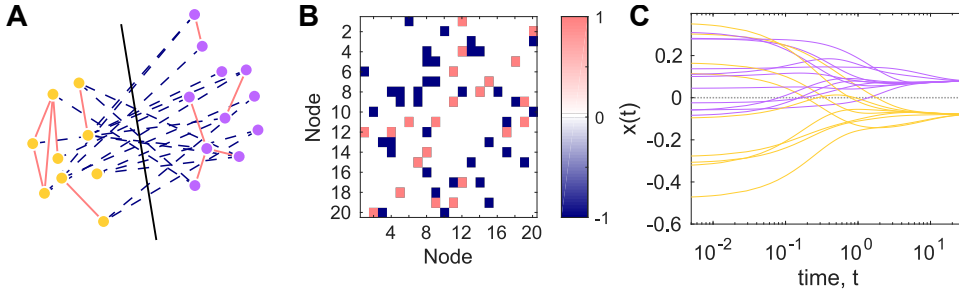


Figure 12.6: **Polarised opinion dynamics in structurally balanced networks.** **A** Visualisation of a structurally balanced network with positive (solid) and negative (dashed) links. As indicated by the drawing, the network can be partitioned into two subsets such that all negative links are between nodes in different sets. **B** The adjacency matrix of the same structurally balanced network. From this representation it is not immediately obvious whether or not the network is structurally balanced. **C** Starting from a random initial condition, the signed consensus dynamics Equation 12.18 on this structurally balanced network converges asymptotically to a state of two polarised opinions Equation 12.19, where the final opinions are exactly the same in magnitude, but opposite in sign for the two groups shown in **A**.

**Discussion: Dynamics of networks towards structural balance** Our discussions up to now have focussed mostly on the effects of graph structure on the dynamics taking place on a fixed network. Let us now briefly discuss a form of structural dynamics (where the network structure itself changes) in the context of signed networks.

Albeit commonly explained from a static perspective (i.e., parity of the cyclic paths in the network), structural balance is essentially a dynamical theory—it posits that networks tend to *evolve* towards a state of structural balance. An important question is therefore how such a structural dynamics of network evolution could look like. Antal et. al [7] provided one of the first explorations of this issue in the discrete time setting, followed by the continuous time version by Kułakowski et al. [38], which was analysed in detail by Marvel and collaborators [45]. Here we focus on the following variation of this model proposed by Traag et al. [80]:

$$\dot{\mathbf{X}} = \mathbf{X}\mathbf{X}^\top \quad \text{with} \quad \mathbf{X}(0) = \mathbf{X}_0, \quad (12.20)$$

where  $\mathbf{X} \in \mathbb{R}^{n \times n}$  denotes the matrix of opinions that agents have of each other.

It can be shown [80] that this model converges to a structurally balanced network for almost all initial conditions, i.e., we reach either a split into two factions of opposing opinions, or a state in which all nodes have positive opinion about each other. More precisely, the normalised solution of Equation 12.20 converges for some time  $t^*$  to:

$$\lim_{t \rightarrow t^*} \frac{\mathbf{X}(t)}{\|\mathbf{X}\|_F} = \mathbf{v}\mathbf{v}^\top,$$

where  $\|\cdot\|_F$  denotes the Frobenius norm and  $\mathbf{v}$  is a real-valued vector whose sign pattern indicates the two factions. However, an important shortcoming of this model is the fact that while the *sign pattern* converges to a balanced network, the *magnitude* of the opinions diverge after a finite time  $t^*$ , unless certain technical conditions are fulfilled [80]. A second shortcoming is the assumption of homogeneity in the network and in the agents: each agent

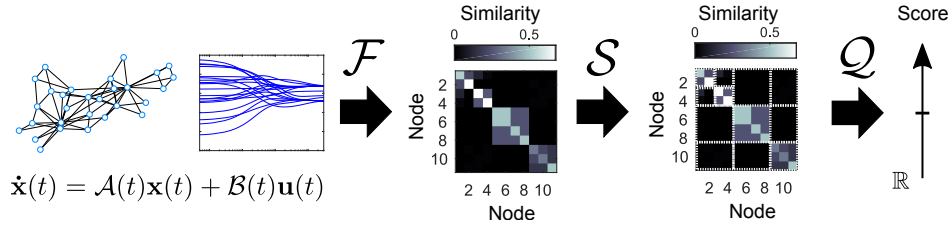


Figure 12.7: **Schematic of the algorithmic framework for partitioning of networks based on dynamics.** A dynamics  $\mathbf{x}(t)$  operating on a network with inputs  $\mathbf{u}(t)$  is filtered through the inference operator  $\mathcal{F}$  to provide a similarity matrix between the nodes, from which a partitioning (or coarse-graining) of the nodes is obtained by a selection and summary operator  $\mathcal{S}$ . The quality of this partition is summarised by a numerical metric through the quality operator  $\mathcal{Q}$ . This abstraction of dynamics-based community detection covers many existing methods in the literature, and opens up a wide array of possibilities suited to different purposes.

has an opinion about every other agent in the network (‘all-to-all’ connectivity) and all agents follow the same update rule. These assumptions are unrealistic in many real world scenarios. However, extended models that relax these simplifications are far more difficult to analyse rigorously, thus providing an interesting object for further study.

## 12.4 Part III – Using dynamical processes to reveal network structure

As we have seen in the previous sections, graph structure can have a notable impact on a dynamics acting on a network. What about the converse? Can we choose a dynamics, let it evolve on a graph, and learn a modular or coarse-grained description of the network based on some of its properties (time scale separation, quasi-invariance, etc)?

In the following, we provide a high-level, algorithmic point of view of how this may be achieved. Interestingly, many community detection methods and graph partitioning heuristics in the literature can be seen as particular cases of this general abstract viewpoint [19, 20]. Although not every method for community detection is best interpreted in terms of dynamical quantities [72], the dynamics-based perspective presented here can serve as a general framework to establish differences and similarities between the plethora of existing methods. Our perspective reveals some surprising relationships between measures and methods that have been proposed in seemingly different areas in the literature.

### 12.4.1 A generic algorithmic framework for dynamics-based network partitioning and coarse graining

Given a complex network, one important task is to find relevant communities (i.e., groups of nodes that can act as a coarse-grained description of the particular network) in an unsupervised manner. There is a very large array of such methods. However, as we will make more precise through an example below, one can interpret many of the community detection algorithms through the following generic ‘algorithmic recipe’, which includes four ingredients:

1. A dynamical process  $\mathbf{x} : t \mapsto \mathbf{x}(t)$  acting on a given network.

2. An *inference operator*  $\mathcal{F}$ , which is used to characterise some statistical information about the dynamical process.
3. A *selection and summary operator*  $\mathcal{S}$ , which assigns nodes into groups and clusters (or coarse-grains) the dynamical process.
4. A *quality function*  $\mathcal{Q}$ , which scores the clustering

While more general scenarios can be used, in the cases we discuss below these ingredients take the following form. Given a graph, we use a dynamical process  $\mathbf{x}(t)$  to assign to each node a particular time trajectory or a time-evolving probability measure. We then apply the inference operator  $\mathcal{F}$  on this time-course data to produce a node-to-node time-dependent similarity measure. The similarity measure is then aggregated according to a given grouping of nodes by the selection operator  $\mathcal{S}$ . The proposed clustering is then evaluated by the quality function  $\mathcal{Q}$ , which can then be optimised (Figure 12.7).

**Example of the framework: Markov Stability for community detection** Let us exemplify these ideas through the Markov Stability framework, as discussed in Ref. [19, 20, 69]. We use this example to demonstrate the applicability of the proposed algorithmic framework, since it can be shown that the Markov Stability framework encompasses a variety of community detection methods as particular cases [19, 20, 69], including modularity, spectral clustering, and various Potts models. For a discussion that includes processes beyond diffusion, and further relations to control-theoretic notions we refer to [73].

Markov Stability is a quality function that scores a given non-overlapping partition  $\mathcal{P}$  of a graph into node groups  $\mathcal{C}_i$ . The Markov Stability [19, 40] score is parametrically dependent on a time parameter  $t$  and is given by:

$$\text{Markov Stability}(t, \mathcal{P}) := r(t, \mathcal{P}) = \sum_{\mathcal{C}_i \in \mathcal{P}} \text{Pr}(\mathcal{C}_i, 0, t) - \text{Pr}(\mathcal{C}_i, 0, \infty) \quad (12.21)$$

where  $\text{Pr}(\mathcal{C}_i, 0, t)$  is the probability for a random walker to be in the same cell  $\mathcal{C}_i$  at time zero and at time  $t$ . For an extended discussion of this framework, see [19, 20, 69, 40, 41].

We now see how Markov Stability fits within the above four ingredient ‘recipe’:

1. To an undirected, connected graph with Laplacian  $\mathbf{L}$  we associate a diffusion dynamics (random walk) given by:

$$\dot{\mathbf{p}}^\top = -\mathbf{p}^\top \mathbf{L}_{\text{RW}}, \quad (12.22)$$

which describes the evolution of the probability mass function  $\mathbf{p}(t)$  of an  $n$ -dimensional random indicator vector  $\mathbf{x}(t)$  with  $x_k(t) = 1$  if the particle is at node  $i$  at time  $t$  and zero otherwise. Recall that  $\mathbf{L}_{\text{RW}} = \mathbf{D}^{-1}\mathbf{L}$  is the random walk Laplacian.

2. The inference operator  $\mathcal{F}$  is set to be the *autocovariance function* of this process, which computes the correlation between the random state vector  $\mathbf{x}(\tau)$  at stationarity with itself after a time  $t$ ,  $\mathbf{x}(\tau+t)$ :

$$\mathcal{F}(\mathbf{x}(t)) = \text{cov}[\mathbf{x}(\tau), \mathbf{x}(\tau+t)] = \mathbf{\Pi P}(t) - \boldsymbol{\pi}\boldsymbol{\pi}^\top. \quad (12.23)$$

$\boldsymbol{\pi} = \mathbf{d}/2w$  denotes the stationary distribution of the diffusion.

3. For a given partition of the graph with indicator matrix  $\mathbf{C}$ , we choose a selection and summary operator  $\mathcal{S}$  that aggregates the entries of an  $n \times n$  matrix defined on the nodes of the graph and aggregates the entries inside the  $k$  cells in the partition

to return a  $k \times k$  matrix. This is achieved simply by the pre- and post-multiplication  $\mathcal{S}(\cdot) = \mathbf{C}^\top (\cdot) \mathbf{C}$ . Applying  $\mathcal{S}$  to the autocovariance matrix Equation 12.23 yields:

$$\mathcal{S}(\mathcal{F}(\mathbf{x}(t))) = \mathbf{C}^\top (\mathbf{P}\mathbf{P}(t) - \boldsymbol{\pi}\boldsymbol{\pi}^\top) \mathbf{C} := \mathbf{R}(t, \mathbf{C}), \quad (12.24)$$

where  $\mathbf{R}(t, \mathbf{C})$  is the *clustered autocovariance* of the dynamics on the network. Due to the linearity of the chosen dynamics and the selection operator, the clustered autocovariance matrix is also the autocovariance of the coarse-grained indicator vectors  $\mathbf{y}(t) = \mathbf{C}^\top \mathbf{x}(t)$  [20]. Whence, with a slight abuse of notation, we conclude that in this case:

$$\mathcal{S}(\mathcal{F}(\mathbf{x}(t))) = \mathcal{F}(\mathcal{S}(\mathbf{x}(t))) = \mathbf{R}(t, \mathbf{C}).$$

4. The quality function in Markov Stability is chosen as  $\mathcal{Q}(\cdot) = \text{trace}(\cdot)$ , i.e., the trace of the matrix. When applied to the clustered autocovariance matrix  $\mathbf{R}(t, \mathbf{C})$ , it sums over the  $k$  autocovariances of each group for a given time  $t$ . Hence, the quality of the partition is the sum of the contributions from the individual groups.

In summary, Markov Stability can be written suggestively as the following function composition:

$$r(t, \mathbf{C}) = \mathcal{Q}(\mathcal{S}(\mathcal{F}(\mathbf{x}(t)))) = \text{trace } \mathbf{C}^\top (\mathbf{P}\mathbf{P}(t) - \boldsymbol{\pi}\boldsymbol{\pi}^\top) \mathbf{C}. \quad (12.25)$$

It is easy to verify that this expression is equivalent to Equation 12.21. To find ‘good’ partitions, our objective will be to maximise  $r(t, \mathbf{C})$  over the space of partitions  $\{\mathcal{P}\}$  for each value of  $t$ . This is usually a hard combinatorial optimisation, which is approached computationally through different heuristics.

## 12.4.2 Extending the framework by using other measures

Thinking of community detection as the composition of operators immediately opens up a range of extensions by varying the individual ingredients. We discuss now some of these possibilities, focussing mainly on  $\mathcal{F}$  and  $\mathcal{Q}$ , the inference and quality operators—the choice of dynamical process  $\mathbf{x}(t)$  has been discussed at length in Refs. [19, 40, 41]. Considering these extensions allows us to establish further connections with other community detection methods and heuristics proposed in the literature.

**12.4.2.1 Varying the dynamical process  $\mathbf{x}(t)$**  From our discussion above, it is clear that different dynamical processes will lead to different community detection and coarse-graining algorithms. We do not dwell on this topic here, as extended studies in [40, 20, 41] have discussed a variety of measures based on different dynamics (discrete and continuous time), and their connection with a variety of methods and heuristics proposed in the literature. Notably, our approach leads naturally to dynamical interpretations of different variants of modularity based on distinct statistical null models which can be understood both as stationary points of the dynamics and as centrality measures of the network. Similarly, generalised modularity measures, such as Potts models, can be understood as linearisations / linear interpolations of Markov Stability [41, 20]. Many further extensions are possible and, in particular translating these ideas to non-Markovian dynamics appears to be a fruitful avenue for future research.

**12.4.2.2 Varying the inference operator  $\mathcal{F}$**  A well known issue with the covariance is the fact that its absolute value is not easy to interpret, as it depends on the magnitude of the random variables involved (e.g., a simple rescaling of the variables can increase or decrease the covariance). A standard way to discount the magnitude of the random variables is to normalise them leading to *correlation measures*. A classic example is the Pearson correlation, in which the random variables are rescaled to have unit variance.

Accordingly, we can change the inference operator to  $\mathcal{F} = \text{corr}(\cdot)$ , where the cross-correlation operator between two random vectors  $\mathbf{x}_1, \mathbf{x}_2$  is defined as:

$$\text{corr}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{S}_{x_1}^{-1/2} \left( \mathbb{E}(\mathbf{x}_1 \mathbf{x}_2^\top) - \mathbb{E}(\mathbf{x}_1) \mathbb{E}(\mathbf{x}_2)^\top \right) \mathbf{S}_{x_2}^{-1/2}. \quad (12.26)$$

Here  $\mathbf{S}_{x_1}$  and  $\mathbf{S}_{x_2}$  are diagonal matrices containing the variances of the components of the vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$ , respectively, and  $\mathbb{E}$  denotes the expectation operator. Substituting the inference operator in this way, leads to a correlation-based equivalent of Markov Stability:

$$\begin{aligned} r_{\text{corr}}(t, \mathbf{C}) &= \text{trace } \mathbf{C}^\top \text{corr}(\mathbf{x}_1(\tau), \mathbf{x}_1(\tau+t)) \mathbf{C} \\ &= \text{trace } \mathbf{C}^\top \mathbf{S}^{-1/2} \left( \mathbf{\Pi} \mathbf{P}(t) - \mathbf{\pi} \mathbf{\pi}^\top \right) \mathbf{S}^{-1/2} \mathbf{C}, \end{aligned} \quad (12.27)$$

where (like in Markov Stability) we assume that the random diffusion process is at stationarity, and thus we need to normalise by  $\mathbf{S} = \mathbf{\Pi}(\mathbf{I} - \mathbf{\Pi})$ , the diagonal matrix with variances of Bernoulli random variables of mean  $\pi_i$  on the diagonal.

Interestingly, using correlation instead of covariance was considered by Shen et al. [75] in the context of modularity community detection for directed graphs. Their simplified formula for undirected networks is:

$$\text{Corr. Modularity} = \sum_{\mathcal{C}} \sum_{i,j \in \mathcal{C}} \frac{A_{ij} - \frac{d_i d_j}{2w}}{\sqrt{d_i \left(1 - \frac{d_i}{2w}\right)} \sqrt{d_j \left(1 - \frac{d_j}{2w}\right)}}. \quad (12.28)$$

It is easy to see that this measure (proposed in [75] from a combinatorial viewpoint) corresponds to the Markov Stability with dynamics governed by a *discrete-time* random walk with transition matrix  $\mathbf{P}(t) = (\mathbf{D}^{-1} \mathbf{A})^t$  evaluated at time  $t = 1$ , and with the Pearson correlation as the inference operator  $\mathcal{F}$ :

$$r_{\text{corr}}(1, \mathbf{C}) = \text{trace } \mathbf{C}^\top \mathbf{S}^{-1/2} \left[ \mathbf{\Pi} \mathbf{D}^{-1} \mathbf{A} - \mathbf{\pi} \mathbf{\pi}^\top \right] \mathbf{S}^{-1/2} \mathbf{C} = \text{Corr. Modularity}.$$

Indeed, in the original formulation of Markov Stability [19], it was already noted that a version of Markov Stability based on correlations is related to normalised cut for graph partitioning.

**Other extensions:** Although we have focussed here on the Pearson correlation, many other metrics are possible. A particular area of interest is the investigation of inference operators  $\mathcal{F}$  based on mutual information. It is also worth remarking that a number of variants arise from changing the order of application of the selection and inference operators, which will not commute in general  $\mathcal{S}(\mathcal{F}(\cdot)) \neq \mathcal{F}(\mathcal{S}(\cdot))$ , leading to different notions of clusterings. These directions will be the focus of future research.

**12.4.2.3 Varying the selection and summary operator  $\mathcal{S}$**  The choice of operator  $\mathcal{S}$  opens up a broad range of directions, which are beyond the scope of this chapter. We mention only one direction. Our focus here has been on *hard* graph partitions, i.e., forcing a hard split of the nodes into non-overlapping groups. We could however allow for overlapping (possibly probabilistic) membership. In terms of  $\mathcal{S}$ , this would amount to relaxing the assumptions of the indicator matrix  $\mathbf{C}$ —instead of requiring that  $C_{ij} = 1$  if node  $i$  belongs to community  $j$  and zero otherwise, we may merely require that  $\sum_j C_{ij} = 1$ . In that case, the node  $i$  may belong to multiple groups with a certain probability. This relaxation is equivalent to requiring that the matrix  $\mathbf{C}$  is row-stochastic, instead of binary, thus opening the possibility of using different optimisation techniques.

**12.4.2.4 Varying the quality function  $\mathcal{Q}$**  We now consider in some detail the choice of the quality function  $\mathcal{Q}$ , for which an array of possibilities is also available. Our algorithmic framework with the standard choices for Markov Stability for the operators are  $\mathcal{F} = \text{cov}(\cdot)$ ,  $\mathcal{S} = \mathbf{C}^\top(\cdot)\mathbf{C}$  on a diffusion dynamics  $\mathbf{x}(t)$  on the network leading to the clustered autocovariance matrix  $\mathbf{R}(t, \mathbf{C})$  given in Equation 12.24. To this matrix, we then apply the quality function  $\mathcal{Q}(\cdot) = \text{trace}(\cdot)$ , which sums over all the community autocovariances, a measure akin to taking an average of the individual autocovariances.

There are of course other options for the quality function. For instance, one could take into account the off-diagonal terms of the matrix  $\mathbf{R}(t, \mathbf{C})$  and define a metric that searches for maximally diagonal matrices. Such an alternative operator quality function would be especially relevant in conjunction with a different, nonlinear inference operator  $\mathcal{F}$  (e.g., mutual information). Another option is to define the quality of the partition in terms not of the average autocovariance over the groups, but of the ‘weakest’ group autocovariance of the partition, i.e., we would favour partitions where no ‘bad’ groups exist. This could be achieved by using the quality function

$$\mathcal{Q} = \min\{\text{diag}(\cdot)\}.$$

We consider this quality function in more detail as it provides an interesting connection with another measure for community detection proposed by Piccardi [57], as we now show.

Let us consider an unbiased random walk discrete-time dynamics  $\mathbf{x}(t)$  with transition matrix  $\mathbf{P}(t) = (\mathbf{D}^{-1}\mathbf{A})^t$  and stationary distribution  $\boldsymbol{\pi} = \mathbf{d}/2w$  and define the inference operator to be  $\mathcal{F} = \text{cov}(\cdot)$ . To make the connection clearer and to avoid a bias towards partitions that contain few communities of large size we need to set our selection operator to perform a normalised block-averaging

$$\mathcal{S}(\cdot) = (\mathbf{C}^\top \boldsymbol{\Pi} \mathbf{C})^{-1} \mathbf{C}^\top(\cdot)\mathbf{C},$$

where  $(\mathbf{C}^\top \boldsymbol{\Pi} \mathbf{C}) = \text{diag}([\pi_{\mathcal{C}_1}, \dots, \pi_{\mathcal{C}_k}])$  and  $\pi_{\mathcal{C}_i} = \sum_{\mathcal{C}_i} d_i/2w$  in this case. Thus, the  $\mathcal{S}$  operator normalises each group according to the starting probability of the diffusion process within each community. Combining these operators gives us a clustered normalised autocovariance matrix, to which we then apply the quality function  $\mathcal{Q} = \min\{\text{diag}(\cdot)\}$  to obtain the following clustering measure:

$$r_{\min}(t, \mathbf{C}) = \min_i \left( (\mathbf{C}^\top \boldsymbol{\Pi} \mathbf{C})^{-1} \mathbf{C}^\top \left( \mathbf{P}\mathbf{P}(t) - \boldsymbol{\pi}\boldsymbol{\pi}^\top \right) \mathbf{C} \right)_{ii}. \quad (12.29)$$

Piccardi [57] proposed a technique for community detection based on the idea that the outflow from a community under a random walk should be small. To that end, he defined a lumped Markov chain with transition matrix:

$$\mathbf{U} = (\text{diag}(\boldsymbol{\pi}\mathbf{C}))^{-1} \mathbf{C}^\top \boldsymbol{\Pi} \mathbf{D}^{-1} \mathbf{A} \mathbf{C}, \quad (12.30)$$

and stationary distribution  $\boldsymbol{\pi}_\ell = \boldsymbol{\pi}\mathbf{C}$ , i.e., the appropriately summed version of the stationary distribution of the original Markov process. The best split into communities (denoted an  $\alpha$ -partition) is found through the following optimisation

$$\max_{\mathbf{C}} k \quad \text{subject to} \quad U_{ii} \geq \alpha, \quad i = 1, 2, \dots, k \quad (12.31)$$

where  $\alpha \in [0, 1]$  is a parameter set *a priori* and defining the minimal amount of flow retained in each community.

It can be shown that a partition  $\mathbf{C}$  with  $r_{\min}(1, \mathbf{C}) \geq \beta_1$  is also an  $\alpha$ -partition with a guaranteed value of  $\alpha$ . This follows directly from the definitions Equation 12.29 and Equation 12.30:

$$r_{\min}(1, \mathbf{C}) \geq \beta_1 \implies U_{ii} \geq \beta_1 + [(\mathbf{C}^\top \boldsymbol{\Pi} \mathbf{C})^{-1} \mathbf{C}^\top \boldsymbol{\pi}^\top \boldsymbol{\pi} \mathbf{C}]_{ii} = \beta_1 + \pi_{\ell,i} := \gamma_i. \quad (12.32)$$

Therefore a partition with  $r_{\min}(1, \mathbf{C}) \geq \beta_1$  is also an  $\alpha$ -partition with  $\alpha = \max_i \gamma_i$ , and optimising  $r_{\min}(t, \mathbf{C})$  in the space of all partitions for any  $t$  encompasses  $\alpha$ -partitions as a special case (for  $t = 1$ ). For a more detailed discussion of these connections, see [68].

## 12.5 Discussion

This chapter has focussed on the relationship between structure and dynamics in complex networks, concentrating on how the notions of time scale separation, external equitable partitions, and structural balance are related to coarse-graining and community detection in networks. We have exemplified these concepts through consensus dynamics and random walks, both of which have been studied extensively linked to social network analysis.

Our discussion is underpinned by the fact that both time scale separation and EEPs are intimately related to the algebraic notion of invariant subspaces. Such invariance provides us with a better understanding of the emerging dynamics on a network, and highlights structural features in the network that are of dynamical importance. For instance, in a diffusion process time scale separation implies regions in the network where the diffusion is trapped far longer than one would expect. Hence, over a particular time scale, a diffusive signal emanating from a node inside such a group will likely reach nodes inside the same group, so that these nodes are *dynamically* almost decoupled from the rest of the network. In another example, the presence of certain graph symmetries in the network implies the presence of EEPs, graph partitions that are invariant under the dynamics. As a consequence, the averages of the consensus dynamics over the cells of the EEP can be described by the reduced dynamics of the lower-dimensional quotient graph.

Alternatively, instead of looking only at the impact of the structure on the dynamics, we can also ask the opposite question: can we use information gathered from dynamics taking place on the network to reveal important structures in the graph? We have proposed an algorithmic perspective to exploit this dynamics-driven route to the analysis of network structure and have shown that, interestingly, by adopting such a viewpoint we recover many methods of community detection proposed in the literature from seemingly different perspectives. This dynamical interpretation for such diverse methods enables us to place them with a unifying framework, providing additional insights that might not have been apparent from their initial definition.

**Open directions** There is a large literature on complex networks and dynamical processes acting on top of those, and hence there are many topics of interest related to the interplay between structure and dynamics that we have not discussed here.

A fruitful area to extend these ideas is to go beyond (first order) Markovian and diffusion-based dynamics. Indeed, many dynamics relevant to real world phenomena, such as epidemic spreading, are not of a diffusive type, yet being able to obtain dynamically important modules in such systems is crucial, e.g., to design effective vaccination strategies. Considering higher-order or non-Markovian models, allowing for the presence of memory, would for instance allow us to study bursty dynamics, or other path-dependent dynamical processes [21].

Furthermore, it will be important to connect the concept of dynamical modules considered here in connection with model order reduction tools formally studied in control theory. Such a link will allow us to quantify how far the modules describe the original dynamics in a precise manner.

Ultimately, a key aim is to describe not only the dynamics on the network in a modular fashion, but also to take into account changes of the network itself, and interactions between different types of dynamics acting on the network. Indeed, a key area for future work (barely considered here) is that of systems where the topology of the network changes, as in social contact networks, and the inclusion of multiple kinds of interactions within one system, formalised mathematically through multiplex networks [11, 36].

## Acknowledgements

MTS gratefully acknowledges funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie grant agreement No 702410. Most of this work was completed while MTS was at Université catholique de Louvain and University of Namur. JCD and RL acknowledge support from: FRS-FNRS; the Flagship European Research Area Network (FLAG-ERA) Joint Transnational Call "FuturICT 2.0"; and the ARC (Action de Recherche Concertée) on Mining and Optimisation of Big Data Models funded by the Wallonia- Brussels Federation. MB acknowledges funding from the EPSRC, through the award EP/N014529/1 funding the EPSRC Centre for Mathematics of Precision Healthcare. The funders had no role in the design of this study. The results presented here reflect solely the authors' views.

## REFERENCES

- [1] D. Acemoğlu, G. Como, F. Fagnani, and A. Ozdaglar. Opinion fluctuations and disagreement in social networks. *Mathematics of Operations Research*, 38(1):1–27, 2013.
- [2] D. Aldous and J. Fill. *Reversible Markov chains and random walks on graphs*. Book in preparation, 2002.
- [3] C. Altafini. Dynamics of opinion forming in structurally balanced social networks. *PloS one*, 7(6):e38135, 2012.
- [4] C. Altafini. Consensus Problems on Networks With Antagonistic Interactions. *Automatic Control, IEEE Transactions on*, 58(4):935–946, April 2013. ISSN 0018-9286. doi: 10.1109/TAC.2012.2224251.
- [5] C. Altafini. Stability analysis of diagonally equipotent matrices. *Automatica*, 49(9):2780 – 2785, 2013. ISSN 0005-1098. doi: <http://dx.doi.org/10.1016/j.automatica.2013.05.016>.
- [6] C. J. Anderson, S. Wasserman, and K. Faust. Building stochastic blockmodels. *Social networks*, 14(1):137–161, 1992.

- [7] T. Antal, P. L. Krapivsky, and S. Redner. Dynamics of social balance on networks. *Physical Review E*, 72(3):036121, 2005.
- [8] A. S. Bandeira. Random laplacian matrices and convex relaxations. *Foundations of Computational Mathematics*, pages 1–35, 2015.
- [9] M. Barahona and L. M. Pecora. Synchronization in Small-World Systems. *Phys. Rev. Lett.*, 89:054101, July 2002. doi: 10.1103/PhysRevLett.89.054101.
- [10] F. Benaych-Georges and R. R. Nadakuditi. The eigenvalues and eigenvectors of finite, low rank perturbations of large random matrices. *Advances in Mathematics*, 227(1):494–521, may 2011. doi: 10.1016/j.aim.2011.02.007.
- [11] S. Boccaletti, G. Bianconi, R. Criado, C. del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, and M. Zanin. The structure and dynamics of multilayer networks. *Physics Reports*, 544(1):1–122, nov 2014. doi: 10.1016/j.physrep.2014.07.001.
- [12] R. W. Brockett. *Finite dimensional linear systems*. SIAM, 2015.
- [13] D. M. Cardoso, C. Delorme, and P. Rama. Laplacian eigenvectors and eigenvalues and almost equitable partitions. *European Journal of Combinatorics*, 28(3):665–673, 2007.
- [14] D. Cartwright and F. Harary. Structural balance: a generalization of Heider’s theory. *Psychological Review*, 63(5):277, 1956.
- [15] A. Chan and C. D. Godsil. Symmetry and eigenvectors. In *Graph Symmetry*, pages 75–106. Springer, 1997.
- [16] F. R. Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.
- [17] E. Davidson and M. Levin. Gene regulatory networks. *Proceedings of the National Academy of Sciences*, 102(14):4935–4935, 2005.
- [18] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová. Inference and phase transitions in the detection of modules in sparse networks. *Physical Review Letters*, 107(6):065701, 2011.
- [19] J.-C. Delvenne, S. N. Yaliraki, and M. Barahona. Stability of graph communities across time scales. *Proceedings of the National Academy of Sciences*, 107(29):12755–12760, 2010. doi: 10.1073/pnas.0903215107.
- [20] J.-C. Delvenne, M. T. Schaub, S. N. Yaliraki, and M. Barahona. The Stability of a Graph Partition: A Dynamics-Based Framework for Community Detection. In A. Mukherjee, M. Choudhury, F. Peruani, N. Ganguly, and B. Mitra, editors, *Dynamics On and Of Complex Networks, Volume 2, Modeling and Simulation in Science, Engineering and Technology*, pages 221–242. Springer New York, 5 2013. ISBN 978-1-4614-6728-1. doi: 10.1007/978-1-4614-6729-8\_11.
- [21] J.-C. Delvenne, R. Lambiotte, and L. E. C. Rocha. Diffusion on networked systems is a question of time or structure. *Nature communications*, 6:7366, June 2015. ISSN 2041-1723. doi: 10.1038/ncomms8366.
- [22] M. Egerstedt, S. Martini, M. Cao, K. Camlibel, and A. Bicchi. Interacting with Networks: How Does Structure Relate to Controllability in Single-Leader, Consensus Networks? *Control Systems, IEEE*, 32(4):66–73, Aug 2012. ISSN 1066-033X. doi: 10.1109/MCS.2012.2195411.
- [23] G. Facchetti, G. Iacono, and C. Altafini. Computing global structural balance in large-scale signed social networks. *Proceedings of the National Academy of Sciences*, 108(52):20953–20958, 2011. doi: 10.1073/pnas.1109521108.
- [24] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, 2010. ISSN 0370-1573. doi: 10.1016/j.physrep.2009.11.002.
- [25] S. Fortunato and D. Hric. Community detection in networks: A user guide. *Physics Reports*, 659:1–44, 2016.

- [26] N. E. Friedkin, A. V. Proskurnikov, R. Tempo, and S. E. Parsegov. Network science on belief system dynamics under logic constraints. *Science*, 354(6310):321–326, 2016. ISSN 0036-8075. doi: 10.1126/science.aag2624. URL <http://science.sciencemag.org/content/354/6310/321>.
- [27] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.
- [28] C. Godsil and G. F. Royle. *Algebraic graph theory*, volume 207. Springer Science & Business Media, 2013.
- [29] F. Heider. Attitudes and cognitive organization. *The Journal of Psychology*, 21(1):107–112, 1946.
- [30] T. Hoffmann, M. A. Porter, and R. Lambiotte. Generalized master equations for non-poisson dynamics on networks. *Phys. Rev. E*, 86:046102, Oct 2012. doi: 10.1103/PhysRevE.86.046102.
- [31] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.
- [32] P. Holme and J. Saramäki. Temporal networks. *Physics Reports*, 519(3):97–125, 2012. ISSN 0370-1573. doi: 10.1016/j.physrep.2012.03.001.
- [33] A. Jadbabaie, J. Lin, and A. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, June 2003. ISSN 0018-9286. doi: 10.1109/TAC.2003.812781.
- [34] A. Jadbabaie, N. Motee, and M. Barahona. On the stability of the Kuramoto model of coupled nonlinear oscillators. In *Proceedings of the American Control Conference, 2004.*, volume 5, pages 4296–4301, 2004. URL [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1383983&tag=1](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1383983&tag=1).
- [35] T. Kailath. *Linear systems*, volume 156. Prentice-Hall Englewood Cliffs, NJ, 1980.
- [36] M. Kivela, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter. Multilayer networks. *Journal of Complex Networks*, 2(3):203–271, jul 2014. doi: 10.1093/comnet/cnu016.
- [37] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborová, and P. Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013.
- [38] K. Kułakowski, P. Gawroński, and P. Groniek. The heider balance: A continuous approach. *International Journal of Modern Physics C*, 16(05):707–716, 2005.
- [39] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. De Luca, and S. Albayrak. Spectral analysis of signed graphs for clustering, prediction and visualization. In *SDM*, volume 10, pages 559–559. SIAM, 2010.
- [40] R. Lambiotte, J.-C. Delvenne, and M. Barahona. Laplacian Dynamics and Multiscale Modular Structure in Networks. *arxiv:0812.1770*, pages 1–29, Oct 2009.
- [41] R. Lambiotte, J. Delvenne, and M. Barahona. Random Walks, Markov Processes and the Multiscale Modular Organization of Complex Networks. *IEEE Transactions on Network Science and Engineering*, 1(2):76–90, July 2014. ISSN 2327-4697. doi: 10.1109/TNSE.2015.2391998.
- [42] J. Lei, A. Rinaldo, et al. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1):215–237, 2015.
- [43] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed networks in social media. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1361–1370. ACM, 2010.

- [44] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási. Controllability of complex networks. *nature*, 473 (7346):167, 2011.
- [45] S. A. Marvel, J. Kleinberg, R. D. Kleinberg, and S. H. Strogatz. Continuous-time model of structural balance. *Proceedings of the National Academy of Sciences*, 108(5):1771–1776, 2011. doi: 10.1073/pnas.1013213108.
- [46] N. Masuda, M. A. Porter, and R. Lambiotte. Random walks and diffusion on networks. *arXiv:1612.03281*, Dec. 2016.
- [47] M. Mobilia, A. Petersen, and S. Redner. On the role of zealotry in the voter model. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(08):P08029, 2007. doi: 10.1088/1742-5468/2007/08/P08029.
- [48] E. Mossel, J. Neeman, and A. Sly. A proof of the block model threshold conjecture. *arXiv:1311.4115*, 2013.
- [49] R. R. Nadakuditi and M. E. J. Newman. Graph spectra and the detectability of community structure in networks. *Physical Review Letters*, 108(18), may 2012. doi: 10.1103/physrevlett.108.188701.
- [50] M. E. J. Newman. *Networks: An Introduction*. Oxford University Press, USA, Mar. 2010.
- [51] N. O’Clery, Y. Yuan, G.-B. Stan, and M. Barahona. Observability and coarse graining of consensus dynamics through the external equitable partition. *Physical Review E*, 88(4), oct 2013. doi: 10.1103/physreve.88.042805.
- [52] R. Olfati-Saber and R. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520 – 1533, sept. 2004. ISSN 0018-9286. doi: 10.1109/TAC.2004.834113.
- [53] R. Olfati-Saber, J. Fax, and R. Murray. Consensus and Cooperation in Networked Multi-Agent Systems. *Proceedings of the IEEE*, 95(1):215 –233, jan. 2007. ISSN 0018-9219. doi: 10.1109/JPROC.2006.887293.
- [54] 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*, 5:4079, 2014.
- [55] L. Peel, D. B. Larremore, and A. Clauset. The ground truth about metadata and community detection in networks. *Science Advances*, 3(5):e1602548, 2017.
- [56] T. P. Peixoto. Eigenvalue spectra of modular networks. *Physical review letters*, 111(9): 098701, 2013.
- [57] C. Piccardi. Finding and Testing Network Communities by Lumped Markov Chains. *PLoS ONE*, 6(11):e27028, 11 2011. doi: 10.1371/journal.pone.0027028.
- [58] M. Porter, J. Onnela, and P. Mucha. Communities in networks. *Notices of the AMS*, 56(9): 1082–1097, 1164–1166, 2009.
- [59] A. V. Proskurnikov and R. Tempo. A tutorial on modeling and analysis of dynamic social networks. part i. *Annual Reviews in Control*, 43:65 – 79, 2017. ISSN 1367-5788. doi: 10.1016/j.arcontrol.2017.03.002.
- [60] A. V. Proskurnikov, A. S. Matveev, and M. Cao. Opinion dynamics in social networks with hostile camps: Consensus vs. polarization. *IEEE Transactions on Automatic Control*, 61(6): 1524–1536, 2016.
- [61] W. Ren, R. W. Beard, and E. M. Atkins. A survey of consensus problems in multi-agent coordination. In *American Control Conference, 2005. Proceedings of the 2005*, pages 1859–1864. IEEE, 2005.
- [62] W. Ren, R. W. Beard, and E. M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems*, 27(2):71–82, 2007.

- [63] K. Rohe, S. Chatterjee, and B. Yu. Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, pages 1878–1915, 2011.
- [64] M. Rosvall, A. V. Esquivel, A. Lancichinetti, J. D. West, and R. Lambiotte. Memory in network flows and its effects on spreading dynamics and community detection. *Nature communications*, 5:4630, Aug. 2014. ISSN 2041-1723. doi: 10.1038/ncomms5630.
- [65] V. Salnikov, M. T. Schaub, and R. Lambiotte. Using higher-order markov models to reveal flow-based communities in networks. *Scientific reports*, 6:23194, 2016.
- [66] R. J. Sanchez-Garcia. Exploiting symmetry in network analysis. *arXiv preprint arXiv:1803.06915*, 2018.
- [67] S. E. Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007. ISSN 1574-0137. doi: 10.1016/j.cosrev.2007.05.001.
- [68] M. T. Schaub. *Unraveling complex networks under the prism of dynamical processes: relations between structure and dynamics*. PhD thesis, Imperial College London, 2014.
- [69] M. T. Schaub, J.-C. Delvenne, S. N. Yaliraki, and M. Barahona. Markov dynamics as a zooming lens for multiscale community detection: non clique-like communities and the field-of-view limit. *PloS one*, 7(2):e32210, 2012.
- [70] M. T. Schaub, Y. N. Billeh, C. A. Anastassiou, C. Koch, and M. Barahona. Emergence of slow-switching assemblies in structured neuronal networks. *PLoS Comput Biol*, 11(7):e1004196, 2015.
- [71] M. T. Schaub, N. O’Clery, Y. N. Billeh, J.-C. Delvenne, R. Lambiotte, and M. Barahona. Graph partitions and cluster synchronization in networks of oscillators. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 26(9):094821, 2016.
- [72] M. T. Schaub, J.-C. Delvenne, M. Rosvall, and R. Lambiotte. The many facets of community detection in complex networks. *Applied Network Science*, 2(1):4, 02 2017. ISSN 2364-8228. doi: 10.1007/s41109-017-0023-6.
- [73] M. T. Schaub, J.-C. Delvenne, R. Lambiotte, and M. Barahona. Multiscale dynamical embeddings of complex networks. *arXiv:1804.03733*, 2018. URL <https://arxiv.org/abs/1804.03733>.
- [74] S. Shai, N. Stanley, C. Granell, D. Taylor, and P. J. Mucha. Case studies in network community detection. *arXiv:1705.02305*, 2017.
- [75] H.-W. Shen, X.-Q. Cheng, and B.-X. Fang. Covariance, correlation matrix, and the multiscale community structure of networks. *Phys. Rev. E*, 82:016114, Jul 2010. doi: 10.1103/PhysRevE.82.016114.
- [76] T. A. Snijders. Statistical models for social networks. *Annual Review of Sociology*, 37, 2011.
- [77] F. Sorrentino, L. M. Pecora, A. M. Hagerstrom, T. E. Murphy, and R. Roy. Complete characterization of the stability of cluster synchronization in complex dynamical networks. *Science advances*, 2(4):e1501737, 2016.
- [78] G. W. Stewart. *Matrix algorithms volume 2: eigensystems*, volume 2. Siam, 2001.
- [79] M. Szell, R. Lambiotte, and S. Thurner. Multirelational organization of large-scale social networks in an online world. *Proceedings of the National Academy of Sciences*, 107(31):13636–13641, 2010. doi: 10.1073/pnas.1004008107.
- [80] V. A. Traag, P. Van Dooren, and P. De Leenheer. Dynamical models explaining social balance and evolution of cooperation. *PloS one*, 8(4):e60063, 2013.
- [81] R. van Handel. Structured random matrices. In *Convexity and Concentration*, pages 107–156. Springer, 2017.
- [82] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

- [83] G. F. Young, L. Scardovi, and N. E. Leonard. Robustness of noisy consensus dynamics with directed communication. In *American Control Conference (ACC), 2010*, pages 6312–6317. IEEE, 2010.
- [84] W. Yu, G. Chen, and M. Cao. Consensus in directed networks of agents with nonlinear dynamics. *IEEE Transactions on Automatic Control*, 56(6):1436–1441, 2011.
- [85] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.
- [86] X. Zhang, R. R. Nadakuditi, and M. E. Newman. Spectra of random graphs with community structure and arbitrary degrees. *Physical Review E*, 89(4):042816, 2014.
- [87] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec. Seismic: A self-exciting point process model for predicting tweet popularity. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1513–1522. ACM, 2015.