Ph.D. Thesis

Structure and controllability of complex networks

Márton Pósfai

Supervisor: Prof. Gábor Vattay, D.Sc.

Department of Physics of Complex Systems
Eötvös Loránd University, Budapest, Hungary

Graduate School of Physics
Head of School: Prof. László Palla, D.Sc.

Doctoral Program for Statistical Physics,
Biological Physics and Physics of Quantum Systems
Head of Program: Prof. Jenő Kürti, D.Sc.

Budapest
2014

Acknowledgments

First I would like to thank my supervisor Prof. Gábor Vattay for his many years of indispensable guidance. I owe special thanks to Prof. Albert-László Barabási for his support during and after my two year visit at Northeastern University, Boston. I am utmost grateful to Prof. Yang-Yu Liu who introduced me to structural controllability and whose advise was essential to complete my thesis. I am indebted to Dr. Philipp Hövel for his kindness and immeasurable support which made my half-year stay at TU Berlin an unforgettable and fruitful experience. I also thank the rest of my collaborators: Dr. Tao Jia, Prof. Jean-Jacques Slotine and Dr. Endre Csóka.

I am obliged to my family and friends for providing things to do that are not work related. And last but definitely not least, I would like to thank my wife Zsófia Pénzváltó who took time off from writing her own thesis to marry me.

Contents

Chapter 1

Introduction

In our everyday life we are surrounded by systems that are profoundly complicated, e.g. society, life or technology. Complexity science aims to mathematically characterize and predict such systems by investigating how interaction of the parts of these systems give rise to emerging phenomena [1, 2, 3, 4, 5, 6]. An incomplete list of topics of complex systems study include nonlinear dynamics, game theory, self-organization and evolution theory.

Starting from around the turn of the century, large scale datasets describing complex systems became increasingly available for researchers. Using specialized software, maps of the World Wide Web were created [7]. Social networking websites and telecommunication companies collected data of millions of interacting users [8, 9]. High throughput biological experiments produced detailed maps of interacting constituents of living cells [10]. Today, the list of available datasets is expanding at an unprecedented rate. One of the most striking discoveries that emerged from analysis of this data is that – although these systems might be very different in nature – the networks formed by the interactions within the different systems posses common traits. For example, these networks are typically scale-free, meaning that a small fraction of elements – called hubs – have orders of magnitude more connections than a randomly selected element [11]. These networks are also small-world: similarly to regular lattices the neighbors of an element are densely connected, but unlike lattices the typical distance between two elements scales with the logarithm of the network size or even slower [12, 13]. These observations prompt the questions: What is the reason behind the existence of common features? Is there a common origin? How do these properties affect the behavior of the system? In pursue of answering these questions a new field of complexity science emerged, now commonly referred to as network science [14, 15, 16, 17, 18].

Consequently, in the last two decades, significant amount of research has been devoted to explore these questions, however, many questions remain unanswered and many new

questions arise. For example, only limited progress has been made in describing how the network structure of a system influences our ability to control it. My thesis aims to shed light on some aspects of this relation based on five publications that I have co-authored during my PhD studies [19, 20, 21, 22, 23].

In Chapter 1, I introduce the model of structured linear systems and controllability, which serve as fundamental tools to study control of complex networks in my thesis. I also provide a short review of related results and alternative approaches to network control. In Chapter 2, I introduce core percolation, a structural phase transition in complex networks closely related to linear controllability of networks. The analytical solution derived in this chapter serves as a tool for later investigations. In Chapter 3, two distinct problems are explored. First, I investigate how typical structural correlations present in complex networks affect controllability. Then, I categorize the constituents of the network based on their role in control, and I examine the relation of the categories to network structure. And in Chapter 4, I extend the framework of structural controllability to systems where the timescale of the dynamics of the system are comparable to the timescale of the changes in the network structure.

1.1 Linear dynamics and controllability

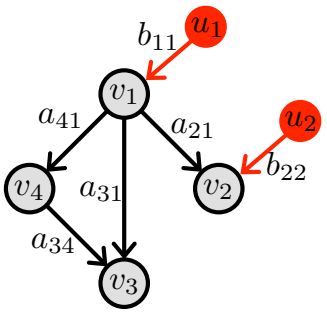
Through out the thesis linear dynamics on directed networks are considered [24, 25]. A directed network $\mathcal{G}(V, E)$ is a set of nodes $V = \{v_1, v_2, \ldots, v_N\}$ and a set of directed links $E = \{e_1, e_2, \ldots, e_L\}$ connecting these nodes. A link $e = (v_i, v_j, w_{ij})$ points from source node v_i to target node v_j, representing that v_i influences v_j in some manner. The strength of the interaction is provided by the weight w_{ij}. The internal dynamics of the system are formally given as

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \tag{1.1}$$

where the vector $\mathbf{x}(t) \in \mathbb{R}^N$ represents the state variables, $x_i(t)$ corresponding to the state of node v_i at time t. For example, $x_i(t)$ may provide the amount of traffic passing through a node in a transportation network, or it may describe the concentration of a molecule in a reaction-diffusion system. The $\mathbf{A} \in \mathbb{R}^{N\times N}$ matrix is the transposed weighted adjacency matrix: $a_{ji} = w_{ij} \neq 0$ if there is a link pointing from source node v_i to target node v_j, and $a_{ji} = 0$ otherwise. All information about the network is contained in matrix $\mathbf{A}$, therefore the network is sometimes referred to as $\mathcal{G}(\mathbf{A})$.

To impose control on the system, a new term is added to Eq. (1.1):

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \tag{1.2}$$



$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ a_{21} & 0 & 0 & 0 \\ a_{31} & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{B} = \begin{pmatrix} b_{11} & 0 \\ 0 & b_{22} \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{C} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & b_{11} & 0 \\ 0 & 0 & 0 & 0 & a_{21}b_{11} & 0 & 0 & b_{22} \\ 0 & 0 & a_{34}a_{41}b_{11} & 0 & a_{31}b_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{41}b_{11} & 0 & 0 & 0 \end{pmatrix}$$

$$N = 4,\ M = 2,\ \text{rank}\mathbf{C} = 4$$

Figure 1.1: **Controlling a simple network.** **(a)** A directed network representing linear dynamics with $N = 4$ internal nodes and $M = 2$ control signals. The control signals are directly linked to input nodes v_1 and v_2. **(b)** Matrix $\mathbf{A}$ describes the internal dynamics, and matrix $\mathbf{B}$ identifies the input nodes. By calculating the rank of the controllability matrix $\mathbf{C}$, the system is found to be controllable. Figure adopted from Ref. [26] with the permission of the authors.

where the vector $\mathbf{u}(t) \in \mathbb{R}^M$ is the control signal imposed on the system by the controller, and the matrix $\mathbf{B}(t) \in \mathbb{R}^{N \times M}$ identifies which nodes are directly connected to the control signal, in general one control signal can be connected to multiple nodes in the network. Note that although Eq. (1.2) describes continuous time dynamics, all results presented are valid for discrete time dynamics as well:

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t). \tag{1.3}$$

The dynamics described in Eq. (1.2) and (1.3) can be represented as a directed network $\mathcal{G}(\mathbf{A}, \mathbf{B})$ with $V = V_A \cup V_B$ and $E = E_A \cup E_B$. Where $V_A = \{v_1, \ldots, v_N\}$ is the set of internal nodes, $V_B = \{u_1, \ldots, u_M\}$ is the set of nodes representing control signals, and $E_B = \{(u_i, v_j, b_{ji}) | b_{ji} \neq 0\}$ is the set of links connecting the control signals to the network. The internal nodes that are directly connected to control signals are called *input nodes*[1] or actuators in control theory terminology. On the example shown on Fig. 1.1a, nodes v_1 and v_2 are input nodes.

In control theory, a system is called *controllable*, if by appropriate choice of inputs, it can be driven from any initial state to any final state in finite time. Following the classical result of Kalman, a possible way to determine if a linear system is controllable

[1] In Ref. [26] the input nodes are called controlled nodes, here I follow the convention of Ref. [22].

is to construct the $N \times NM$ controllability matrix

$$\mathbf{C} = \left[\mathbf{A}^{N-1}\mathbf{B}; \ldots; \mathbf{A}^{2}\mathbf{B}; \mathbf{AB}, \mathbf{B}\right], \tag{1.4}$$

where $[\cdot, \cdot]$ is the concatenation of matrices. The system is controllable if $\mathbf{C}$ has full rank, that is

$$\operatorname{rank} \mathbf{C} = N, \tag{1.5}$$

this condition is generally known as Kalman's rank condition [27]. The example shown on Fig. 1.1 has four internal nodes and $\operatorname{rank} \mathbf{C} = 4$, therefore the network is controllable.

Kalman's rank condition is particularly simply understood in the case of discrete time dynamics. Without the loss of generality, consider a system at the origin at time $t = 0$, that is $\mathbf{x}(0) = \mathbf{0}$. At $t = 1$, the state of the system is determined by the control signal: $\mathbf{x}(1) = \mathbf{Bu}(0)$. At $t = 2$, the state of the system depends on its state at $t = 1$ and the control signal: $\mathbf{x}(2) = \mathbf{ABu}(0) + \mathbf{Bu}(1)$. Repeating this N times, the state of the system is

$$\mathbf{x}(N) = \mathbf{A}^{N-1}\mathbf{Bu}(0) + \ldots + \mathbf{ABu}(N-2) + \mathbf{Bu}(N-1), \tag{1.6}$$

which can be written in the form

$$\mathbf{x}(N) = \mathbf{Cu}, \tag{1.7}$$

where $\mathbf{u} = [\mathbf{u}(0)^T, \mathbf{u}(1)^T, \ldots, \mathbf{u}(N-1)^T]^T$. According to the definition, a system is controllable, if $\mathbf{x}(N)$ can be set to any desired value by appropriately choosing $\mathbf{u}$, that is Eq. (1.7) has a solution, which in turn is equivalent with Kalman's rank condition (1.5).

1.2 Structured systems

Kalman's rank condition is useful to determine the controllability of a system if $\mathbf{A}$ and $\mathbf{B}$ are given and the system is sufficiently small. However, complex systems are generally large and the exact values of the link weights are often unknown. Furthermore, $\mathbf{B}$ is not given: our goal is to design a $\mathbf{B}$ that renders the system controllable and satisfies certain requirements. For example, if we directly control all nodes ($M = N$ and $\mathbf{B} = \mathbf{1}$) the system is trivially controllable, however this is an unrealistic scenario and we do not make use of the internal dynamics. A more meaningful question is to ask is what is the minimum number of control signals needed to control the system? Originally introduced by Lin, structural controllability provides an avenue to answer this question despite incomplete link weight information [28].

For a linear system defined by the pair $(\mathbf{A}, \mathbf{B})$, incomplete knowledge of link weights means that we do not know the precise value, just the location of the nonzero entries. That

is we know where the links are in $\mathcal{G}(\mathbf{A}, \mathbf{B})$, but we do not know the weights themselves. Such systems can be modeled by so-called *structured systems*: the zero entries in $(\mathbf{A}, \mathbf{B})$ are kept fixed and the nonzero values are treated as free parameters. The list of free parameters is denoted by the $\mathbf{w} \in \mathbb{R}^L$ vector.

Two linear systems $(\mathbf{A}, \mathbf{B})$ and $(\mathbf{A}^*, \mathbf{B}^*)$ are said to have the *same structure* if the zero entries in $(\mathbf{A}, \mathbf{B})$ and $(\mathbf{A}^*, \mathbf{B}^*)$ are in the same positions and only the value of the nonzero elements can be different, i.e. the links in the corresponding networks $\mathcal{G}(\mathbf{A}, \mathbf{B})$ and $\mathcal{G}(\mathbf{A}^*, \mathbf{B}^*)$ connect the same nodes, only the weights can be different.

Why is the structured system approach useful? It is clear that the properties of a system depend on the specific choice of $\mathbf{w}$. However, for a number of interesting properties – including controllability – it is true that if we find a choice of parameter values such that the property is true, it is true for almost all parameter values. That is the property is true for all parameter values except for those in an algebraic variety in the parameter space $\mathbb{R}^L$, and this algebraic variety has Lebesgue measure zero. Such property is said to be a *generic* property or *generically* true [29, 30].

To illustrate this consider a structured matrix $\mathbf{M}$. If $\mathbf{M}$ has full rank, $\det \mathbf{M} \neq 0$ holds. The determinant can be zero for two reasons: (i) the position of fixed zero entries, e.g. a row in matrix $\mathbf{M}$ is all fixed zeros; or (ii) the specific choice of free parameters $\mathbf{w}$ satisfies the equation $\det \mathbf{M}(\mathbf{w}) = 0$.

In case (i), $\mathbf{M}$ does not have full rank, independent of the choice of parameters. For example consider

$$\mathbf{M} = \begin{pmatrix} w_1 & w_2 & 0 \\ w_3 & w_4 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{1.8}$$

where $\mathbf{w} = (w_1, w_2, w_3, w_4)^T \in \mathbb{R}^4$ is the set of free parameters, the rest of the entries in $\mathbf{M}$ are fixed zeros. For any choice of $\mathbf{w}$, $\det \mathbf{M} = 0$.

In case (ii), $\mathbf{M}$ always has full rank except for an algebraic variety in the parameter space defined by the polynomial equation $\det \mathbf{M}(\mathbf{w}) = 0$. Therefore, if there exists a choice of $\mathbf{w}$ such that $\mathbf{M}$ has full rank, it has full rank for almost all values of $\mathbf{w}$. For example consider

$$\mathbf{M} = \begin{pmatrix} w_1 & w_2 & 0 \\ w_3 & w_4 & 0 \\ 0 & 0 & w_5 \end{pmatrix}, \tag{1.9}$$

where $\mathbf{w} \in \mathbb{R}^5$ is the set of free parameters, the rest of the entries in $\mathbf{M}$ are fixed zeros. The determinant is only zero if the equation $w_1 w_4 w_5 - w_2 w_3 w_5 = 0$ is satisfied which defines an algebraic variety in $\mathbb{R}^5$. That is $\det \mathbf{M} \neq 0$ for almost all choices of $\mathbf{w}$. This illustrates that structured systems are relevant models of real linear systems if the weights in the

original system do not have any particular symmetry. In the next section the approach is applied to study controllability.

1.3 Sturctural controllability

A system $(\mathbf{A}, \mathbf{B})$ is *structurally controllable* if there exists a system $(\mathbf{A}^*, \mathbf{B}^*)$ with the same structure, such that $(\mathbf{A}^*, \mathbf{B}^*)$ is controllable in the original sense.

To understand the difference between the two concepts of controllability consider the following examples adopted from Ref. [26].

Example 1 The dynamics governing the network show on Fig. 1.2a can be written as

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ a_{21} & 0 & 0 \\ a_{31} & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} + \begin{pmatrix} b_{11} \\ 0 \\ 0 \end{pmatrix} \cdot u_1(t). \tag{1.10}$$

The corresponding controllability matrix is

$$\mathbf{C} = \left[\mathbf{A}^2\mathbf{B}, \mathbf{AB}, \mathbf{B}\right] = b_{11} \begin{pmatrix} 0 & 0 & 1 \\ 0 & a_{21} & 0 \\ 0 & a_{31} & 0 \end{pmatrix}. \tag{1.11}$$

Since $\operatorname{rank} \mathbf{C} = 2$ independent of the values of a_{21}, a_{31} and b_{11}, the network is both uncontrollable and structurally uncontrollable. Indeed node v_1 broadcasts the same message to its neighbors v_2 and v_3, therefore $x_2(t)$ and $x_3(t)$ remain correlated.

Example 2 The dynamics governing the network show on Fig. 1.2b can be written as

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ a_{21} & 0 & 0 \\ a_{31} & 0 & a_{33} \end{pmatrix} \cdot \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} + \begin{pmatrix} b_{11} \\ 0 \\ 0 \end{pmatrix} \cdot u_1(t). \tag{1.12}$$

The corresponding controllability matrix is

$$\mathbf{C} = \left[\mathbf{A}^2\mathbf{B}, \mathbf{AB}, \mathbf{B}\right] = b_{11} \begin{pmatrix} 0 & 0 & 1 \\ 0 & a_{21} & 0 \\ a_{33}a_{31} & a_{31} & 0 \end{pmatrix}. \tag{1.13}$$

Now $\operatorname{rank} \mathbf{C} = 3$, independent of a_{21}, a_{31}, a_{33} and b_{11}, therefore the system is both controllable and structurally controllable. The self-loop added to node v_3 breaks the

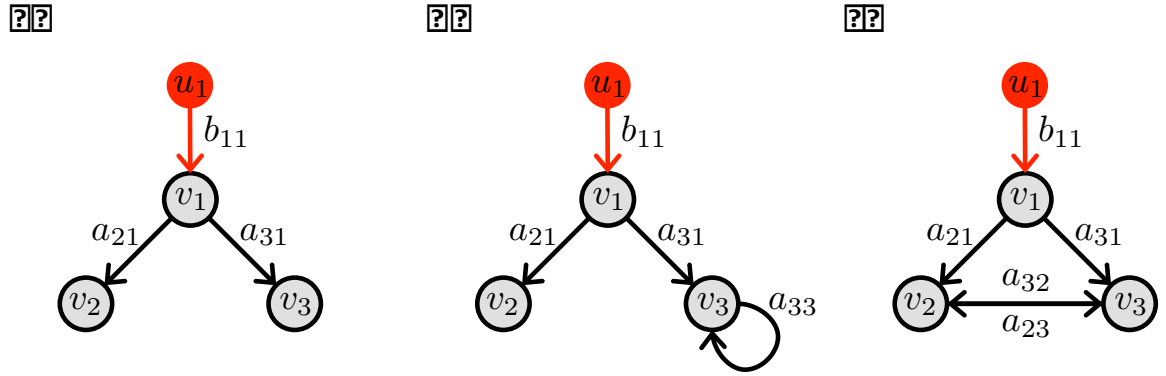


Figure 1.2: **Comparing controllability and structural controllability.** **(a)** A small star network with one central nodes and two neighbors. The state of nodes v_2 and v_3 remain coupled independently of the detailed values of the weights, therefore the network is both uncontrollable and structurally uncontrollable. **(b)** By adding a self-loop to node v_3, the symmetry is destroyed, and the system becomes both controllable and structurally controllable. **(c)** This example is controllable for almost all values of $\mathbf{A}$, except if equation $a_{32}a_{21}^2 - a_{23}a_{31}^2 = 0$ is satisfied. Therefore the network is structurally controllable, but it is not controllable for all values of the nonzero entries of $\mathbf{A}$. The examples were adopted from Ref. [26] with permission of the authors.

symmetry between v_2 and v_3, rendering the network controllable.

Example 3 The dynamics governing the network show on Fig. 1.2c can be written as

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ a_{21} & 0 & a_{23} \\ a_{31} & a_{32} & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} + \begin{pmatrix} b_{11} \\ 0 \\ 0 \end{pmatrix} \cdot u_1(t). \tag{1.14}$$

The corresponding controllability matrix is

$$\mathbf{C} = \left[\mathbf{A}^2\mathbf{B}, \mathbf{AB}, \mathbf{B}\right] = b_{11} \begin{pmatrix} 0 & 0 & 1 \\ a_{23}a_{31} & a_{21} & 0 \\ a_{32}a_{21} & a_{31} & 0 \end{pmatrix}. \tag{1.15}$$

In this example the rank of $\mathbf{C}$ depends on the value of the entries in $\mathbf{A}$. For majority of cases $\operatorname{rank}\mathbf{C} = 3$ and the network is controllable. Only if $a_{32}a_{21}^2 - a_{23}a_{31}^2 = 0$ is satisfied, will the system become uncontrollable ($\operatorname{rank}\mathbf{C} = 3$). Therefore the system is structurally controllable, but not always controllable in the original sense. Note that if the network is uncontrollable, an arbitrarily small change in the any link weight makes the system controllable.

As illustrated by the third example, structural controllability is a generic property,

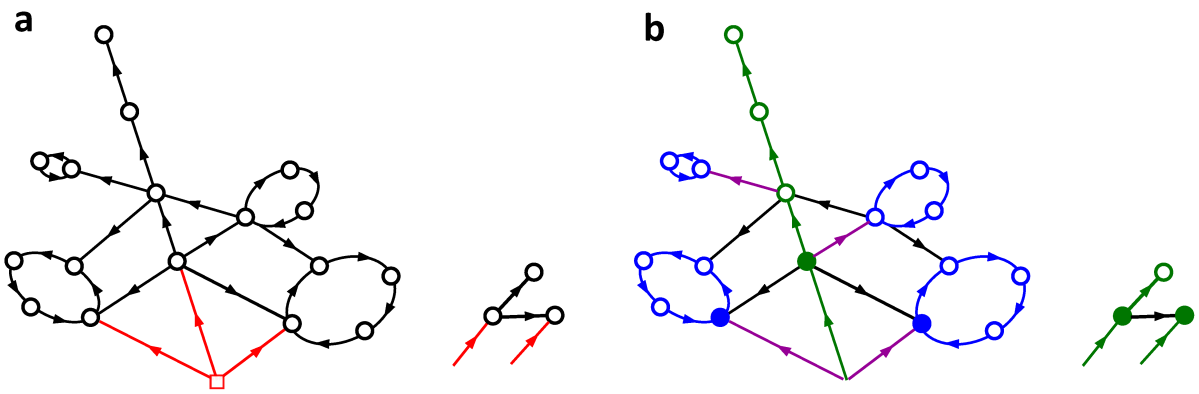


Figure 1.3: **Lin's controllability theroem. (a)** A directed network $\mathcal{G}(\mathbf{A}, \mathbf{B})$ with $N = 19$ internal nodes (black circles) controlled by $M = 3$ control signals (red squares). **(b)** The network is spanned by three disjoint cacti, therefore it is structurally controllable. The input nodes (full circles) are the nodes directly connected to the control signals. The stems (green) are paths originating from control signals, the first internal node of the stem is the driver node (full green circles). The buds are cycles (blue) that are connected by the distinguished link (purple) to the rest of the cacti. Figure was adopted from Ref. [26] with permission from authors.

therefore it is meaningful to use it as a tool to study directed weighted networks where no underlying symmetry in the weights is expected, e.g. the weights of the links are real numbers drawn from some distribution. However, if the network is unweighted (all weights are equal), or the network is undirected (if there is a link $(v_i \to v_j)$, then there is also a link $(v_j \to v_i)$ with the same weight) alternative approaches have to be considered.

Given that the above requirements are met, structural controllability of a system $(\mathbf{A}, \mathbf{B})$ can be determined using efficient graph combinatorial methods. To introduce a test for controllability few further definitions are needed.

A *stem* is an elementary path originating from a node representing a control signal in $\mathcal{G}(\mathbf{A}, \mathbf{B})$ (highlighted with green on Fig. 1.3).

A *bud* is an elementary cycle (blue on Fig. 1.3) and an additional link (purple on Fig. 1.3) that ends, but not starts at a node in the cycle. The additional link is referred to as the *distinguished link* .

A *cactus* C is a subnetwork which is defined recursively: C is a stem or it is obtained from a smaller cactus C^*, to which a bud is added such that the distinguished link of the bud originates from any node of C^* except the final node of the stem.

Using these definitions, Lin's controllability theroem states that a dynamical system $(\mathbf{A}, \mathbf{B})$ is structurally controllable iff there exists a node disjoint union of cacti that covers all nodes in $\mathcal{G}(\mathbf{A}, \mathbf{B})$ [28, 31]. For example the network on Fig. 1.3 is spanned by three disjoint cacti, therefore it is structurally controllable.

Each stem starts at a control signal, and the second node in the stem is an internal node which is called a *driver node* (full green circles on Fig. 1.3) [26]. Note that driver nodes are not equivalent to the previously defined input nodes (or actuators). In a cacti, one control signal can be connected to multiple input nodes: one driver node and any number of nodes in buds. A driver node is somewhat distinguished as it requires an independent control signal, while multiple buds can be controlled by the same signal. The number of driver nodes is equal to the number of independent control signals.

Following the seminal work of Lin, the control theory community studied many aspects of structured systems. For a review see Ref. [30].

1.4 Minimum input theorem

Liu et al. were the first to study the control of complex networks using structural controllability [26]. They introduced the minimum input theorem which made it possible to efficiently identify the minimum number of independent control signals N_{CS}, or equivalently the minimum number of driver nodes N_{D} needed to impose full control on the system. In this section, I briefly introduce the theorem.

To be able to state the minimum input theorem, the concept of matching has to be extended to directed networks. A *matching* in an undirected network is a set of independent links, meaning that no pair of links share endpoints [32]. In a directed network a *matching* is a set of links such that no pair of links share common start- or endpoints [26].

In both cases a *maximum matching* is matching with maximum cardinality. In general for a given network there exists many different maximum matchings. A maximum matching can be efficiently found using the Ford-Fulkerson algorithm for undirected and the Hopcroft-Karp algorithm for directed networks [33, 34, 35]. In undirected networks a node is said to be *matched*, if it is adjacent to a link in the matching in undirected networks or there exists a link in the matching that points at it in directed networks. If all nodes are matched in the network, the matching is called a *perfect matching*.

The minimum input theorem by Liu et al. is stated as follows [26]: The minimum number of independent control signals (N_{CS}) or equivalently the minimum number of driver nodes (N_{D}) needed to fully control a network $\mathcal{G}(\mathbf{A})$ is one if there is a perfect matching in $\mathcal{G}(\mathbf{A})$, and any node can be chosen to be the driver. Otherwise it is equal to the number of unmatched nodes with respective to any maximum matching. In this case, the driver nodes are the unmatched nodes.

For rigorous proof I encourage the reader to consult the Supplementary Information of Ref. [26]. Here I provide a simple heuristic for discrete dynamics to understand the theorem. Consider a system with discrete dynamics $\mathcal{G}(\mathbf{A})$ (Fig. 1.4a). We make two copies

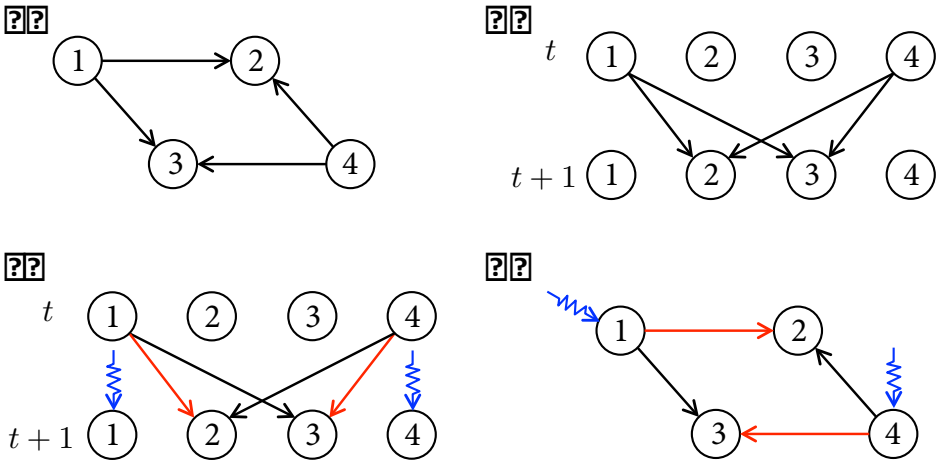


Figure 1.4: **Minimum input theorem. (a)** A small network $\mathcal{G}(\mathbf{A})$ with $N = 4$ nodes. The goal is to design a $\mathbf{B}$, such that $\mathcal{G}(\mathbf{A}, \mathbf{B})$ is structurally controllable and N_{D} is minimal. **(b)** Two copies are made of each node v_i, one representing the state of v_i at time t and one representing the state at time t. Links connect nodes in layer t with nodes in layer $t + 1$, representing that the state of node v_i at $t + 1$ depends on the state of its in-neighbors at time t. Following this procedure a bipartite network is obtained. **(c)** Let us assume that the network is controllable. This means that it is possible to set the state of each node independently at time t. We can use the state of the nodes at time t to set the state of nodes at time $t + 1$ by designating a node at time t to control a node connected to it at time $t + 1$.Therefore to make the most use of the internal dynamics, the nodes at time t are paired up with nodes at $t + 1$, that is we find the maximum matching in the bipartite network. The matched nodes in layer $t+1$ are controlled by the internal dynamics, the rest has to be controlled using independent control signals. **(d)** In the original network, the driver nodes are the nodes that do not have a link pointing at them in the maximum matching, in line with the minimum input theorem.

of the set of nodes V: one representing the system at time t and one at time $t + 1$. The state of node v_i at $t + 1$ is determined by the state of the nodes pointing at it at time t, to represent this we connect the node v_j at time t to node v_i at time $t + 1$ if they are connected in $\mathcal{G}(\mathbf{A})$ (Fig. 1.4b). Assume that the network is controllable, by definition we can set the state of all nodes independently at time t. The system is also controllable at time $t+1$, we can use the state of the nodes at time t to set the state of nodes at time $t+1$: we can designate a node at time t to control a node connected to it at time $t+1$. Therefore to make the most use of the internal dynamics, we pair up the nodes at time t with nodes at $t + 1$ (Fig. 1.4c). This is equivalent with finding a maximum matching in $\mathcal{G}(\mathbf{A})$. The nodes that cannot be controlled using the internal dynamics are the unmatched nodes. Therefore, these unmatched nodes will be the driver nodes (Fig. 1.4d).

1.5 The role of the degree distribution

In order to systematically study the underlying networks of complex systems, we have to mathematically describe their structure. The most important property of a node is its degree k, representing the number of connections the node has in the network. The degree sequence $(k_1, k_2, \ldots, k_N)$ provides the degree of each node, and the degree distribution $p(k)$ represents the probability that a randomly selected node has degree k. The average $\langle k \rangle = \sum_k k p_k$ and the second moment $\langle k^2 \rangle = \sum_k k^2 p_k$ of the degree distribution provide simple means to compare the degree distribution of different networks.

Consider a large network with given average degree $\langle k \rangle$. If the network would have no additional structure, i.e. the nodes are connected randomly, the degree distribution would be expected to be Poissonian $p(k) = e^{-\langle k \rangle} \langle k \rangle^k / k!$ [36, 37]. For Poisson distribution $\langle k^2 \rangle = \langle k \rangle + \langle k \rangle^2$, meaning that the fluctuations in node degree are small compared to the system size. Therefore $\langle k \rangle$ can be thought of as a typical value or scale of node degree. Such networks attracted much interest following the pioneering work of Erdős and Rényi [37]. However, real networks differ from this simple assumption in a way that significantly affects the behavior of the system.

It was found that in almost all real networks $\langle k^2 \rangle$ is significantly higher than expected by chance [14], that is

$$\langle k^2 \rangle \gg \langle k \rangle + \langle k \rangle^2. \tag{1.16}$$

In many cases the empirical degree sequence measured in real systems can be approximated as a sample drawn from heavy tailed distributions [38], that is for large k

$$p_k \sim k^{-\gamma}, \tag{1.17}$$

where γ is called the degree exponent. For $\gamma < 3$ the second moment $\langle k^2 \rangle$ is infinite, meaning that the node degree does not have have a typical value. For this reason such networks are called *scale-free* [11]. The outcome of dynamical processes is known to be fundamentally different on scale-free and Poisson distributed networks, e.g. percolation or spreading processes [39, 40].

In the context of structural controllability only directed networks are investigated, that is each node has an in- and an out-degree, representing the number of links pointing at and originating from the node, respectively. Hence directed networks have an in-degree distribution $p_{\text{in}}(k)$ and an out-degree distribution $p_{\text{out}}(k)$.

Liu et al. applied the minimum theorem to investigate how network structure affects N_{D}, finding that the degree distribution of the network is the main determining factor [26]. They gathered a large collection of real network topologies encompassing a wide

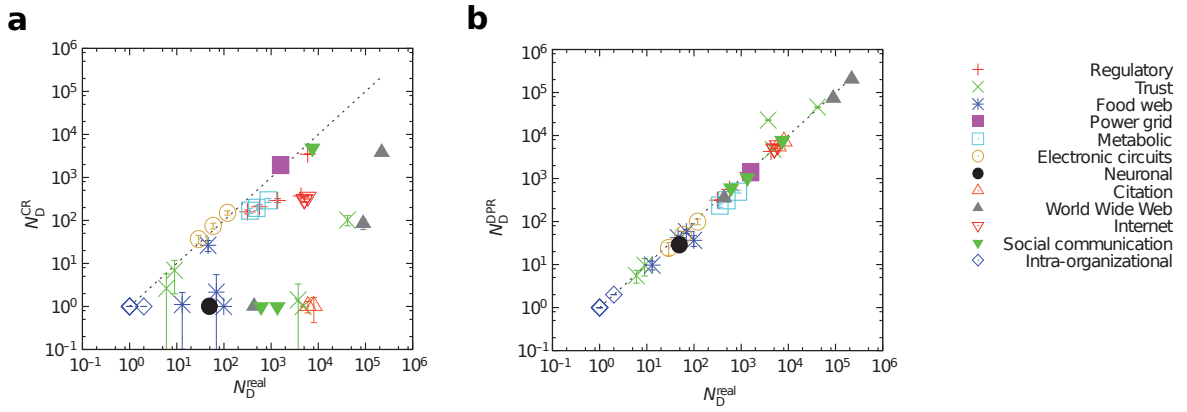


Figure 1.5: **Randomizing real networks. (a)** Comparison of the number of driver nodes in the original networks (N_{D}) and the number of driver nodes after complete randomization ($N_{\mathrm{D}}^{\mathrm{CR}}$) on log-log scale. Each symbol corresponds to a real network, the type of the network is indicated by the legend. Complete randomization eliminates all structure in the network and only the average degree $\langle k \rangle$ is preserved. The majority of symbols are not on the diagonal, showing that $\langle k \rangle$ is not enough information to correctly predict even the order of N_{D}. **(b)** Comparison of the number of driver nodes in the original networks (N_{D}) and the number of driver nodes after degree preserving randomization ($N_{\mathrm{D}}^{\mathrm{DPR}}$) on log-log scale. All symbols but one line up on the diagonal, indicating that the degree sequence of the network is the main factor determining N_{D}. Figure was adopted from Ref. [26] with permission from the authors.

range of systems, and calculated N_{D}. Then the authors randomized the networks using two different methods: (i) Complete randomization, shuffling the end points of the links keeping only the number of nodes and number of links the same. The resulting network has Poisson degree distribution with the same average degree, but all other structure – including the scale-free property – is eliminated. (ii) Degree preserving randomization [41], two links $(v_i \to v_j)$ and $(v_k \to v_l)$ are randomly chosen, the two links are broken and rewired creating new links $(v_i \to v_l)$ and $(v_k \to v_j)$, this step is repeated until a steady state is reached. The resulting network has the same degree sequence as the original one, but all other structure is eliminated. After randomization authors recalculated the number of driver nodes $N_{\mathrm{D}}^{\mathrm{CR}}$ and $N_{\mathrm{D}}^{\mathrm{DPR}}$ for complete and degree preserving randomization, respectively. Comparing $N_{\mathrm{D}}^{\mathrm{CR}}$ to N_{D} they found very different values, indicating that the average degree by itself is not enough information to accurately predict N_{D} (Fig. 1.5a). On the other hand, $N_{\mathrm{D}}^{\mathrm{DPR}}$ and N_{D} are approximately the same, providing evidence that the degree distribution is the main factor determining N_{D} (Fig. 1.5b).

Knowing the importance of the degree distribution prompts the question: how does the shape of the degree distribution influence N_{D}? Liu et al. used model networks to systematically study the effect of various degree distributions, finding that for networks

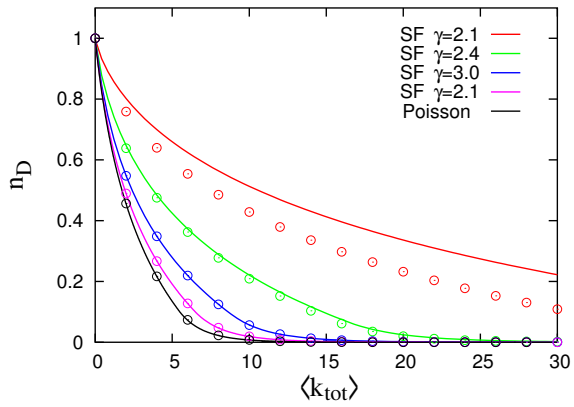


Figure 1.6: **Fraction of driver nodes in model networks.** The fraction of driver nodes is shown $n_{\mathrm{D}} = N_{\mathrm{D}}/N$ in function of the average total degree $\langle k_{\mathrm{tot}} \rangle = \langle k_{\mathrm{in}} \rangle + \langle k_{\mathrm{out}} \rangle$. The symbols correspond to measurements in networks generated with scale-free and Poisson degree distribution with size $N = 10^5$ [37, 43], the continuous lines correspond to analytical solution derived in the $N \to \infty$ limit [26]. The finite size effect is increasingly important as $\gamma \to 2$. The calculations indicate that scale-free networks require than Poisson distributed networks, and dense networks are easier to control. Figure was adopted from Ref. [26] with permission from the authors.

with the same average degree $\langle k \rangle$, scale-free networks are harder to control than Poisson distributed networks (Fig. 1.6). Recently, Menichelli et al. showed that the important quantity of the degree distribution is the density of nodes with degree one $p_{\mathrm{in/out}}(k = 1)$ and degree two $p_{\mathrm{in/out}}(k = 2)$ [42].

Liu et al. also investigated the properties of driver nodes, finding that driver nodes are typically low degree nodes and tend to avoid hubs, a result that might seem surprising at first. However, we have to keep in mind that in our linear model a node broadcasts the same message to all of its neighbors, therefore even though a hub has many connections, we can only use it to control one adjacent node.

1.6 Further advances in structural controllability of complex networks

In the wake of Ref. [26] a number of papers appeared that rely on structural controllability to investigate complex networks. In this section, I provide a brief review of recent results.

Liu et al. introduced a new measure called control centrality with the goal of quantifying the ability of an individual node to control a network [44]. The authors developed an algorithm to identify the subnetwork that can be controlled using a single node as an input node. They investigated the relation between hierarchy and control centrality,

finding that in directed acyclic networks the control centrality is uniquely determined by the topological position of the node in the underlying hierarchical structure.

Ruths and Ruths investigated what structures in the network induce the need to add a driver node [45]. Based on their findings, they introduce a control profile to characterize the control structure in more detail, uncovering that the control structure of random network models and real networks are very different. They also found that the control profile of real networks can be divided in three distinct classes, providing some insight into high-level organization of the complex systems.

It was suggested by Cowan et al. that many real-world systems represented by networks have intrinsic nodal dynamics [46], i.e. a node's state will vary over time even in the absence of influence from other nodes. Thus if we want to model the dynamics of these systems in a network context, we require self-links, even if they were not made explicit in the original network description. This is a modeling or time-scale issue, for example in electronics the distinction between integrator, leaky integrator, and low-pass filter is based on similar numerical considerations. However, if we add a self-link to each node, then the minimum input theorem predicts that $N_{\mathrm{D}} = 1$ which is correct in principle, but we learn little from it. Cowan et al. instead advocate that the minimum number of input nodes is more meaningful. Systematic study of this set of questions is still lacking.

Pequito et al. recently devised an efficient algorithm to identify the minimum number of input nodes in a general network [47]. This might provide means to investigate the effect of self-loops on the number of input nodes instead of driver nodes.

Nepusz and Vicsek introduced and investigated the control of a dynamical system that takes place on the links of the network [48]. The authors assigned a state variable to each link, each node takes the state of the links pointing at it as input, and the output is the state of links originating from the node. The input and the output are linked together through a mixing matrix $\mathbf{M}$. Control is imposed by setting the values of the $\mathbf{M}$ matrix for a subset of nodes. The authors demonstrated that the control of such systems is substantially different from the control of systems where the state variables are associated to the nodes.

Liu et al. also investigated observability [49], a problem closely related to controllability. A system is called observable if we can reconstruct the system's complete internal state from its outputs. The authors provided a method to select nodes whose state necessarily has to be measured to assure observability. Applying their approach to biochemical reaction systems, they found that the identified sensors are not only necessary but also sufficient for observability.

1.7 Alternative approaches to network control

A number of network control frameworks exist complementary to structural controllability, here I provide a brief and inherently incomplete overview of recent efforts of the network science community.

Yuan et al. studied the linear controllabilty of networks where all link weights are known [50]. To emphasis the difference from structural controllability, the authors reefer to it as exact controllability. The importance of their result is that it is applicable to unweighted and undirected networks. Investigating a range of real and model networks, the authors found that dense networks with identical weights are difficult to be controlled.

Pinning control of complex networks is a fundamentally different approach. In case of controllability, we aim to be able to explore the entire phase space. While pinning control aims to stabilize synchronous states of the system by applying local feedback injections to a fraction of network nodes. The effect of network structure and different strategies has been intensely studied [51, 52].

Fiedler et al. investigated the network representation of systems of nonlinear differential equations [53, 54]. The authors showed that it is possible to determine the long-time dynamics of the system based on only measurements of the so-called feedback vertex set [55]. They also showed that applying open loop control at the feedback vertex set forces the remaining network to stably follow prescribed stable or unstable trajectories.

Cornelius et al. examined networks with given nonlinear dynamics [56]. The authors applied perturbation to the state of the nodes to push the system to the basin of attraction of a desired target attractor.

In many realistic scenarios, the goal of network control is not to drive the detailed dynamics of the system, but to reduce systematic risk of failure. For example, Noël et al. showed for the sandpile model on networks the risk of global cascades can be reduced by inducing finite size local cascades [57].

Chapter 2

Core percolation and maximum matching

Structural phase transitions in networks were extensively studied due to their impact on dynamical processes taking place on networks [58]. Such phase transitions include the emergence of a giant connected component [37, 59, 60] , k-core percolation [61, 62], k-clique percolation [63, 64] and explosive percolation [65, 66]. These transitions affect many properties of networks, e.g. robustness and resilience to breakdowns [39, 60], epidemic and information spreading on socio-technical systems [40, 67].

In this chapter, core percolation is studied which is a transition related to matching and network controllability. Previous theoretical work has focused on random networks with Poisson degree distribution. Here, analytical solution for both directed and undirected random networks with arbitrary degree distribution are derived, the equations obtained can be connected to the zero temperature limit of the cavity method solution of the size of maximum matching in random networks [68]. The analytical framework will serve as a tool to study controllability in later chapters. Core percolation has also been related to many other problems, including conductor-insulator transitions [69, 70] and the vertex cover problem [71].

The bulk of the chapter is based on Ref. [19] and Sec. 2.3 relies on Refs. [22, 23].

2.1 Network models

In this chapter, I provide analytical solution for core percolation on a class of network models. To understand the consequences and limitations of the results it is essential to understand the network models used. In this section, the definition and properties of these models are discussed.

Uncorrelated networks with tunable degree distribution are investigated, meaning that

only the degree distribution $p(k)$ is set and everything else is random. These network models are widely used in network science, and this popularity has two main reasons: (i) They are suitable to capture the degree distribution of real systems, the most important network characteristic that influence dynamical processes an them. (ii) The simplicity of the model allows analytic solution of properties of the network and dynamical processes on the network. There are several methods to generate such networks, here I list the ones used in the thesis.

2.1.1 Erdős-Rényi model

The Erdős-Rényi model is the most simple random network model, it was defined originally by its namesakes Erdős and Rényi (and independently by Gilbert) [36, 37]. To generate an Erdős-Rényi network we start with N unconnected nodes, then $L = cN/2$ links are randomly placed. In most cases multiple links between node pairs and self-loops are avoided. In the thesis, multiple links are not, but self-loops are allowed. In the sparse large network limit, that is $N \to \infty$ while the expected average degree $\langle k \rangle = 2L/N = c$ is kept constant, the degree distribution is Poissonian

$$p(k) = \frac{c^k}{k!} e^{-\langle c \rangle}. \tag{2.1}$$

The moment generating function defined as

$$G(x) = \sum_{k=0}^{\infty} p(k) x^k \tag{2.2}$$

is an important representation of degree distribution, particularly useful to study ensembles of random networks. Its role is similar to the role of partition functions in statistical physics. For Poisson distribution it is given by

$$G(x) = e^{-c(1-x)}. \tag{2.3}$$

The distribution $q(k)$ provides the probability that by following a randomly selected link we find a node with degree k, in general it can be calculated as

$$q(k) = \frac{k}{\langle k \rangle} p(k). \tag{2.4}$$

Particularly for Erdős-Rényi networks it is

$$q(k) = \frac{c^{k-1}}{(k-1)!} e^{-\langle c \rangle}. \tag{2.5}$$

The distribution $q(k+1)$ is called the excess degree distribution, because after reaching a node with degree $k+1$ we can proceed in k different directions other than the one we arrived from. In general, the generating function associated to the excess degree distribution can be calculated as

$$H(x) = \sum_{k=0}^{\infty} q(k+1)x^k = \frac{G'(x)}{\langle k \rangle}. \tag{2.6}$$

For Erdős-Rényi networks $q(k+1) \equiv p(k)$, therefore

$$H(x) = G(x) = e^{-c(1-x)}. \tag{2.7}$$

To illustrate the use of generating functions, I present a simple calculation. A *connected component* is a subnetwork in which any node pair is connected to each other through paths, and which is not connected to any other node in the network. It was shown by Erdős and Rényi that in the limit $N \to \infty$ for $c < 1$ the largest connected component is finite, at $c = 1$ a component containing infinite nodes emerges, and for $c > 1$ it spans a finite fraction of the network [37]. If the largest connected component occupies finite fraction of the network it is called the *giant component*.

Here, I calculate the relative size of the giant component S using generating function formalism [60, 72, 15]. First note that in the limit $N \to \infty$, Erdős-Rényi networks are locally treelike, meaning that the probability of finding finite length cycles diminishes. Next observation is that S is the probability that a randomly selected node belongs to the giant component, and, because in Poisson distributed networks $p(k) \equiv q(k+1)$, S is also equal to the probability that following a randomly selected link leads to a node belonging to the giant component.

Now consider a randomly selected link e, with probability $q(k+1)$ it leads to a node with k further links. The network is locally treelike, meaning each of the k links leads to an independent component, therefore they can be treated as independently selected with uniform probability. The probability that link e leads to a finite component is $1-S$, this has to be equal to $(1-S)^k$ which is the probability that none of the k excess links lead to the giant component. Averaging over k, the following self-consistent equation is obtained:

$$1 - S = \sum_{k=0}^{\infty} q(k+1)(1-S)^k = H(1-S) = e^{-cS}. \tag{2.8}$$

For $c < 1$, the only solution of the equation is $S = 0$; at the critical point $c^* = 1$, a new solution emerges, and for $c < 1$, $S > 0$ is finite (Fig. 2.1). The key to the solution was that the property that a node belongs to the giant component can be locally decided.

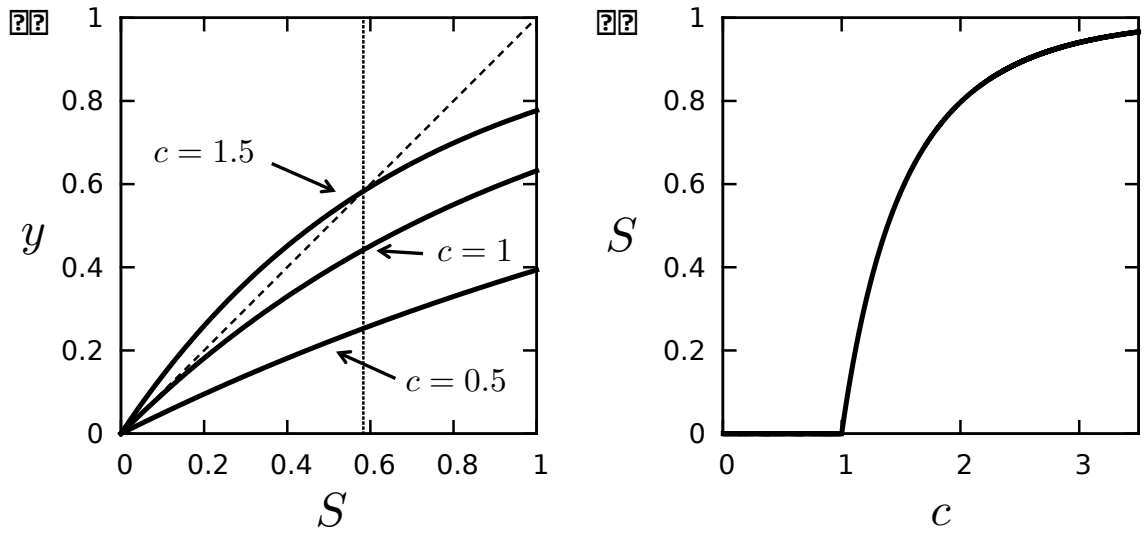


Figure 2.1: **Emergence of the giant component in Erdős-Rényi networks. (a)** Graphical solution of the self-consistent equations for various average degree c. The solution is given by the intersection of $y = 1 - e^{-cS}$ (continuous line) and $y = S$ (dashed line). For $c \leq 1$, the only solution is $S = 0$. At $c^* = 1$, the continuous line touches the dashed line, and for $c > 1$ the two lines intersect at finite S (dotted vertical line). **(b)** The relative size of the giant component in function of the average degree.

Analogous solutions can be derived of properties for which similar local rules can be devised.

The Erdős-Rényi model is simply extended to directed networks: starting with N unconnected nodes, $L = cN$ directed links are randomly placed. In the large network limit, both the in- and the out-degree distribution is Poissonian with average degree $\langle k \rangle \equiv \langle k_{\text{in/out}} \rangle = L/N = c$, therefore $G(x)$ and $H(x)$ are the same as the undirected case.

Most real networks are scale-free, therefore in network science the Erdős-Rényi model is mostly used as comparison to show the role of the scale-free property.

2.1.2 Configuration model

The configuration model provides means to actually set $p(k)$ [72, 73, 74]. We again start with N unconnected nodes. We assign k_i stubs or half-links to each node v_i, where k_i is drawn from the prescribed distribution $p(k)$, then the stubs are randomly paired and connected. Usually multiple links between node pairs are not allowed. If unconnected stubs remain at the end of the pairing process, they are discarded. For large sparse networks ($N \gg \langle k \rangle$), the number of discarded stubs remains small and does not influence the global properties of the network. Similarly to Erdős-Rényi networks, sparse and large

networks generated by the configuration model are locally treelike, enabling us to use the generating function formalism.

In the thesis, the configuration model is used to generate purely power-law distributed networks with

$$p(k) = \frac{k^{-\gamma}}{\zeta(\gamma)} \tag{2.9}$$

for $k \geq 1$, $\gamma > 2$ and $\zeta(\gamma)$ the Riemann ζ function. The associated moment generating functions are

$$G(x) = \frac{\mathrm{Li}_\gamma(x)}{\zeta(\gamma)}, \tag{2.10}$$

$$H(x) = \frac{\mathrm{Li}_{\gamma-1}(x)}{x\zeta(\gamma-1)}, \tag{2.11}$$

where $\mathrm{Li}_n(x)$ is the nth polylogarithm of x.

To generate directed networks, both in- and out-degree distributions are prescribed. The k_i^{in} in-stubs and k_i^{out} out-stubs are assigned to v_i, and the out-stubs are randomly connected to in-stubs.

The disadvantage of such networks is that the only tunable parameter is the degree exponent γ. By introducing $k_{\min} > 1$, we can change $\langle k \rangle$, however the method is not flexible and rather cumbersome.

2.1.3 Hidden parameter model and static model

The hidden parameter model grants a simple method to generate networks with heavy tailed distribution and with smoothly tunable average degree [43, 75, 76, 77]. We start with N unconnected nodes, each node v_i is assigned a hidden parameters $w(i)$. Then two nodes v_i and v_j are randomly selected with probability proportional to $w(i)$ and $w(j)$, respectively. If v_i and v_j are previously unconnected, a new link $(v_i - v_j)$ is formed. Otherwise v_i and v_j is discarded. This process is repeated until $L = cN/2$ links are placed, and c is the average degree. Sparse and large networks generated by the hidden parameter model are also locally treelike.

By properly choosing the hidden parameters, we can tune the degree distributions $p(k)$:

$$p(k) = \sum_{i=1}^{N} \exp\left[-\frac{w(i)}{\sum_i w(i)} cN\right] \frac{(cNw(i)/\sum_i w(i))^k}{k!}, \tag{2.12}$$

The corresponding generating function of this distribution is

$$G(x) = \sum_{k=0}^{\infty} p(k)x^k = \sum_{i=1}^{N} \exp\left[-\frac{w(i)cN}{\sum_i w(i)}(1-x)\right]. \tag{2.13}$$

Note that with choice $w(i) \equiv 1$ the Erdős-Rényi model is recovered.

The static model is a special case of the hidden parameter model that aims to generate networks with scale-free degree distribution [43, 77]. The hidden parameter of each node v_i is set to $w(i) = i^{-\alpha}$ where $i = 1, 2, \ldots N$. For large N, this choice yields the degree distribution

$$p(k) = \frac{[(1-\xi)c]^{1/\xi}}{\xi} \frac{\Gamma(k - 1/\xi, [1-\xi]c)}{\Gamma(k+1)}, \tag{2.14}$$

where $\Gamma(n, x)$ is the upper incomplete gamma function. For large k the degree distribution is power-law:

$$p(k) \sim k^{-(1+1/\xi)} = k^{-\gamma}, \tag{2.15}$$

where $\gamma = 1 + 1/\xi$ determines the exponent of the tail of the distribution. The corresponding generating functions are

$$G(x) = \frac{1}{\xi} E_{1+\frac{1}{\xi}}\left[c(1-\xi)(1-x)\right], \tag{2.16}$$

$$H(x) = \frac{1-\xi}{\xi} E_{\frac{1}{\xi}}\left[c(1-\xi)(1-x)\right] \tag{2.17}$$

where $E_n(x) = \int_1^\infty dt e^{-xt} t^{-n}$ is the exponential integral function.

To generate directed networks, each node v_i is assigned two hidden parameters $w_{\text{in}}(i)$ and $w_{\text{out}}(i)$, and $L = cN$ directed links are randomly place by choosing the start- and endpoint of the link with probability proportional to $w_{\text{in}}(i)$ and $w_{\text{out}}(i)$, respectively. Now the average in- and out-degree is $\langle k \rangle \equiv \langle k_{\text{in/out}} \rangle = L/N = c$. Note that choosing different ξ_{in} and ξ_{out} yields networks that have different degree distributions $p_{\text{in}}(k)$ and $p_{\text{out}}(k)$.

2.2 Core percolation

2.2.1 Greedy leaf removal for undirected networks

The *core* of an undirected network is defined as a subnetwork which remains after the following greedy leaf removal (GLR) procedure [35, 69]: As long as the network has leaves, i.e. nodes of degree one, choose an arbitrary leaf v_1 and its neighbor v_2, and remove them together with all the links incident with v_2. Finally, remove all isolated nodes. The resulting network is independent of the order of removals [69].

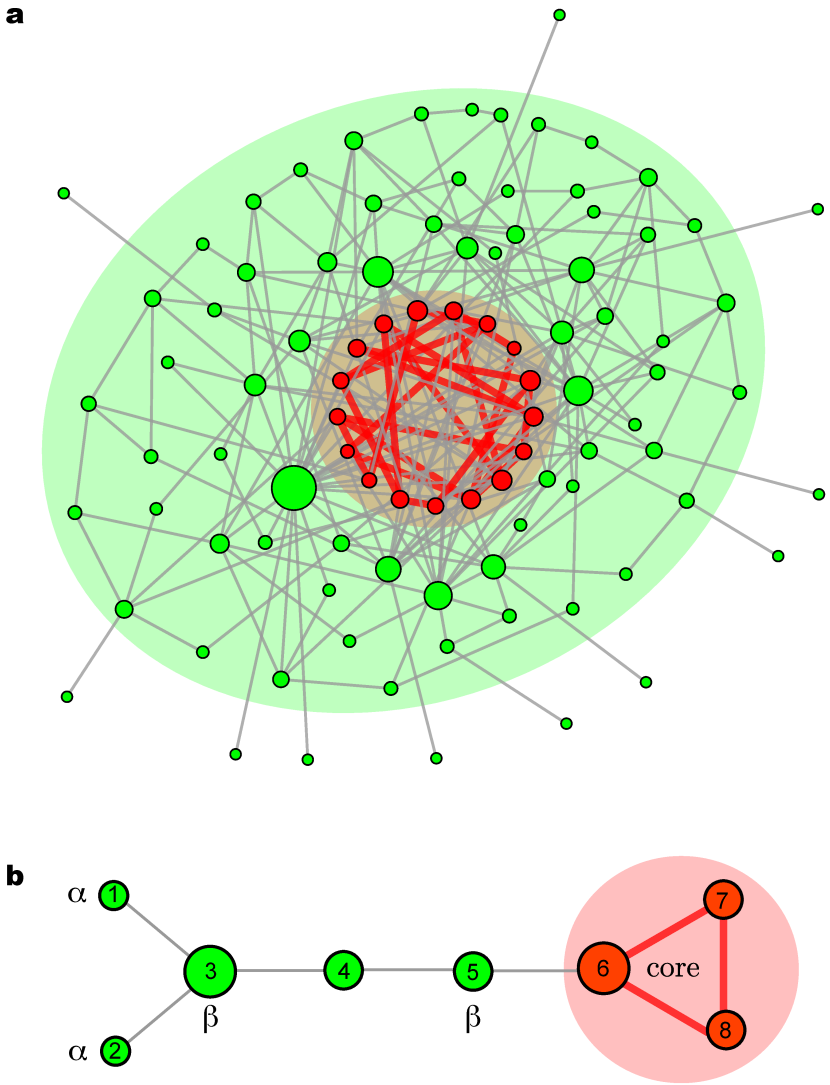


Figure 2.2: **The core of a small network. (a)** The core (highlighted in red) obtained after the greedy leaf removal procedure is fundamentally different from the 2-core (highlighted in green) obtained by iteratively removing nodes of degree less than 2. The 2-core contains the core, whereas the opposite is not true. Size of nodes are roughly proportional to the degree of nodes. **(b)** Removal categories of nodes according to how they can be removed during the greedy leaf removal procedure. Red nodes are non-removable, i.e. they belong to the core. Green nodes are removable: nodes v_1 and v_2 are α-removable; nodes v_3 and v_5 are β-removable. Node v_4 is removable but it is neither α-removable nor β-removable. Node v_5 is β-removable because v_4 will become a leaf node after removing node v_1 (or v_2) together with v_3.

Note that the core described above is fundamentally different from the k-core of a network. The latter is defined to be the maximal subnetwork having minimum node degree of at least k, which can be obtained by iteratively removing nodes of degree less than k. The GLR procedure is more destructive than the node removal procedure used to obtain the 2-core (see Fig. 2.2a).

As a first step towards the analytical solution, the nodes are categorized according to how they can be removed during GLR:

(i) α-removable: nodes that can become isolated (e.g. v_1 and v_2 in Fig.2.2b);

(ii) β-removable: nodes that can become a neighbor of a leaf (e.g. v_3 and v_5 in Fig.2.2b);

(iii) γ-removable: nodes that can become a leaf, but cannot become α-removable nor β-removable. Note that such nodes only exist if there is a core, and the category will not be used in the calculations.

(iv) non-removable: nodes that cannot be removed and hence belong to the core (e.g. v_6, v_7 and v_8 in Fig.2.2b).

All nodes belong to one category only.

Now consider an uncorrelated random network with arbitrary degree distribution $p(k)$. In the large network limit, the category of a node v in a network $\mathcal{G}$ can be determined by the categories of its neighbors in $\mathcal{G} \setminus v$, i.e. the subnetwork of $\mathcal{G}$ with node v and all its links removed, using the following rules:

(i) α-removable: all neighbors are β-removable;

(ii) β-removable: at least one neighbor is α-removable;

(iii) non-removable: no neighbor is α-removable, and at least two neighbors are not β-removable.

Let α and β denote the probability that a random neighbor of a random node v in a network $\mathcal{G}$ is α-removable and β-removable in $\mathcal{G} \setminus v$, respectively. To calculate α and β, two self-consistent equations can be derived by averaging over the node degree

$$\alpha = \sum_{k=1}^{\infty} q(k)\beta^{k-1} = H(\beta), \tag{2.18}$$

$$1-\beta = \sum_{k=1}^{\infty} q(k)(1-\alpha)^{k-1} = H(1-\alpha) \tag{2.19}$$

where $q(k+1)$ is the excess degree distribution. These two equations indicate that α satisfies equation $\alpha = H\big(1-H(1-\alpha)\big)$. If the equation has multiple solutions, α is given by the smallest root of the function $f(x) \equiv H\big(1-H(1-x)\big) - x$.

The expected fraction of non-removable nodes, i.e. the normalized core size ($n_{\text{core}} \equiv N_{\text{core}}/N$), then can be calculated:

$$n_{\text{core}} = \sum_{k=0}^{\infty} p(k) \sum_{s=2}^{k} \binom{k}{s} \beta^{k-s} (1-\beta-\alpha)^s, \tag{2.20}$$

which can be simplified in terms of $G(x) \equiv \sum_{k=0}^{\infty} p(k) x^k$, i.e. the generating function of the degree distribution $p(k)$. The final result is given by

$$n_{\text{core}} = G(1-\alpha) - G(\beta) - \langle k \rangle \, (1-\beta-\alpha)\, \alpha. \tag{2.21}$$

For random networks with Poisson degree distribution, $G(x) = H(x) = e^{-\langle k \rangle (1-x)}$, Eq. (2.21) can be further simplified as $n_{\text{core}} = (1-\beta-\alpha)(1-\langle k \rangle \, \alpha)$, confirming previous results [69].

The normalized number of links in the core ($l_{\text{core}} \equiv L_{\text{core}}/N$) can also be calculated in terms of α and β. Consider a uniform random link, which remains in the core if and only if both of its endpoints are non-removable without removing the link. The probability of one endpoint being non-removable without removing the link is $1-\alpha-\beta$, and for the two endpoints the probabilities are independent. Therefore, the expected normalized number of links in the core is

$$l_{\text{core}} = \frac{\langle k \rangle}{2} (1-\alpha-\beta)^2. \tag{2.22}$$

with $\langle k \rangle / 2 = L/N$ the normalized number of links in the network. Clearly, both $n_{\text{core}} > 0$ and $l_{\text{core}} > 0$ if and only if $1-\beta-\alpha > 0$.

2.2.2 Greedy leaf removal for directed networks

To extend GLR to directed networks, first the directed network $\mathcal{G}$ is transformed to its bipartite network representation $\mathcal{B}$. A bipartite network is a network whose nodes can be partitioned into two disjoint sets V^+ and V^- such that all links connect one node in V^+ with one node in V^-. To obtain the bipartite representation $\mathcal{B}$ each node v is split into two nodes v^+ (upper) and v^- (lower), and we connect v_1^+ to v_2^- in $\mathcal{B}$ if there is a link $(v_1 \to v_2)$ in $\mathcal{G}$ (Fig. 2.3). The core of directed network $\mathcal{G}$ can then be defined as the core of its corresponding bipartite network $\mathcal{B}$ obtained using the original definition of GLR.

Now consider an uncorrelated directed network $\mathcal{G}$ with given in- and out-degree distributions, denoted by $p_{\text{in}}(k)$ and $p_{\text{out}}(k)$, respectively. Or equivalently in the bipartite

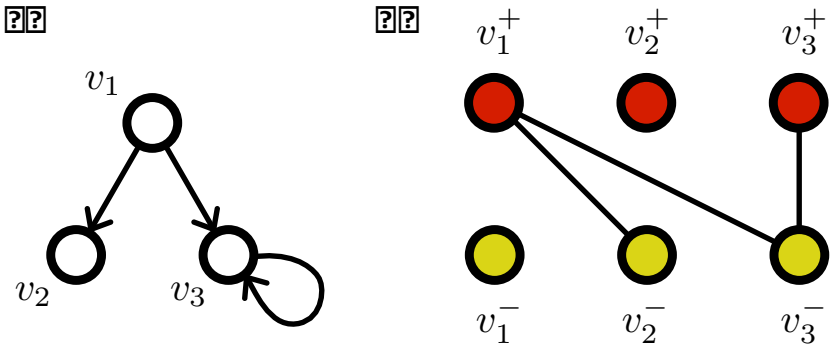


Figure 2.3: **Bipartite representation of directed networks. (a)** A small directed network with four nodes and three links. **(b)** To construct the undirected bipartite representation of a directed network, each node v is split into two nodes v^+ (upper) and v^- (lower), and we connect v_1^+ to v_2^- in the bipartite network if there is a link $(v_1 \to v_2)$ in the original directed network.

representation $\mathcal{B}$, the degree distribution of the lower nodes is $p^-(k) \equiv p_{\text{in}}(k)$ and the degree distribution of the upper nodes is $p^+(k) \equiv p_{\text{out}}(k)$. Let $\langle k \rangle$ denote the mean degree of each partition V^+ and V^-, i.e. the mean in-degree (or out-degree) of $\mathcal{G}$. Define $q^\pm(k) \equiv kp^\pm(k)/\langle k \rangle$, which is the degree distribution of the upper or lower end, respectively, of a random link in $\mathcal{B}$. And define $H^\pm(x) \equiv \sum_{k=0}^{\infty} q^\pm(k+1)x^k$. Then the same argument as we used in the undirected case gives that

$$\alpha^\pm = H^\pm(\beta^\mp), \tag{2.23}$$

$$1 - \beta^\pm = H^\pm(1 - \alpha^\mp) \tag{2.24}$$

and $\alpha^\pm$ is the smallest fixpoint of $H^\pm\big(1 - H^\mp(1-x)\big)$. Now we can calculate the size of the core for each partition in $\mathcal{B}$ as

$$n_{\text{core}}^\pm = \sum_{k=0}^{\infty} p^\pm(k) \sum_{s=2}^{k} \binom{k}{s} (\beta^\mp)^{k-s}(1 - \beta^\mp - \alpha^\mp)^s \tag{2.25}$$

and we define the size of the core in the directed network $\mathcal{G}$ as

$$n_{\text{core}} = (n_{\text{core}}^+ + n_{\text{core}}^-)/2. \tag{2.26}$$

The normalized number of edges in the core can also be calculated

$$l_{\text{core}} = \langle k \rangle \ (1 - \alpha^+ - \beta^+)(1 - \alpha^- - \beta^-). \tag{2.27}$$

2.2.3 Condition for core percolation

It is easy to see that the core in a undirected network with degree distribution $p(k)$ is the very same as in a directed network with the same out- and in-degree distributions, i.e. $p^+(k) = p^-(k) = p(k)$. Hence it is sufficient to consider directed network for generality. As n_{core} is a continuous function of $\alpha^\pm$, therefore properties of $\alpha^\pm$ are investigated, which is the smallest root of the function $f^\pm(x) \equiv H^\pm\big(1 - H^\mp(1-x)\big) - x$.

Several observations can be made about function $f^\pm(x)$. First of all, since $H^\pm(1-x)$ is a monotonically decreasing function for $x \in [0,1]$ and $H^\pm(1) = 1$ is the maximum (see Figs. 2.4, 2.5), we have $f^\pm(0) > 0$ and $f^\pm(1) < 0$ (see Fig. 2.5c,d). Consequently, the number of roots (with multiplicity) of $f^\pm(x)$ in $[0,1]$ is odd, and numerical calculations suggest that this number is either 1 or 3 (see Figs. 2.4, 2.5). Secondly, if $f^\pm(x_0) = 0$ then $f^\mp\big(H^\mp(1-x_0)\big) = 0$, which means $H^\mp(1-x)$ transforms the roots of $f^\pm(x)$ to the roots of $f^\mp(x)$. This also suggests that $f^\pm(x)$ *always* has a trivial root $\alpha^\pm = H^\pm(1-\alpha^\mp) = 1-\beta^\pm$. (For undirected networks, $f(x)$ *always* has a trivial root $\alpha = H(1-\alpha) = 1-\beta$.) Since $H^\mp(1-x)$ is a monotonically decreasing function and $\alpha^\pm$ is the smallest root of $f^\pm(x)$, $H^\mp(1-\alpha^\pm) = 1-\beta^\mp$ is therefore the largest root of $f^\mp(x)$. Hence $1-\beta^\pm-\alpha^\pm$ is the difference between the largest and the smallest roots of $f^\pm(x)$ (see Fig.2.4). Consequently, if $f^\pm(x)$ has only one root (which then must be the trivial root $\alpha^\pm = H^\pm(1-\alpha^\mp) = 1-\beta^\pm$), then $1-\beta^\pm-\alpha^\pm = 0$. According to Eq. (2.25), this implies that there is no core. On the other hand, if multiple roots exist and they are different then $1-\beta^\pm-\alpha^\pm > 0$, and the core will develop.

The above condition is applied to the random undirected networks defined in Sec. 2.1.

(i) Erdős-Rényi networks with Poisson degree distribution $p(k) = e^{-c}c^k/k!$. As shown in Fig. 2.5a, the core percolation occurs at $c = c^* = e$, which agrees with previous theoretical results [35, 69].

(ii) Purely power-law distributed networks with $p(k) = k^{-\gamma}/\zeta(\gamma)$ for $k \geq 1$, $\gamma > 2$ and $\zeta(\gamma)$ is the Riemann ζ function. It is found that $f(x)$ has no multiple roots and hence $n_{\text{core}} = 0$ for all $\gamma > 2$. In other words, for purely scale-free networks, the core does not exist.

(iii) For asymptotically scale-free networks generated by the static model the core may or may not exist depending on the average degree. For small k, $p(k)$ deviates significantly from the power-law distribution and there are much fewer small-degree

nodes than the purely scale-free networks, which results in a drastically different core percolation behavior.

Hereafter, the net effect of adding more links (i.e. increasing mean degree c, yet without changing other parameters in $p(k)$) on core percolation is studied. Erdős-Rényi networks and the asymptotically scale-free networks generated by the static model naturally serve this purpose, since their mean-degree is an independent and explicit tuning parameter.

2.2.4 Nature of core percolation

If the mean degree c is small, then $f^{\pm}(x)$ has one root, but if c is large, $f^{\pm}(x)$ has three roots (see Figs.2.4, 2.5). At the critical point $c = c^*$, the number of roots jumps from 1 to 3 by the appearance of one new root with multiplicity 2. Note that $f^{\pm}(x)$ cannot immediately intersect the x-axis at two new points, but it touches first. This explains why the core percolation occurs at $c = c^*$.

According to the transformation from the roots of $f^{\pm}(x)$ to the roots of $f^{\mp}(x)$ through $H^{\mp}(1-x)$, for either $f^{+}(x)$ or $f^{-}(x)$ (depending on the details of $p^{+}(k)$ and $p^{-}(k)$) its new root at $c = c^*$ is smaller than its original root; and for either $f^{-}(x)$ or $f^{+}(x)$ the new root at $c = c^*$ is larger than the original root; or there is a degenerate case when this new root is the same as the original root for both $f^{+}(x)$ and $f^{-}(x)$. For example, for directed asymptotically scale-free networks generated by the static model with $\gamma_{\text{in}} = 2.7, \gamma_{\text{out}} = 3.0$, the new root (marked as green dot) of $f^{+}(x)$ at $c = c^*$ is smaller than the original root (green square) of $f^{+}(x)$ (see Fig.2.5c), and the new root (green square) of $f^{-}(x)$ at $c = c^*$ is larger than the original root (green circle) of $f^{-}(x)$ (see Fig. 2.5d). In other words, at the critical point, for either $f^{+}(x)$ or $f^{-}(x)$, its smallest two roots are the same, and for the other function (either $f^{-}(x)$ or $f^{+}(x)$), its largest two roots are the same (see Fig. 2.5c,d). While for directed networks with $p^{+}(k) = p^{-}(k) = p(k)$, i.e. the degenerate case, we have $f^{+}(x) = f^{-}(x) = f(x)$, and the new root of $f(x)$ at $c = c^*$ has to be the same as the the original root of $f(x)$, i.e. all three roots must be the same (see Fig. 2.5a). Therefore at the critical point, unless in the degenerate case, α^{+} together with β^{-} (or α^{-} together with β^{+}) decrease discontinuously, which implies a discontinuous transition in the core size.

To sum up, in the degenerate case that $p^{+}(k) = p^{-}(k) = p(k)$ core percolation is continuous, but for general non-degenerate case $p^{+}(k) \neq p^{-}(k)$, we have a discontinuous transition in both n_{core} and l_{core}. These results are clearly shown in Fig. 2.5b,e.

At the critical point c^*, $f^{\pm}(x)$ touches the x-axis at its new root (see Fig. 2.5c,d), hence we have either $f^{+}(\alpha^{+}) = (f^{+})'(\alpha^{+}) = 0$ (or $f^{-}(1-\beta^{-}) = (f^{-})'(1-\beta^{-}) = 0$), which enable us to calculate the core percolation threshold c^*. In the degenerate case, if $c \leq c^*$

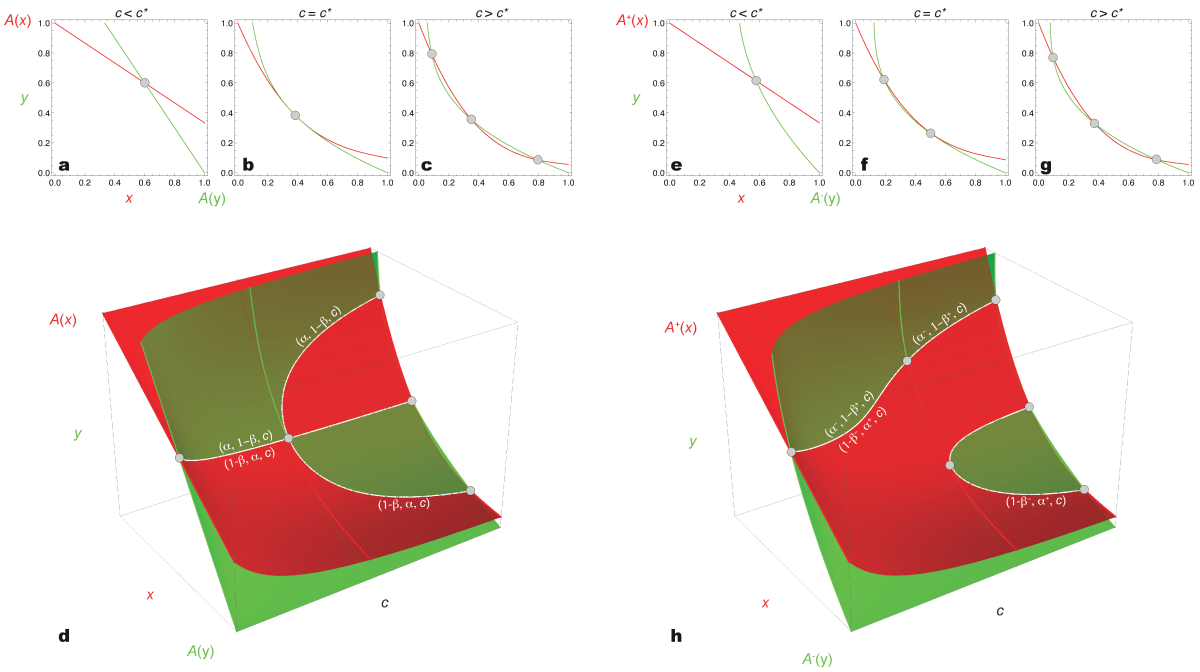


Figure 2.4: **Graphical solution of the self-consistent equations.** **(a-d)** For undirected networks, the function $H(1-x)$ transforms the roots of $f(x)$ to the roots of the same function $f(x)$. The graphical solution of $f(x) = H\big(1 - H(x)\big) - x = 0$ is best illustrated by plotting the two curves $H(1-x)$ vs. x (in red) and y vs. $H(1-y)$ (in green) in the same coordinate system. The coordinates of the intersection point(s) of the two curves give the solution(s) of $f(x) = 0$. **(a-c)** Graphical solution for $c <, =$, and $> c^*$. **(d)** By drawing the two curves ($H(1-x)$ vs. x) and (y vs. $H(1-y)$) at different mean degrees c, we get two surfaces. The intersection curve of the two surfaces yields the solutions of $f(x) = 0$ at different c values. For $c < c^*$, the intersection curve has one branch given by $(\alpha, 1-\beta, c) = (1-\beta, \alpha, c)$. For $c > c^*$, the intersection curve has three branches. The top and bottom branches are given by $(\alpha, 1-\beta, c)$ and $(1-\beta, \alpha, c)$, respectively. **(e-h)** For directed networks, $H^{\pm}(1-x)$ transforms the roots of $f^{\mp}(x)$ to the roots of $f^{\pm}(x)$. The graphical solution of $f^{\pm}(x) = H^{\pm}(1 - H^{\mp}(1-x)) - x = 0$ can be illustrated by plotting $H^{+}(1-x)$ vs. x (in red) and y vs. $H^{-}(1-y)$ (in green) in the same coordinate system. The x-coordinate (or y-coordinate) of the intersection point(s) of the two curves give the solution(s) of the equation $f^{-}(x) = 0$ (or $f^{+}(x) = 0$, respectively). **(e-g)** Graphical solution for $c <, =$, and $> c^*$. **(h)** By drawing the two curves ($H^{+}(1-x)$ vs. x) and (y vs. $H^{-}(1-y)$) at different mean degrees c, we get two surfaces. The intersection curve of the two surfaces yields the solutions of $f^{\pm}(x) = 0$ at different c values. For $c < c^*$, the intersection curve has one branch given by $(\alpha^{-}, 1-\beta^{+}, c) = (1-\beta^{-}, \alpha^{+}, c)$. For $c > c^*$, the intersection curve has three branches. The top and bottom branches are given by $(\alpha^{-}, 1-\beta^{+}, c)$ and $(1-\beta^{-}, \alpha^{+}, c)$, respectively.

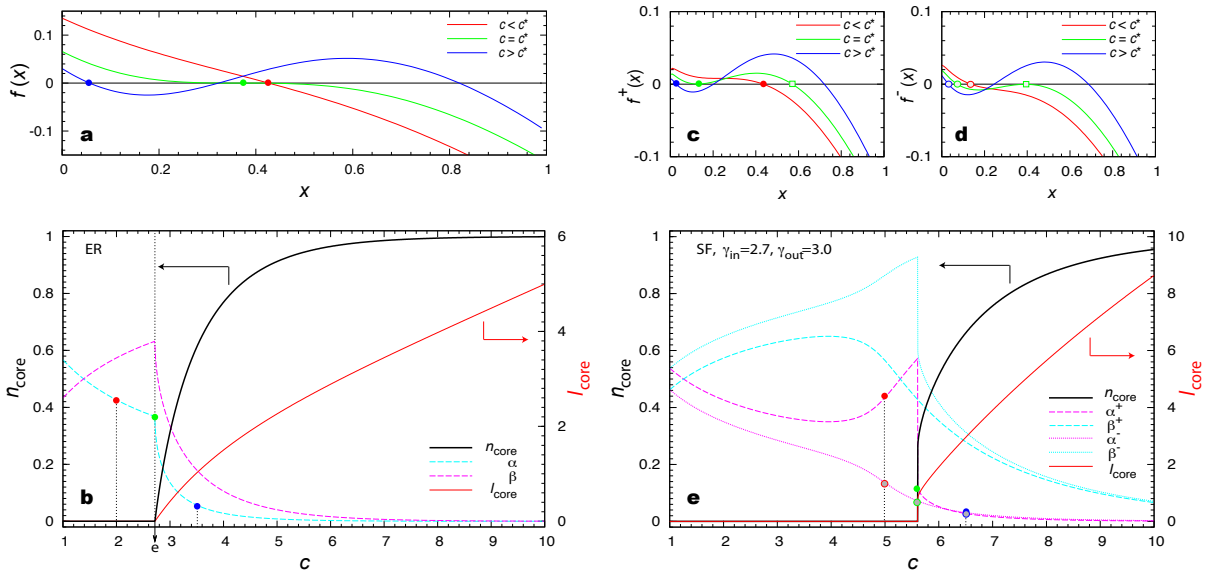


Figure 2.5: **Analytical solution of the core percolation. (a-b)** Undirected Erdős-Rényi random networks. **(a)** α is the smallest root of the function $f(x) \equiv H\big(1 - H(1 - x)\big) - x$, represented by red, green, and blue dots for $c <, =,$ and $> c^* = e$, respectively. **(b)** $\alpha, \beta, n_{\rm core}$ and $l_{\rm core}$ as functions of the mean degree c. **(c-e)** Directed asymptotically scale-free random networks generated by the static model. Both the in-degree and out-degree distributions of the networks are scale-free with degree exponents $\gamma_{\rm in} = 2.7$ and $\gamma_{\rm out} = 3.0$. **(c,d)** $\alpha^{\pm}$ is the smallest root of the function $f^{\pm}(x) \equiv H^{\pm}\big(1 - H^{\mp}(1 - x)\big) - x$, represented by red, green, and blue dots for $c <, =,$ and $> c^* \simeq 11.2$, respectively. **(e)** $\alpha^{\pm}, \beta^{\pm}, n_{\rm core}$ and $l_{\rm core}$ as functions of the mean degree c. The jumps in α^+ and β^- result in the jumps in $n_{\rm core}$ and $l_{\rm core}$, hence the first-order core percolation occurs.

then $f(\alpha) = f'(\alpha) = 0$ can be further simplified as $H(1 - \alpha) = \alpha$ and $[H'(1 - \alpha)]^2 = 1$. The results of c^* for Erdős-Rényi and scale-free networks generated by the static model are shown in Fig. 2.6a.

The discontinuity in $n_{\rm core}$ and $l_{\rm core}$ at c^*, denoted by $\Delta_{\rm n}$ and $\Delta_{\rm l}$ respectively, can also be calculated

$$\Delta_{\rm n} = \frac{1}{2}\left(\Delta_{\rm n}^{+} + \Delta_{\rm n}^{-}\right) \tag{2.28}$$

$$\Delta_{\rm l} = c^*(1 - \beta^{-,*} - \alpha^{-,*})(1 - \beta^{+,*} - \alpha^{+,*}) \tag{2.29}$$

with $\Delta_{\rm n}^{\pm} \equiv G^{\pm}(1 - \alpha^{\mp,*}) - G^{\pm}(\beta^{\mp,*}) - c^*\left(1 - \beta^{\mp,*} - \alpha^{\mp,*}\right)\alpha^{\pm,*}$. The results of $\Delta_{\rm n}$ for Erdős-Rényi and scale-free networks generated by the static model are shown in Fig. 2.6b. It can be seen that $\Delta_{\rm n} \to 0$ as $\gamma_{\rm in} \to \gamma_{\rm out}$, consistent with the result obtained above that core percolation is continuous for undirected networks or directed networks with $p^+(k) = p^-(k)$. Furthermore, $\Delta_{\rm n}$ increases as the differences between $\gamma_{\rm in}$ and $\gamma_{\rm out}$ increases.

It can be shown that in the general non-degenerate case, core percolation is actually

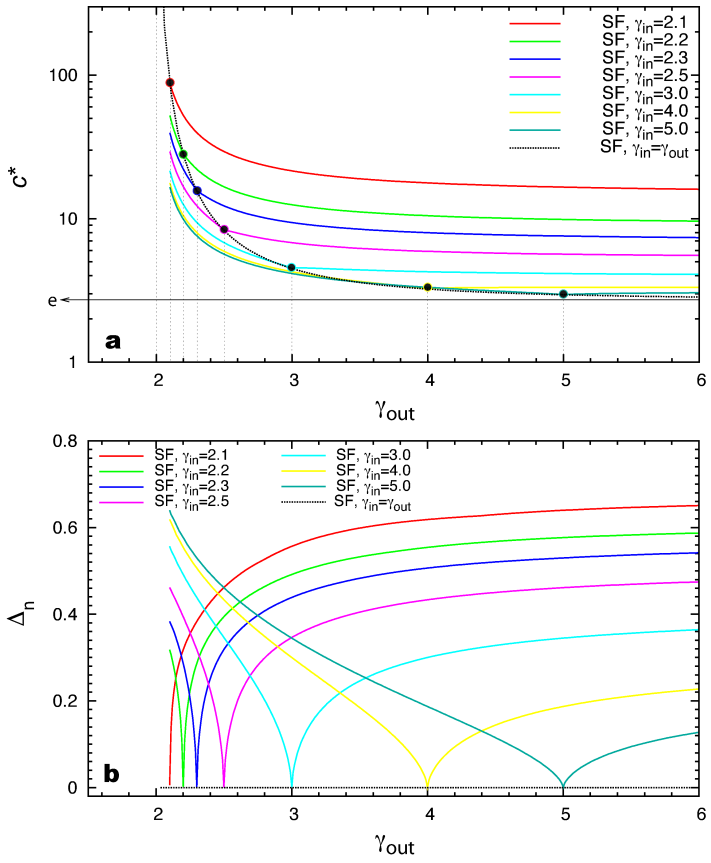


Figure 2.6: **Threshold and discontinuity of core percolation. (a)** Analytical solution of the core percolation threshold c^* calculated by solving $f^{\pm}(x) = f^{\pm\prime}(x) = 0$ for model networks. For Erdős-Rényi networks, $c^* = e$. For undirected asymptotically scale-free networks generated by the static model, $c^* \to \infty$ as $\gamma \to 2$, and and $c^* \to e$ as $\gamma \to \infty$. **(b)** The discontinuity $\Delta_{\rm n}$ in $n_{\rm core}$ at $c = c^*$ for model networks. For undirected or directed networks with $p^+(k) = p^-(k)$, $\Delta_{\rm n} = 0$. For directed network, $\Delta_{\rm n}$ increases as the difference between the in- and out-degree distributions (quantified by the difference between the degree exponents $\gamma_{\rm in}$ and $\gamma_{\rm out}$) increases.

a hybrid phase transition [58], i.e. $n_{\rm core}$ (or $l_{\rm core}$) has a jump at the critical point as at a first-order phase transition but also has a critical singularity as at a continuous transition. The results are summarized here: in the critical regime $\epsilon = c - c^* \to 0^+$

$$n_{\rm core} - \Delta_{\rm n} \sim (c - c^*)^{\eta} \tag{2.30}$$

$$l_{\rm core} - \Delta_{\rm l} \sim (c - c^*)^{\theta} \tag{2.31}$$

with the critical exponents $\eta = \theta = \frac{1}{2}$. The calculations do not use any specific functional form of $H^{\pm}(1-x)$. Instead, it is only assumed that they are continuous functions of the mean degree c. Interestingly, in the degenerate or undirected case, one has a continuous phase transition ($\Delta_{\rm n} = \Delta_{\rm l} = 0$) but with a completely different set of critical exponents: $\eta' = \theta' = 1$ [69]. Detailed calculations of the critical exponents can be found in the Supplementary Information of Ref. [19].

2.2.5 Numerical verification

The analytical results are checked with extensive numerical calculations by performing the GLR procedure on finite discrete networks. Figs. 2.7a and 2.7b show $n_{\rm core}$ and $l_{\rm core}$ (in symbols) for undirected Erdős-Rényi networks and asymptotically scale-free networks with different degree exponents. For comparison, analytical results for infinite large networks are also shown (in lines). Clearly, core percolation is continuous in this case. This is fundamentally different from the $k \geq 3$-core percolation, which becomes discontinuous for Erdős-Rényi networks and scale-free networks with $\gamma > 3$ [61, 62].

Figures 2.7c and 2.7d show the results of $n_{\rm core}$ and $l_{\rm core}$ for directed networks. For directed networks with the same in- and out-degree distributions, e.g. directed Erdős-Rényi networks and directed scale-free with $\gamma_{\rm in} = \gamma_{\rm out}$ generated by the static model, the core percolation is still continuous. But for directed networks with different in- and out-degree distributions, e.g. directed scale-free networks with $\gamma_{\rm in} \neq \gamma_{\rm out}$ generated by the static model, the core percolation looks discontinuous. The discontinuity in $n_{\rm core}$ (or $l_{\rm core}$) increases as the difference between $\gamma_{\rm in}$ and $\gamma_{\rm out}$ increases (see Fig. 2.7e,f).

2.3 Relation to matching

The GLR procedure can be used to construct a maximum matching in both unipartite and bipartite undirected networks [35, 22, 23]. For each step in GLR, a leaf is removed that consist of node v_i with degree 1, and node v_j with possibly higher degree. To construct the maximum matching, the link $(v_i - v_j)$ is added to the matching. Now all links adjacent to v_j are not allowed in the matching, and therefore have to be removed too. Any node that

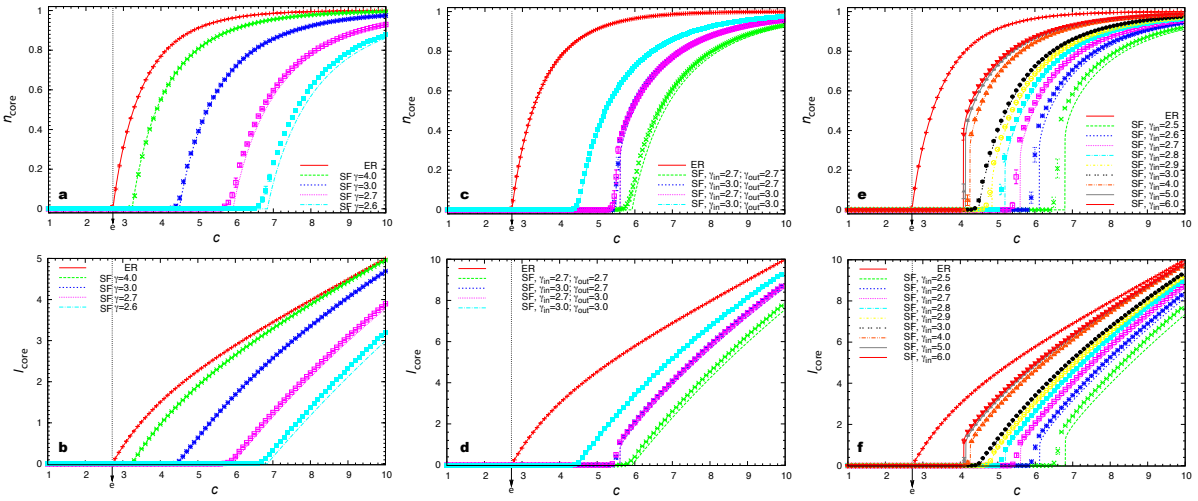


Figure 2.7: **Core percolation in random networks.** Symbols are numerical results calculated from the GLR procedure on finite discrete networks constructed with the Erdős-Rényi model and the static model with $N = 10^5$. The numerical results are averaged over 20 realizations with error bars defined as standard error of the mean. Lines are analytical results for infinite large system ($N \to \infty$) calculated from Eqs. (2.21) and (2.22) for undirected networks or Eqs. (2.26) and (2.27) for directed networks. **(a-b)** The normalized core size ($n_{\text{core}} = N_{\text{core}}/N$) and the normalized number of edges in the core ($l_{\text{core}} = L_{\text{core}}/N$) for undirected model networks: Erdős-Rényi and asymptotically scale-free with different values of γ. **(c-d)** n_{core} and l_{core} for directed Erdős-Rényi and asymptotically scale-free model networks. The core percolation is continuous if the out- and in-degree distributions are the same ($p^+(k) = p^=(k)$) while it becomes discontinuous if $p^+(k) \neq p^=(k)$. **(c-d)** For directed scale-free networks with fixed $\gamma_{\text{out}} = 3.0$, by tuning γ_{in} we see that the discontinuity in both n_{core} and l_{core} become larger as the difference between γ_{in} and γ_{out} increases.

becomes isolated remains unmatched. This is continued, until no leaves are left, i.e. only the core remains (Fig. 2.8). Finding the maximum matching in the core cannot be done in a greedy manner. However, it was shown that in large non-bipartite random networks the core can be asymptotically matched, i.e. the probability of randomly choosing an unmatched node is zero [68]. This allows to compute the relative size of the matching: as $N \to \infty$, by assuming that all nodes in the core are matched only a diminishing fraction of nodes are identified matched mistakenly. However, in bipartite networks there is another limiting factor: if the size of the core is different on the two sides ($n^+_{\text{core}} \neq n^-_{\text{core}}$), the size of the matching in the core cannot be larger than the smaller side. Note that this limitation was not considered in Ref. [26] and in the subsequent Ref. [20]. However, in these papers only networks with $p^+(k) = p^-(k)$ networks are studied, therefore their results remain valid.

In the following, the size of maximum matching is calculated for undirected bipartite

representations of directed networks. Let M be the number of matched nodes in either V^+ or V^-. And $m = M/N$ is the relative size of the maximum matching. As stated above, m is the sum of the contribution of the leaf removal and the core.

First, the contribution of leaf removal is calculated. For each leaf removal, one link is added to the matching, increasing the number of matched nodes by 2, one on both sides. For each β-removable node there is one leaf removal. Therefore, to calculate the contribution of the leaf removal, the β-removable nodes are counted on both sides. Node is β-removable, if in its absence atleast one of its neighbors is an α-node. This rule translates to

$$n_\beta^\pm = 1 - G^\pm(1 - \alpha^\mp). \tag{2.32}$$

However, by counting all β-nodes, we have double counted the case when two β-nodes are removed together. This can only happen, if in the absence of the link connecting the two nodes, both nodes are α-nodes, the probability of this event is $\alpha^+\alpha^-$ for each link. Therefore, the overall contribution is

$$[1 - G^+(1 - \alpha^-)] + [1 - G^-(1 - \alpha^+)] - L\alpha^+\alpha^-. \tag{2.33}$$

To determine the contribution of the core, the size of the core on both sides has to be calculated. The following formula is obtained by transforming Eq. (2.25):

$$n_{\text{core}}^\pm = G^\pm(1 - \alpha^\mp) - G^\pm(\beta^\mp) - c\alpha^\pm(1 - \beta^\mp - \alpha^\mp). \tag{2.34}$$

If $n_{\text{core}}^+ \neq n_{\text{core}}^-$, the smaller side has to be selected. Therefore, all together

$$m = 1 - G^-(1 - \alpha^+) + 1 - G^+(1 - \alpha^-) - c\alpha^+\alpha^- + \min_\pm n_{\text{core}}^\pm \tag{2.35}$$

is obtained.

If $n_{\text{core}}^+ = n_{\text{core}}^-$, the contribution of the core can be calculated as the average core size Eq. (2.26):

$$\begin{aligned} m = 2 - \frac{1}{2}\big\{&G^-(1 - \alpha^+) + G^-(\beta^+) + G^+(1 - \alpha^-) + G^+(\beta^-) + \\ &+ c\left[\alpha^+(1 - \beta^-) + \alpha^-(1 - \beta^+)\right]\big\}, \end{aligned} \tag{2.36}$$

According to the minimum input theorem, the unmatched nodes have to be used as driver

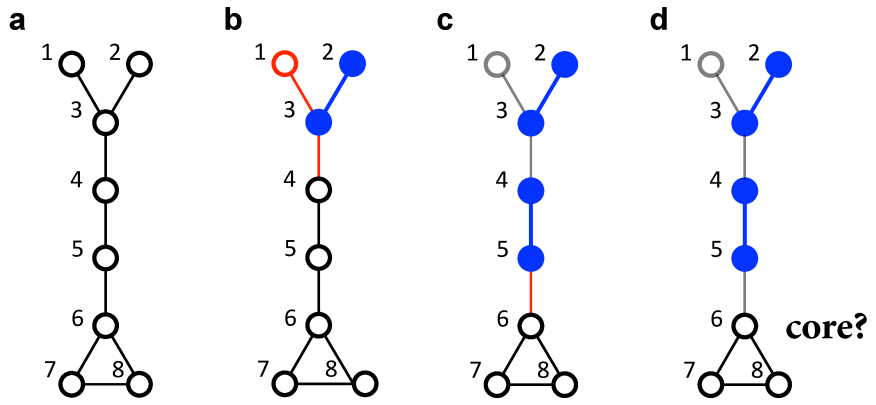


Figure 2.8: **Constructing the maximum matching via greedy leaf removal.** **(a)** The maximum matching of an undirected network of eight nodes is constructed. **(b)** In the first step step of the greedy leaf removal, the leaf v_2 is removed, therefore link $(v_2 - v_3)$ is added to the matching (blue link), now the adjacent links (red links) are no longer allowed in the matching, therefore they are removed. Node v_1 becomes isolated, therefore it remains unmatched. **(c)** In the next step link $(v_4 - v_5)$ (blue) is added to the marching. **(d)** Finding the maximum matching in the remaining core cannot be done using a greedy algorithm. However, in large uncorrelated networks the almost all nodes in the core are matched.

nodes to impose full control on the network. Therefore the fraction of driver nodes is

$$n_{\mathrm{D}} = 1 - m = \frac{1}{2}\Big\{ \left[G^{-}(\beta^{+}) + G^{-}(1-\alpha^{+}) - 1\right] + \left[G^{+}(\beta^{-}) + G^{+}(1-\alpha^{-}) - 1\right] + \\ + c\left[\alpha^{+}(1-\beta^{-}) + \alpha^{-}(1-\beta^{+})\right]\Big\}, \tag{2.37}$$

recovering the result first derived in Ref. [26]. Whenever dealing with networks with $n^{+}_{\mathrm{core}} = n^{-}_{\mathrm{core}}$ in the following chapters, I rely on Eq. (2.37) to assure compatibility with the literature.

2.4 Summary

This chapter focused on core percolation, a structural transition in complex networks that is connected to the maximum matching problem. In turn, maximum matching is connected to controllability through the minimum input theorem. This connection is exploited in later chapters to shed light on the relation of network structure and control.

Uncorrelated model networks with prescribed degree distributions are essential tools of network science. Such models capture important properties of complex networks while remaining simple enough for analytical treatment. Section 2.1 introduced the models used in the chapter and – with some modifications – in later parts of the thesis.

In Sec. 2.2, core percolation was studied on uncorrelated networks. The definition of

the core and the greedy leaf removal process were introduced and extended to the previously unstudied case of directed networks by applying the procedure to the undirected bipartite representation of directed networks. Analytical solution of the core size was derived for both undirected and directed cases in the large network limit. For Poisson and scale-free distributed networks with adjustable average degree c, it was found that the relative size of the core n_{core} is zero below a critical average degree c^*, and above c^* the core spans finite fraction of the network. In case of undirected networks and directed networks with the same in- and out-degree distributions the transition is continuous. For directed networks with different in- and out-degree distributions, core percolation is a hybrid transition. This means that n_{core} is discontinuous at the critical point c^* – a characteristic of first order transitions, however, after the jump and in the vicinity of the critical point the core size is $n_{\text{core}} \sim (c-c^*)^\eta$ – a characteristic of second order transitions. It was shown that in scale-free networks c^* is greater than in Erdős-Rényi networks.

The relationship between core percolation and maximum matching was discussed in Sec. 2.3. It was shown that greedy leaf removal can be used to construct a maximum matching in a given network. In each step of the procedure one link is added to the matching until only the core remains. Finding an exact maximum matching in the core cannot be done in a greedy manner, however, in the large network limit the core is asymptotically matched, meaning that the probability of randomly finding an unmatched node is zero. This connection is the most important message of the chapter as it serves as the basis of later investigations.

Chapter 3

Network structure and controllability

Structural controllability provides a principled method to study how network structure affects our ability to control complex networks. In this chapter, I apply the minimum input theorem to study two distinct problems. In Sec. 3.1, I investigate how network features beyond the degree distribution influence the number of driver nodes necessary to impose full control on the network. And in Sec. 3.2, the nodes are categorized according to their role in control based on the observation that multiple possible maximum matching configurations can exist. I show that above a critical density of links, two control modes emerge: either majority of nodes can be used as driver nodes or driver nodes can only be selected from a small subset of nodes. I provide evidence that structure of the core determines the control mode of uncorrelated model networks.

This chapter is largely based on Refs. [20, 21, 23].

3.1 Beyond the degree distribution

Liu et al. found that N_{D} is mainly determined by the degree distribution $p(k^{\mathrm{in}}, k^{\mathrm{out}})$. It is known, however, that a series of characteristics, from degree correlations [78, 41, 79] to local clustering [12] and communities [80], cannot be accounted for by $p(k^{\mathrm{in}}, k^{\mathrm{out}})$ alone, prompting the question: which network characteristics affect the system's controllability?

To motivate the study, the observed N_{D} is compared to the prediction based on the degree sequence for several real networks (Table 3.1). For this each network is randomized preserving only its degree sequence and eliminating all other structure [41]. The degree preserving randomization is preformed the following way. Two links $(v_i \to v_j)$ and $(v_k \to v_l)$ are randomly chosen, the two links are broken and rewired creating new links $(v_i \to v_l)$ and $(v_k \to v_j)$. This step is repeated until a steady state is reached. Then $N_{\mathrm{D}}^{\mathrm{rand}}$, the

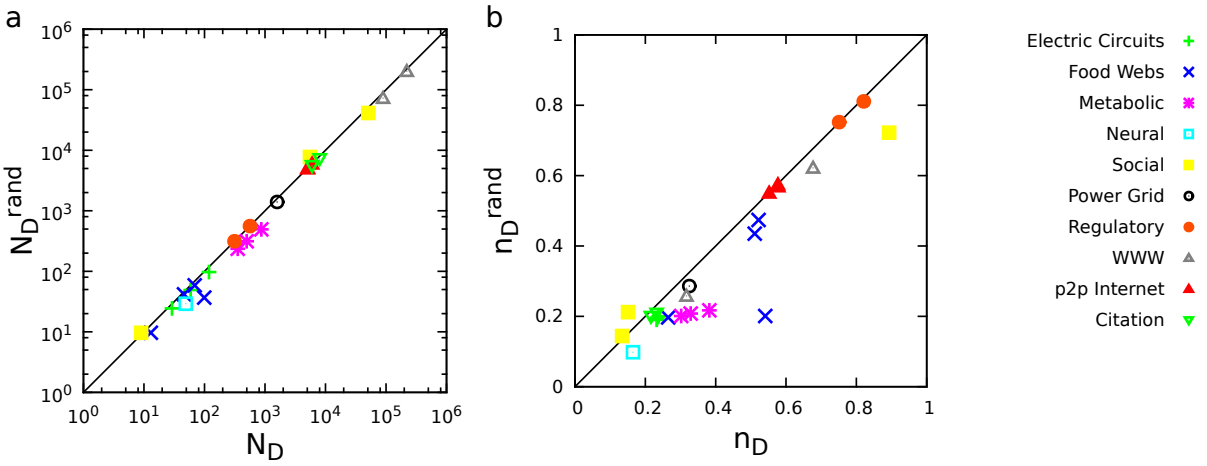


Figure 3.1: **Randomization of real networks. (a)** Comparison of $N_{\rm D}$ for real systems and $N_{\rm D}^{\rm rand}$, representing the number of driver nodes needed to control their randomized counterparts. Randomization eliminates all local and global correlations, only preserving the degree sequence of the original system. It is found that the degree sequence predicts the order of magnitude of $N_{\rm D}$ correctly, however, small deviations are hidden by the log scale, needed to show the whole span of $N_{\rm D}$ seen in real systems. **(b)** These deviations are more obvious if the density of driver nodes $n_{\rm D} = N_{\rm D}/N$ and $n_{\rm D}^{\rm rand}$ are compared in linear scale, finding that for some systems (e.g. regulatory and p2p Internet networks) the degree sequence serves as a good predictor of $n_{\rm D}$, while for other systems (e.g. metabolic networks and food webs) $n_{\rm D}$ deviates from the prediction based solely on the degree sequence.

number of driver nodes for the randomized network, is calculated. Plotting $N_{\rm D}$ versus $N_{\rm D}^{\rm rand}$ on log-log scale indicates that the degree sequence correctly predicts the order of magnitude of $N_{\rm D}$ despite known correlations (Fig. 3.1a). However, by plotting $n_{\rm D} = N_{\rm D}/N$ versus $n_{\rm D}^{\rm rand} = N_{\rm D}^{\rm rand}/N$ clear deviations are observed from the degree based prediction (Fig. 3.1b). The goal is to understand the origin of these deviations and the degree to which network correlations can explain the observed $n_{\rm D}$.

In Sec. 3.1.1, I test the effect of various network features using numerical simulations to identify which characteristics are important with respect to controllability. The list of these features include the most commonly studied deviations from the random network configuration: (i) clustering, manifested as a higher clustering coefficient C than expected based on the degree distribution [81]; (ii) community structure, representing the agglomeration of nodes into distinct communities, captured by the modularity parameter Q [82]; (iii) single node degree correlation, the correlation between the in-degree $k_{\rm in}$ and the out-degree $k_{\rm out}$ of the same node; and (iv) degree-degree correlations of connected node pairs [83]. In addition, I also include the density of self-loops in the survey, because of the possible importance of self-loops in controllability [46].

These numerical simulations provide evidence that – apart from the degree distribution – degree-degree correlations are the most important network characteristic influencing N_{D}. In Sec. 3.1.2, I develop an analytical solution for n_{D} for class of correlated model networks. The analytical solution confirms the numerical results, and offers insight into mechanism of the effect. In Sec. 3.1.3, further evidence is obtain supporting the importance of degree correlations by preforming measurement in real networks.

3.1.1 Numerical simulations

To generate networks with prescribed features, first uncorrelated directed networks are generated with Poisson or scale-free degree distribution. Then simulated annealing is used to add various network characteristics by link rewiring, while leaving the in- and out-degrees unchanged, tuning each measure to a desired value. Finally, n_{D} is computed using the Hopcroft-Karp algorithm [34].

Generating uncorrelated scale-free networks

Scale-free network are generated by the static model similarly as described in Sec. 2.1 with a few differences. First, N disconnected nodes are created and weight $w(i) = (i + i_0)^{-\xi}$ is assigned to each node i ($i = 1 \dots N$). Two nodes v_i and v_j are randomly selected with probability proportional to $w(i)$ and $w(j)$, respectively. If they are not yet connected, link $(v_i \to v_j)$ is added. Self-loops are allowed, but multi-links are avoided. The process is repeated until L links have been placed. The resulting network has average degree $\langle k_{\mathrm{in/out}} \rangle = L/N$, $p_{(\mathrm{in/out})}(k) \sim k^{-\gamma}$ for large k, where $\gamma = 1 + \frac{1}{\xi}$, and maximum degree $k_{\max} \sim i_0^{-\xi}$. Note that for simplicity only networks with $p(k^{\mathrm{in}}) \equiv p(k^{\mathrm{out}})$ are considered here, choosing $p(k^{\mathrm{in}}) \neq p(k^{\mathrm{out}})$ does not qualitatively change the picture.

To systematically study correlations, the starting network has to be uncorrelated. However, the presence of hubs combined with the constraint that multi-links are not allowed may induce unwanted degree correlations [84], and may also considerably limit the maximum and minimum correlations accessible via rewiring [85]. To overcome these difficulties, the so-called structural cutoff is introduced in the degrees by choosing i_0 to ensure $k_{\max} < (\langle k \rangle N)^{1/2}$ [86]. Note that $i_0 = 0$ recovers the model described in Sec. 2.1.

As both in- and out-degree of node v_i is proportional to $w(i)$, the above procedure results in correlations between the in- and out-degrees of node v_i. To eliminate these correlations, the in-degree sequence is shuffled while keeping the out-degree sequence unchanged.

Rewiring algorithm

Degree preserving rewiring is used to add each network characteristic [41]. Suppose that the chosen network characteristic is quantified by a metric X. To set its value to X^*, $E(X) = |X - X^*|$ energy is defined, so $E(X^*)$ is a global minimum. This energy is minimized by simulated annealing [87]:

(1) Two links $(v_i \to v_j)$ and $(v_k \to v_l)$ are chosen at random with uniform probability.

(2) The two links are rewired creating new links $(v_i \to v_l)$ and $(v_k \to v_j)$ and the energy $E(X)$ is recalculated for the new network.

(3) The new configuration is accepted with probability

$$p = \begin{cases} 1, & \text{if } \Delta E \leq 0 \\ e^{-\beta \Delta E}, & \text{if } \Delta E > 0, \end{cases}, \tag{3.1}$$

where the β parameter is the inverse temperature.

(4) The procedure is repeat from step one and β is gradually increased. The procedure stops if $|E(X) - E(X^*)|$ is smaller than a predefined value.

Note that keeping the degree sequence bounds the possible values of X that can be reached by rewiring. In all cases we study the full interval of accessible X values.

Clustering

An apparent way real networks differ from random uncorrelated networks is the high number of closed triangles [12, 81]. For example consider your own social network, two of your friends are more likely to know each other than you would expect by chance, i.e. the probability that two randomly selected individuals are friends. This property is not unique to social networks, almost all real networks are characterized by high clustering. In contrast to this, in large uncorrelated random networks – such as the Erdős-Rényi model and the static model – the probability that two neighbors of a given node are connected tends to zero.

To quantify to what extent are triangles over-represented, the global clustering coefficient C is used [81]. For directed networks it is defined as

$$C = \frac{3 \cdot \text{ number of triangles}}{2 \cdot \text{number of adjacent link pairs}}. \tag{3.2}$$

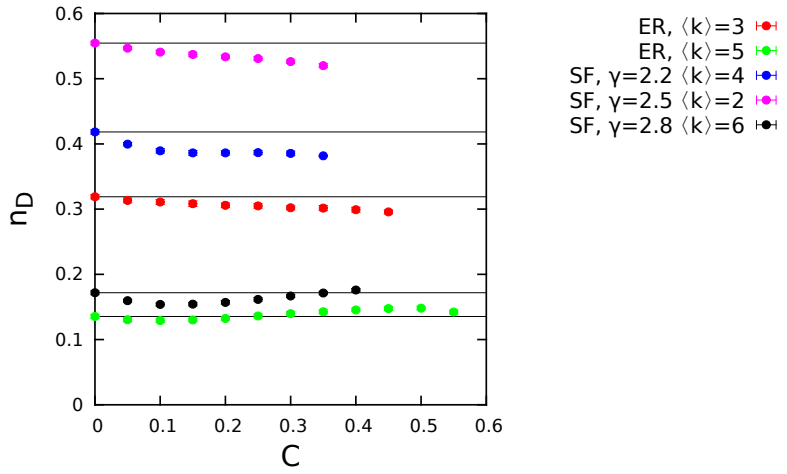


Figure 3.2: **Effect of the clustering coefficient C on the density of driver nodes n_{D}.** Network size is $N = 10,000$, $\langle k \rangle = \langle k^{\mathrm{in}} \rangle + \langle k^{\mathrm{out}} \rangle$ is the average total degree. Each data point is an average over 50 independent runs; the error bars, typically smaller than the symbol size, represent the standard deviation of the measurements.

The simulations indicate that changes in C only slightly alter n_{D} and that the effect is not systematic (Fig. 3.2). Alternative definitions of the clustering coefficient [88] yield the same result. Hence it is concluded that C plays a negligible role in determining n_{D}.

Community structure

Another way real networks are different from the simple uncorrelated model is community structure [80]. There is no universally accepted rigorous definition of a community in a network, usually it is described as a set of nodes that are more densely connected to each other than to the rest of the network. Using the social network example, communities such as families or students in the same class form tightly knit groups with more connections inside the group than outside. The elusiveness of the problem inspired a long list of methods to identify communities and to asses their quality [89, 90, 63, 91, 92, 93]. Here it is not the goal to identify communities, but to plant predefined communities in a network. For this each node is randomly assigned to one of N_{C} equally sized groups. Given the preassigned communities, modularity Q is used to quantify how pronounced the community structure is, for directed networks it is defined as [82]

$$Q = \frac{1}{E} \sum_{ij} \left[A_{ij} - \frac{k_i^{\mathrm{in}} k_j^{\mathrm{out}}}{E} \right] \delta_{c_i, c_j}, \tag{3.3}$$

where A_{ij} is the adjacency matrix, c_i and c_j indicate the communities that nodes v_i and v_j belong to, respectively. Modularity measures the fraction of links inside the communities

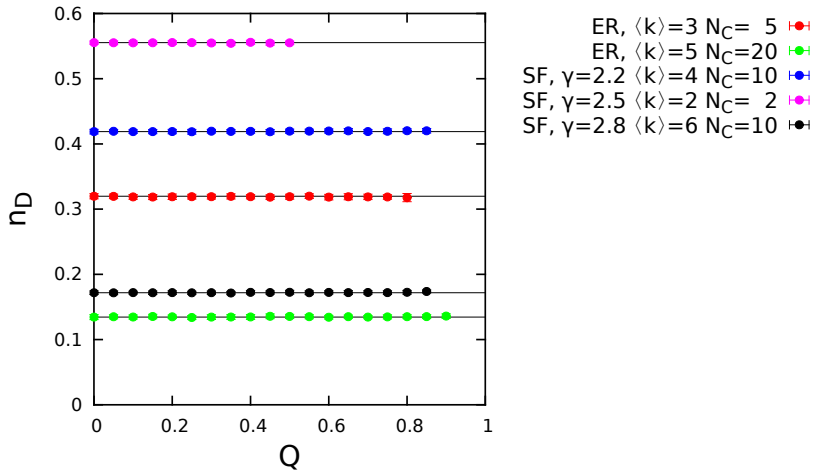


Figure 3.3: **Effect of modularity Q on the density of driver nodes n_{D}.** Network size is $N = 10,000$, $\langle k \rangle = \langle k^{\mathrm{in}} \rangle + \langle k^{\mathrm{out}} \rangle$ is the average total degree. Each data point is an average over 50 independent runs; the error bars, typically smaller than the symbol size, represent the standard deviation of the measurements.

minus the fraction expected in an uncorrelated network with the same degree sequence. The link rewiring algorithm is used to increase the link density within the prescribed groups, elevating Q to the desired value.

The simulations indicate that this community structure has no effect on n_{D} (Fig. 3.3), prompting the conclusion that Q does not play a leading role in n_{D}.

It should be noted that adding communities to networks can be achieved in many different ways , e.g. hierarchical organization of communities [94], overlapping community structure [63] or link communities [93]. These scenarios were not investigated, and remain as possible future work.

Degree-degree correlations

In uncorrelated networks, the degrees of two connected nodes do not depend on each other, in real networks, however, this is not true [78, 95, 79, 41]. In social networks, hubs connect to other hubs more the it would be expected by chance, meaning that popular individuals associate with other popular individuals [96]. This manifests in positive correlation between the degree of connected nodes, and such networks are referred to as assortative networks. In most biological and technological networks, hubs avoid connecting to other hubs, that is the correlation between the degree of connected node pairs is negative [96]. Such networks are called disassortative.

This correlation is often quantified by the Pearson correlation coefficient despite a number of disadvantages [85]. In directed networks each node v_i has an in-degree k_i^{in} and an out-degree k_i^{out}, thus four correlation coefficients can be defined: correlations between

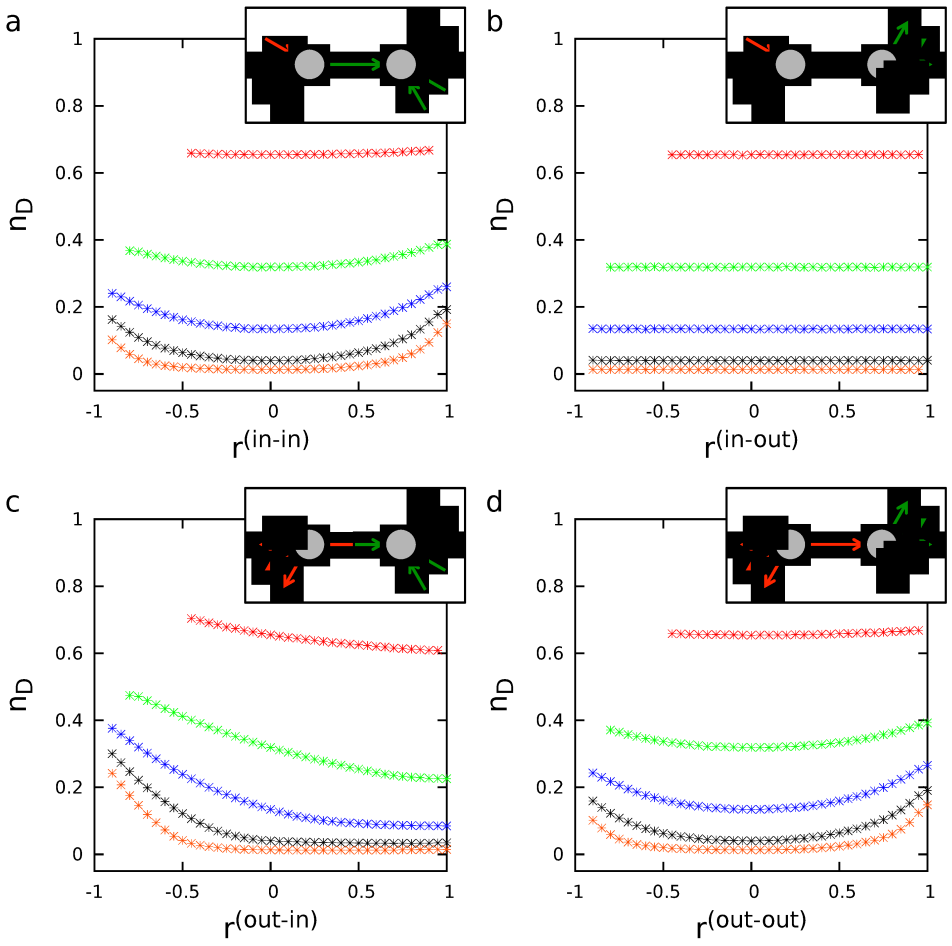


Figure 3.4: **The impact of degree-degree correlations in Erdős-Rényi networks.** The density of driver nodes (n_D) is measured for the Erdős-Rényi model ($N = 10,000$) for average total degrees $\langle k \rangle = 1$ (red), $\langle k \rangle = 3$ (green),$\langle k \rangle = 5$ (blue), $\langle k \rangle = 7$ (black) and $\langle k \rangle = 9$ (orange). The results are similar for the scale-free model (see Fig. 3.5). Each data point is an average of 100 independent runs.

the source node's in- and out-degree, and the target node's in- and out-degree (Figs. 3.4, 3.5) [83]. The Pearson coefficient can be calculated as

$$r^{a-b} = \frac{\frac{1}{E}\sum_e \left(k_e^a - \overline{k^a}\right)\left(j_e^b - \overline{j^b}\right)}{\sigma^a \sigma^b}, \tag{3.4}$$

where $\sum_e \cdot$ sums over all links, $a, b \in \{\text{in, out}\}$ is the degree type, k_e^a is the degree of the source node of link e, j_e^b is the degree of the target node of link e. And $\overline{j^a} = \frac{1}{E}\sum_e j_e^a$ is the average degree of the nodes at the beginning of each link, $(\sigma^a)^2 = \frac{1}{E}\sum_e \left(k_e^a - \overline{k^a}\right)^2$ is the variance; $\overline{k^b}$ and σ^b are defined similarly.

Simulations shown in Figs. 3.4 and 3.5 indicate that degree correlations systematically affect n_D. Three distinct types of behavior are observed:

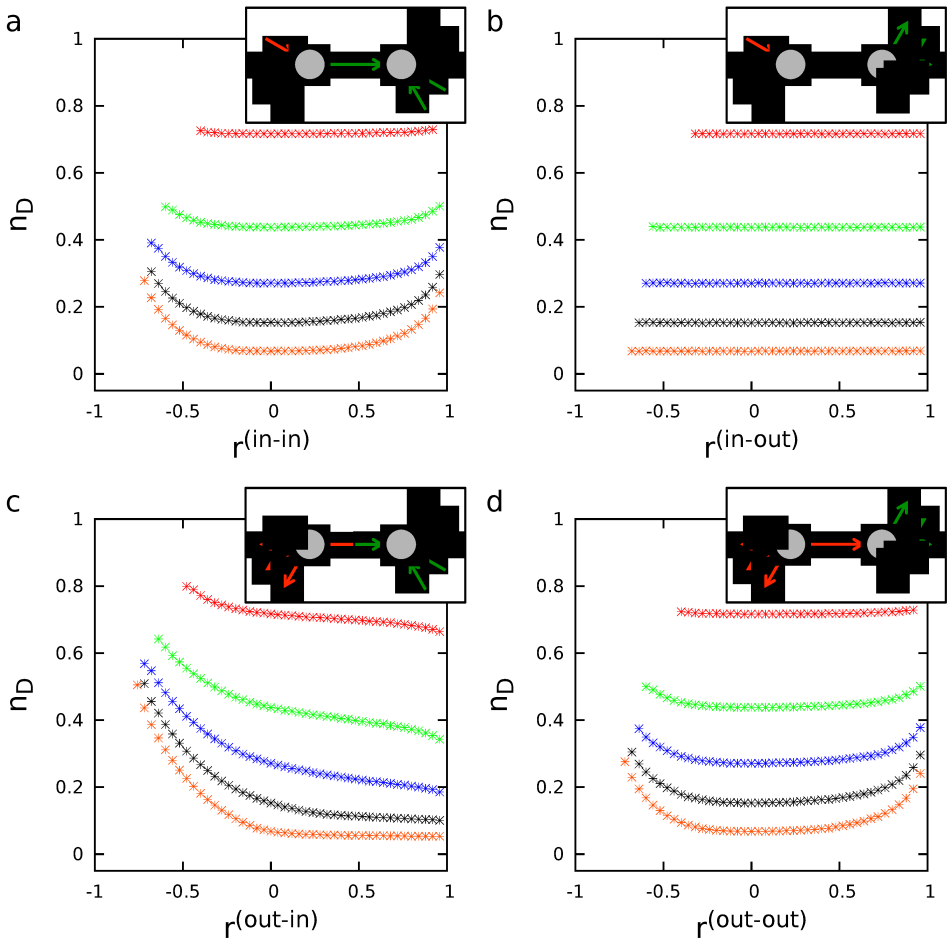


Figure 3.5: **The impact of degree-degree correlations in scale-free networks.** The density of driver nodes (n_D) is measured for the scale-free model ($N = 10,000$, $\gamma = 2.5$) for average total degrees $\langle k \rangle = 1$ (red), $\langle k \rangle = 3$ (green),$\langle k \rangle = 5$ (blue), $\langle k \rangle = 7$ (black) and $\langle k \rangle = 9$ (orange). The results are similar for the Erdős-Rényi model (see Fig. 3.4). Each data point is an average of 100 independent runs.

(i) n_D depends monotonically on $r^{(\text{out-in})}$, so that low (negative) correlations increase n_D and high (positive) correlations lower n_D (Figs. 3.4c, 3.5c);

(ii) Both $r^{(\text{in-in})}$ and $r^{(\text{out-out})}$ increase n_D, independent of the sign of the correlations (Figs. 3.4a, 3.4d, 3.5a, 3.5d);

(iii) $r^{(\text{in-out})}$ has no effect on n_D (Figs. 3.4c, 3.5c).

The behavior is qualitatively the same for Erdős-Rényi (Fig. 3.4) and scale-free (Fig. 3.5) networks.

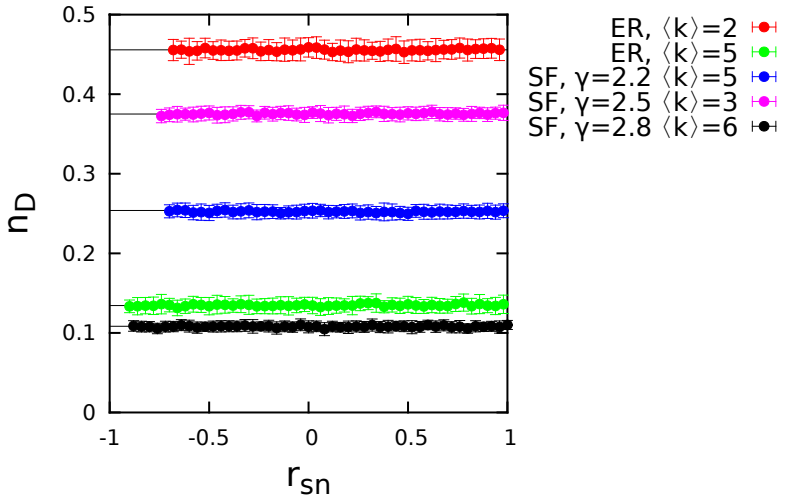


Figure 3.6: **Effect of r_{sn} single node degree correlation on the density of driver nodes n_{D}.** Network size is $N = 1,000$, $\langle k \rangle = \langle k^{\mathrm{in}} \rangle + \langle k^{\mathrm{out}} \rangle$ is the average total degree. Each data point is an average over 50 independent runs; the errors are estimated by the standard deviation of the measurements.

Single node degree correlation

Single node degree correlation describes the correlations of the in-degree k_i^{in} and the out-degree k_i^{out} of a single node v_i. In real networks, nodes with high in-degree tend to have high out-degree too. This property affects dynamical processes on the network, e.g. formation of giant components [97]. It is also known to be important feature that influences structural controllability of switchboard dynamics [48], for brief discussion see Sec. 1.6.

Single node degree correlation is an exception in the study, because it is fully captured by the joint degree distribution $P(k^{\mathrm{in}}, k^{\mathrm{out}})$. However, the model networks used here do not contain single node degree correlation, that is $P(k^{\mathrm{in}}, k^{\mathrm{out}}) \equiv P(k^{\mathrm{in}})P(k^{\mathrm{out}})$.

The correlations between the in- and out-degree of a single node can be quantified by the coefficient

$$r_{\mathrm{sn}} = \frac{\frac{1}{N}\sum_{i=1}^{N}\left(k_i^{\mathrm{in}} - \overline{k^{\mathrm{in}}}\right)\left(k_i^{\mathrm{out}} - \overline{k^{\mathrm{out}}}\right)}{\sigma^{\mathrm{in}}\sigma^{\mathrm{out}}}, \tag{3.5}$$

where $\sum_v \cdot$ sums over all nodes, and $\overline{k} = \frac{1}{N}\sum_{i=1}^{N} k_i$ and $\sigma = \frac{1}{N}\sum_v \left(k_v - \overline{k}\right)^2$.

Degree preserving link rewiring described earlier does not change r_{sn}, so instead of rewiring randomly selected links, in each step two random nodes are chosen, and the incoming links are swapped.

Simulations indicate that r_{sn} has absolutely no impact on n_{D} (Fig. 3.6). This is simply understood if the reader recalls that the driver nodes are the unmatched nodes in the

bipartite representation of the directed network. Single node degree correlation in the bipartite representation is the correlation between the degree of nodes v_i^+ and v_i^-. This correlation can be eliminated by shuffling the labels of the nodes. On the other hand, the size of the maximum matching – or equivalently n_D – cannot depend on the choice of the labels. Therefore single node degree correlation cannot have any effect on n_D.

Self-loop density

The fact that self-loops can have an effect on controllabililty has been demonstrated by Liu et al. [26]. In the first and second example of Sec. 1.3, it was shown that adding a self-loop to an end node of a directed star makes the system controllable. It is also obvious that in the limit that the weight of the self-loop $a_{33} \to 0$, the controllability matrix becomes almost rank deficient and the time constant of node-3 becomes too large to be relevant.

It was suggested by Cowan et al. that many real-world systems represented by networks have intrinsic dynamics, i.e. a node's state x_i will vary over time even in the absence of influence from other nodes [46]. Thus if one wants to model the dynamics of these systems in a network context, self-links are required, even if they were not made explicit in the original network description. This is a modeling or time-scale issue. For example in electronics, the modeling distinction between integrator, leaky integrator, and low-pass filter is based on similar numerical considerations. Incidentally, the entire field of singular perturbations is motivated by similar time-scale issues in modeling. Furthermore, if we add a self-link to *each* node, then the framework based on structural controllability will predict $N_D = 1$, i.e. a single input can be used to control the whole network, irrespective of the network topology and even the network size [26]. This result is correct in principle but not meaningful in reality.

To explore the role of self-loops, the density of directed self-loops is defined as

$$\rho_{sl} = \frac{\text{number of self-loops}}{N}. \tag{3.6}$$

This way the measure will be 0 if no self-loops are present, and 1 if every node is connected to itself.

The simulations indicate (Fig. 3.7) that only extremely overrepresented self-loops decrease n_D. A self-loop can only effect the controllability if it is attached to a node that in some control configuration is a driver node. Liu et al. showed that typically the low degree nodes have to be controlled, not the hubs [26]. If a self-loop is added by random rewiring to a network, it is more likely to belong to a hub, simply because a hub has more possibilities to create one. So, as the number of self-loops increases, first the hubs gain

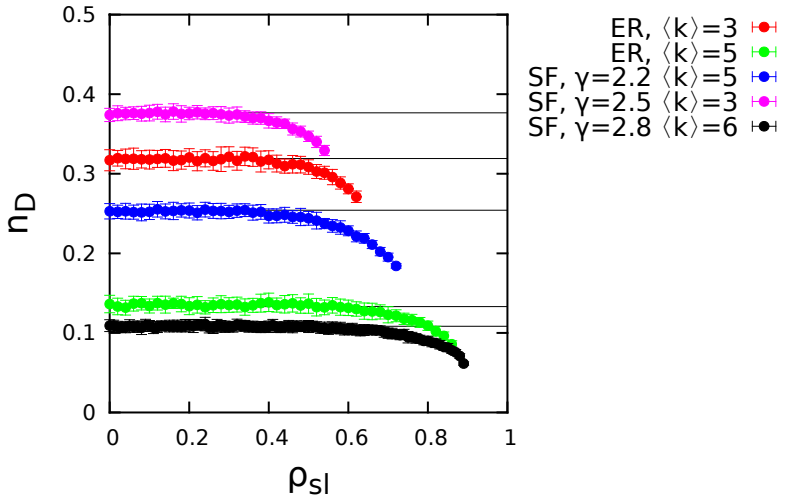


Figure 3.7: Effect of ρ_{sl} self-loop density on the density of driver nodes, n_{D}. Network size is $N = 1,000$, $\langle k \rangle = \langle k^{\mathrm{in}} \rangle + \langle k^{\mathrm{out}} \rangle$ is the average total degree. Each data point is an average over 50 independent runs; the errors are estimated by the standard deviation of the measurements.

self-connections, and in only the extreme case will the low-degree driver nodes. However, this also indicates that targeted adding of self-loops to low-degree nodes could lead to a more abrupt fall in n_{D}.

3.1.2 Analytical solution for networks with degree correlations

Numerical simulations indicated that degree-degree correlations between connected node pairs are the most important network characteristics determining n_{D} apart from the degree distribution. The goal of this section is to understand the origin of the behavior observed on Figs. 3.4 and 3.5.

As described in Sec. 1.4, the task of identifying the driver nodes can be mapped to the problem of finding a maximum matching of the network. A matching is a subset of links that do not share start or end points. A node is called matched if a link in the matching points at it, and a network is fully controlled if the unmatched nodes are the driver nodes. The generating function formalism has been successfully used to calculate the size of the maximum matching for undirected and directed network ensembles with given degree distribution [68, 26]. Here, a solution is derived for network ensembles with given degree correlation profile. The solution relies on the framework developed in Chapter 2, therefore the same notations are used: $p_{\mathrm{in}}(k) \equiv p^{-}(k)$ and $p_{\mathrm{out}}(k) \equiv p^{+}(k)$.

The fraction of driver nodes n_{D} is analytically calculated for a given $p^{-}(k)$, $p^{+}(k)$ and degree-degree correlation $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$, representing the probability of a directed link

pointing from a node with in-degree j_i and out-degree j_o to a node with in-degree k_i and out-degree k_o. In the absence of degree correlations (neutral case)

$$e_0(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = p^-(j_\mathrm{i})q^+(j_\mathrm{o})q^-(k_\mathrm{i})p^+(k_\mathrm{o}), \tag{3.7}$$

where $q^+(j_\mathrm{o}) = (j_\mathrm{o}/\langle j_\mathrm{o}\rangle)\, p^+(j_\mathrm{o})$ and $q^-(k_\mathrm{i}) = (k_\mathrm{i}/\langle k_\mathrm{i}\rangle)\, p^-(k_\mathrm{i})$, where $\left\langle k_{\mathrm{i/o}}\right\rangle$ is the average degree. The following forms of $e_0(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o})$ are used to ensure analytical tractability [79]

$$e_{\mathrm{ii}}(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = q^+(j_\mathrm{o})p^+(k_\mathrm{o})\left[p^-(j_\mathrm{i})q^-(k_\mathrm{i}) + r_{\mathrm{ii}}m_{\mathrm{ii}}(j_\mathrm{i}, k_\mathrm{i})\right], \tag{3.8a}$$

$$e_{\mathrm{io}}(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = q^+(j_\mathrm{o})q^-(k_\mathrm{i})\left[p^-(j_\mathrm{i})p^+(k_\mathrm{o}) + r_{\mathrm{io}}m_{\mathrm{io}}(j_\mathrm{i}, k_\mathrm{o})\right], \tag{3.8b}$$

$$e_{\mathrm{oi}}(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = p^-(j_\mathrm{i})p^+(k_\mathrm{o})\left[q^+(j_\mathrm{o})q^-(k_\mathrm{i}) + r_{\mathrm{oi}}m_{\mathrm{oi}}(j_\mathrm{o}, k_\mathrm{i})\right], \tag{3.8c}$$

$$e_{\mathrm{oo}}(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = p^-(j_\mathrm{i})q^-(k_\mathrm{i})\left[q^+(j_\mathrm{o})p^+(k_\mathrm{o}) + r_{\mathrm{oo}}m_{\mathrm{oo}}(j_\mathrm{o}, k_\mathrm{o})\right]. \tag{3.8d}$$

Each choice ensures a network that has only one type of correlation out of the possible four, and the corresponding Pearson correlation coefficient is equal to r_{ab}. This way the role of each correlation can be studied independently.

By fixing $m_{ab}(j,k)$ $(a, b \in \{\mathrm{in}, \mathrm{out}\})$ a one parameter network ensemble is obtained characterized by r_{ab}, where $m_{ab}(j,k)$ satisfies the constraints

$$\sum_{j=0}^{\infty} m_{ab}(j,k) = \sum_{k=0}^{\infty} m_{ab}(j,k) = 0, \tag{3.9}$$

$$\sigma_a\sigma_b \sum_{j,k=0}^{\infty} jk \cdot m_{ab}(j,k) = 1, \tag{3.10}$$

and all elements of $e_{ab}(j,k)$ are between 0 and 1.

It was demonstrated in Sec. 2.3 that the analytical solution describing core percolation in bipartite networks can be used to calculate the size of maximum matching in a directed network and therefore n_D can also be obtained. The bipartite representation $\mathcal{B}$ of a directed network $\mathcal{G}$ is constructed as follows. For each node v, two nodes v^- and v^+ are created, and a new undirected link $(v^- - w^+)$ is added to $\mathcal{B}$ for each directed link $(v \to w)$ in $\mathcal{G}$. The maximum matching of $\mathcal{B}$ is constructed by applying the greedy leaf removal procedure to the network.

In uncorrelated networks, α^- represents the probability that a node w^- found at the end of a randomly selected link $(v^+ - w^-)$ is α-removable in the absence of link $(v^+ - w^-)$, and the probability α does not depend on the degree of node v. This is not true in general in correlated networks, therefore $\alpha^-(k_\mathrm{i}, k_\mathrm{o})$ denotes the probability that an α-removable node w^- is found at the end of a link originating from a node with degree k_o (which is

the out-degree of node v in the original directed network) and its pair v^- has degree k_i (which is the in-degree of node v in the original directed network). Probability $\alpha^-(k_\mathrm{i}, k_\mathrm{o})$ and $\beta^\pm(k_\mathrm{i}, k_\mathrm{o})$ is defined similarly.

In the bipartite representation of the correlated network model the same local rules hold as in uncorrelated networks:

(i) α-removable: all neighbors are β-removable;

(ii) β-removable: at least one neighbor is α-removable.

Averaging over the network ensemble a set of equations are obtained similar to Eq. (2.23)

$$\begin{aligned}
\alpha^-(j_\mathrm{i}, j_\mathrm{o}) &= \left\{ 1 - \sum_{l_\mathrm{i}, l_\mathrm{o}=0}^{\infty} p_{\rightarrow}(l_\mathrm{i}, l_\mathrm{o}|j_\mathrm{i}, j_\mathrm{o}) \left[1 - \sum_{m_\mathrm{i}, m_\mathrm{o}=0}^{\infty} p_{\leftarrow}(m_\mathrm{i}, m_\mathrm{o}|l_\mathrm{i}, l_\mathrm{o}) \alpha^-(m_\mathrm{i}, m_\mathrm{o}) \right]^{l_\mathrm{i}-1} \right\}^{j_\mathrm{o}-1}, \\
\beta^-(j_\mathrm{i}, j_\mathrm{o}) &= 1 - \left\{ 1 - \sum_{l_\mathrm{i}, l_\mathrm{o}=0}^{\infty} p_{\rightarrow}(l_\mathrm{i}, l_\mathrm{o}|j_\mathrm{i}, j_\mathrm{o}) \left[\sum_{m_\mathrm{i}, m_\mathrm{o}=0}^{\infty} p_{\leftarrow}(m_\mathrm{i}, m_\mathrm{o}|l_\mathrm{i}, l_\mathrm{o}) \beta^-(m_\mathrm{i}, m_\mathrm{o}) \right]^{l_\mathrm{i}-1} \right\}^{j_\mathrm{o}-1}.
\end{aligned} \tag{3.11}$$

Similarly for α^- and β^-

$$\begin{aligned}
\alpha^+(k_\mathrm{i}, k_\mathrm{o}) &= \left\{ 1 - \sum_{l_\mathrm{i}, l_\mathrm{o}=0}^{\infty} p_{\leftarrow}(l_\mathrm{i}, l_\mathrm{o}|k_\mathrm{i}, k_\mathrm{o}) \left[1 - \sum_{m_\mathrm{i}, m_\mathrm{o}=0}^{\infty} p_{\rightarrow}(m_\mathrm{i}, m_\mathrm{o}|l_\mathrm{i}, l_\mathrm{o}) \alpha^+(m_\mathrm{i}, m_\mathrm{o}) \right]^{l_\mathrm{o}-1} \right\}^{k_\mathrm{i}-1}, \\
\beta^+(k_\mathrm{i}, k_\mathrm{o}) &= 1 - \left\{ 1 - \sum_{l_\mathrm{i}, l_\mathrm{o}=0}^{\infty} p_{\leftarrow}(l_\mathrm{i}, l_\mathrm{o}|k_\mathrm{i}, k_\mathrm{o}) \left[\sum_{m_\mathrm{i}, m_\mathrm{o}=0}^{\infty} p_{\rightarrow}(m_\mathrm{i}, m_\mathrm{o}|l_\mathrm{i}, l_\mathrm{o}) \beta^+(m_\mathrm{i}, m_\mathrm{o}) \right]^{l_\mathrm{o}-1} \right\}^{k_\mathrm{i}-1},
\end{aligned} \tag{3.12}$$

where

$$p_{\rightarrow}(l_\mathrm{i}, l_\mathrm{o}|j_\mathrm{i}, j_\mathrm{o}) = \frac{e(j_\mathrm{i}, j_\mathrm{o}; l_\mathrm{i}, l_\mathrm{o})}{p^-(j_\mathrm{i}) q^+(j_\mathrm{o})} \tag{3.13}$$

gives the probability of finding a node with degree l_i and l_o following a link from a node with degree j_i and j_o. And

$$p_{\leftarrow}(l_\mathrm{i}, l_\mathrm{o}|k_\mathrm{i}, k_\mathrm{o}) = \frac{e(l_\mathrm{i}, l_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o})}{q^-(k_\mathrm{i}) p^+(k_\mathrm{o})}, \tag{3.14}$$

is defined similarly only following a link backwards.

Similarly to Eq. (2.36), the fraction of driver nodes is

$$
\begin{aligned}
n_{\mathrm{D}} = & \frac{1}{2} \sum_{j_{\mathrm{i}}, j_{\mathrm{o}}=0}^{\infty} p^{+}(j_{\mathrm{o}}) p^{-}(j_{\mathrm{i}}) \left[\left(1 - \sum_{l_{\mathrm{i}}, l_{\mathrm{o}}=0}^{\infty} p_{\rightarrow}(l_{\mathrm{i}}, l_{\mathrm{o}} | j_{\mathrm{i}}, j_{\mathrm{o}}) \alpha^{+}(l_{\mathrm{i}}, l_{\mathrm{o}}) \right)^{j_{\mathrm{o}}} + \right. \\
& \left. + \left(\sum_{l_{\mathrm{i}}, l_{\mathrm{o}}=0}^{\infty} p_{\rightarrow}(l_{\mathrm{i}}, l_{\mathrm{o}} | j_{\mathrm{i}}, j_{\mathrm{o}}) \beta^{+}(l_{\mathrm{i}}, l_{\mathrm{o}}) \right)^{j_{\mathrm{o}}} - 1 \right] + \\
& + \frac{1}{2} \sum_{k_{\mathrm{i}}, k_{\mathrm{o}}=0}^{\infty} p^{+}(k_{\mathrm{o}}) p^{-}(k_{\mathrm{i}}) \left[\left(1 - \sum_{l_{\mathrm{i}}, l_{\mathrm{o}}=0}^{\infty} p_{\leftarrow}(l_{\mathrm{i}}, l_{\mathrm{o}} | k_{\mathrm{i}}, k_{\mathrm{o}}) \alpha^{-}(l_{\mathrm{i}}, l_{\mathrm{o}}) \right)^{k_{\mathrm{i}}} + \right. \\
& \left. + \left(\sum_{l_{\mathrm{i}}, l_{\mathrm{o}}=0}^{\infty} p_{\leftarrow}(l_{\mathrm{i}}, l_{\mathrm{o}} | k_{\mathrm{i}}, k_{\mathrm{o}}) \beta^{-}(l_{\mathrm{i}}, l_{\mathrm{o}}) \right)^{k_{\mathrm{i}}} - 1 \right] + \\
& + \frac{\langle k \rangle}{2} \sum_{j_{\mathrm{i}}, j_{\mathrm{o}}, k_{\mathrm{i}}, k_{\mathrm{o}}=0}^{\infty} e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}}) \left[\alpha^{-}(j_{\mathrm{i}}, j_{\mathrm{o}}) \left(1 - \beta^{+}(k_{\mathrm{i}}, k_{\mathrm{o}}) \right) + \left(1 - \beta^{-}(j_{\mathrm{i}}, j_{\mathrm{o}}) \right) \alpha^{+}(k_{\mathrm{i}}, k_{\mathrm{o}}) \right] .
\end{aligned}
\tag{3.15}
$$

Equipped with the equations governing $\alpha^{\pm}(k_{\mathrm{i}}, k_{\mathrm{o}})$ and $\beta^{\pm}(k_{\mathrm{i}}, k_{\mathrm{o}})$, each type of correlation is investigated one by one.

Neutral

If there are no correlations in the network, $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}}) = e^{(0)}(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$ given by Eq. (3.7).

The following averages are defined:

$$
\begin{aligned}
\alpha^{-} &= \sum_{j_{\mathrm{i}}, j_{\mathrm{o}}=0}^{\infty} p^{-}(j_{\mathrm{i}}) q^{+}(j_{\mathrm{o}}) \alpha^{-}(j_{\mathrm{i}}, j_{\mathrm{o}}), \\
\alpha^{+} &= \sum_{k_{\mathrm{i}}, k_{\mathrm{o}}=0}^{\infty} q^{-}(k_{\mathrm{i}}) p^{+}(k_{\mathrm{o}}) \alpha^{+}(k_{\mathrm{i}}, k_{\mathrm{o}}),
\end{aligned}
\tag{3.16}
$$

and $\beta^{\pm}$ are defined similarly. Combining this with Eqs. (3.11) and (3.12), we recover Eq. (2.23) derived in Chapter 2:

$$
\begin{aligned}
\alpha^{\pm} &= H^{\pm}(\beta^{\mp}), \\
\beta^{\pm} &= 1 - H^{\pm}(1 - \alpha^{\mp}),
\end{aligned}
\tag{3.17}
$$

where $H^{\pm}(x) = \sum_{k=0}^{\infty} q^{\pm}(k+1) x^{k}$ is the usual generating function.

Now using Eq. (3.15), we recover Eq. (2.36):

$$n_{\mathrm{D}}^{(0)} = \frac{1}{2}\left\{\left[G^{+}(\beta^{-}) + G^{+}(1-\alpha^{-}) - 1\right] + \left[G^{-}(\beta^{+}) + G^{-}(1-\alpha^{+}) - 1\right]\right\} + \\ + \frac{\langle k\rangle}{2}\left[\alpha^{-}(1-\beta^{+}) + \alpha^{+}(1-\beta^{-})\right], \tag{3.18}$$

where $G^{\pm}(x) = \sum_{k=0}^{\infty} p^{\pm}(k)x^{k}$ are the generating functions of the in- and out-degree distribution.

Out-in correlations

For out-in correlations, $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}}) = e_{\mathrm{oi}}(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$ is given by Eq. (3.8c). The correlations are treated as perturbations to the neutral case, that is r_{oi} is assumed to be small. Using Eqs. (3.11) and (3.12) and keeping the first nonzero correction the following solutions are obtained:

$$\alpha^{\pm}(j_{\mathrm{i}}, j_{\mathrm{o}}) = (\beta^{\mp})^{j_{\mathrm{o}}-1} - r_{\mathrm{oi}} \cdot (j_{\mathrm{o}}-1)(\beta^{\mp})^{j_{\mathrm{o}}-2} \sum_{l_{\mathrm{i}}=0}^{\infty} \frac{m_{\mathrm{oi}}(j_{\mathrm{o}}, l_{\mathrm{i}})}{q^{\mp}(j_{\mathrm{o}})}(1-\alpha^{\pm})^{l_{\mathrm{i}}-1} \\ \beta^{\pm}(j_{\mathrm{i}}, j_{\mathrm{o}}) = 1 - (1-\alpha^{\mp})^{j_{\mathrm{o}}-1} + r_{\mathrm{oi}} \cdot (j_{\mathrm{o}}-1)(1-\alpha^{\mp})^{j_{\mathrm{o}}-2} \sum_{l_{\mathrm{i}}=0}^{\infty} \frac{m_{\mathrm{oi}}(j_{\mathrm{o}}, l_{\mathrm{i}})}{q^{\mp}(j_{\mathrm{o}})}(\beta^{\pm})^{l_{\mathrm{i}}-1}, \tag{3.19}$$

where $\alpha^{\pm}$ and $\beta^{\pm}$ are the solutions for the neutral case. Using Eq. (3.15) we find that the first nonzero correction is of first order

$$n_{\mathrm{D}}^{\mathrm{oi}} = n_{\mathrm{D}}^{(0)} - r_{\mathrm{oi}}\frac{\langle k\rangle}{2}\left[M_1(\beta^{+}, 1-\alpha^{-}) + M_1(1-\alpha^{+}, \beta^{-})\right], \tag{3.20}$$

where

$$M_1(x, y) = \sum_{j,k=1}^{\infty} m_{\mathrm{oi}}(j,k)x^{j-1}y^{k-1}. \tag{3.21}$$

Note that the slope of the correction depends on the details of $m_{\mathrm{oi}}(j,k)$, and therefore on the specific choice of $e_{\mathrm{oi}}(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$.

Equation (3.20) predicts that n_{D} depends linearly on r_{oi}, a prediction supported by simulations for small r^{oi} (Figs. 3.4c and 3.5c). This behavior can be intuitively understood by recalling that driver nodes are the unmatched nodes in the maximum matching. For a node v with out-degree k_{o}, by definition only one link can be in the matching. If the remainder $k_{\mathrm{o}}-1$ links point to nodes with degree 1 (disassortative case), v inhibits them from being matched, so we have to control each of them individually, increasing n_{D}. If the remainder $k_{\mathrm{o}}-1$ links point to hubs (assortative case), these hubs are likely to be matched through another incoming link, decreasing n_{D}.

Out-out correlations

For out-in correlations, $e(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = e_{\mathrm{oo}}(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o})$ is given by Eq. (3.8d). The fraction of driver nodes n_D is obtained similarly to the out-in case, only now the first nonzero correction is of second order

$$n_\mathrm{D}^{\mathrm{oo}} = n_\mathrm{D}^{(0)} + r_{\mathrm{oo}}^2 \frac{\langle k \rangle}{4} \left[H^{-\prime}(1-\alpha^-) M_2(\beta^+) + H^{+\prime}(\beta^-) M_2(1-\alpha^+) \right], \tag{3.22}$$

where $H^{\pm\prime}(x)$ is the derivative of $H^{\pm}(x)$ and

$$M_2(x) = \sum_{j,k=1,l=0}^{\infty} \frac{m_{\mathrm{oo}}(l,j) m_{\mathrm{oo}}(l,k)}{p^+(l)} x^{j-1} x^{k-1}. \tag{3.23}$$

To interpret $M_2(x)$, consider the following scenario. Two adjacent links e_1 and e_2 are randomly chosen such that they share endpoints. Let P be the probability that e_1 originates from node with out-degree k_o and e_2 originates to a node with out-degree j_o. This probability P can be calculated

$$\begin{aligned} P = \sum_{l_\mathrm{i}, l_\mathrm{o}, j_\mathrm{i}, k_\mathrm{i}=0}^{\infty} \frac{e_{\mathrm{oo}}(l_\mathrm{i}, l_\mathrm{o}, j_\mathrm{i}, j_\mathrm{o}) e_{\mathrm{oo}}(l_\mathrm{i}, l_\mathrm{o}, k_\mathrm{i}, k_\mathrm{o})}{p^+(l_\mathrm{o}) q^-(l_\mathrm{i})} = \\ = q^+(j_\mathrm{o}) q^+(k_\mathrm{o}) + r_{\mathrm{oo}}^2 \sum_{l_\mathrm{o}=0}^{\infty} \frac{m_{\mathrm{oo}}(l_\mathrm{o}, j_\mathrm{o}) m_{\mathrm{oo}}(l_\mathrm{o}, k_\mathrm{o})}{p^+(l_\mathrm{o})}. \end{aligned} \tag{3.24}$$

One can recognize the second term as the deviation from the neutral case. That is, $n_\mathrm{D}^{\mathrm{oo}}$ does not depend on the out-out correlation of the directly connected nodes, only on the correlation between second neighbors. Positive r_{oo} correlation between the immediate neighbors means that if node v_1 (see Fig. 3.8) has high out-degree, then node v_2 is expected to have high out-degree, and therefore v_3 is likely to have high out-degree as well. Negative r_{oo} means that if node v_1 has high out-degree, node v_2 has low out-degree and v_3 will have high out-degree again. That is, both negative and positive one-step out-out correlations induce positive two-step correlation between v_1 and v_3, accounting for the symmetry of the effect observed in simulations (Figs. 3.4d and 3.5d).

In-in correlations

For out-in correlations, $e(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o}) = e_{\mathrm{ii}}(j_\mathrm{i}, j_\mathrm{o}; k_\mathrm{i}, k_\mathrm{o})$ is given by Eq. (3.8a).

The case of in-in correlations is essentially the same as the out-out case: switching the direction of each link does not change the matching, but turns the out-out into in-in correlations. Therefore $n_\mathrm{D}^{\mathrm{ii}}$ can be obtained simply by exchanging $p^-(k_\mathrm{i})$ and $p^+(k_\mathrm{o})$,

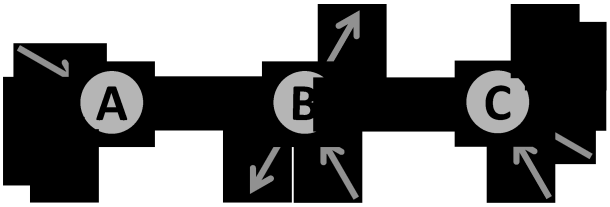


Figure 3.8: **One-step out-out correlations induce positive two-step correlation.** Positive (negative) correlation between neighboring nodes means that if node v_1 has high out-degree, then node v_2 is likely to have high (low) out-degree, and hence v_3 will likely have high out-degree.

indicating that

$$n_{\mathrm{D}}^{\mathrm{ii}} = n_{\mathrm{D}}^{(0)} + r_{\mathrm{ii}}^2 \frac{\langle k \rangle}{4} \left[H^{+\prime}(1-\alpha^+) M_2(\beta^-) + H^{-\prime}(\beta^-) M_2(1-\alpha^+) \right], \tag{3.25}$$

where

$$M_2(x) = \sum_{j,k=1,l=0}^{\infty} \frac{m_{\mathrm{ii}}(l,j) m_{\mathrm{ii}}(l,k)}{p^-(l)} x^{j-1} x^{k-1}, \tag{3.26}$$

predicting again the same dependence on r_{ii} as observed for out-out correlations, supported by the numerical simulations (Figs. 3.4a and 3.5a).

In-out correlations

For out-in correlations, $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}}) = e_{\mathrm{io}}(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$ is given by Eq. (3.8b).

Substituting into Eqs. (3.11) and (3.12) all corrections are found to be zero, meaning

$$n_{\mathrm{D}}^{\mathrm{oi}} = n_{\mathrm{D}}^{(0)}, \tag{3.27}$$

a prediction supported by the simulations (Figs. 3.4b and 3.5b).

Taken together, the functional dependence of n_{D} on degree-degree correlations defines three classes of behaviors, depending on the matching problem's underlying symmetries: (i) n_{D} has no dependence on r_{io}, (ii) linear dependence on r_{oi} and (iii) quadratic dependence on r_{ii} and r_{oo} for small correlation coefficients. These predictions are fully supported by numerical simulations (Figs. 3.4 and 3.5): for small r we see no dependence on $r^{(\text{in-out})}$, an asymmetric, monotonic dependence on $r^{(\text{out-in})}$, and a symmetric on $r^{(\text{in-in})}$ and $r^{(\text{out-out})}$.

To directly compare the analytical predictions to simulations the complete $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$ distribution has to be known, which is not explicitly set in the simulations in Sec. 3.1.1. So to test the results, a different rewiring method is used that sets the $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$ distribution, not only the r correlation coefficient [79]. This method is not as robust as

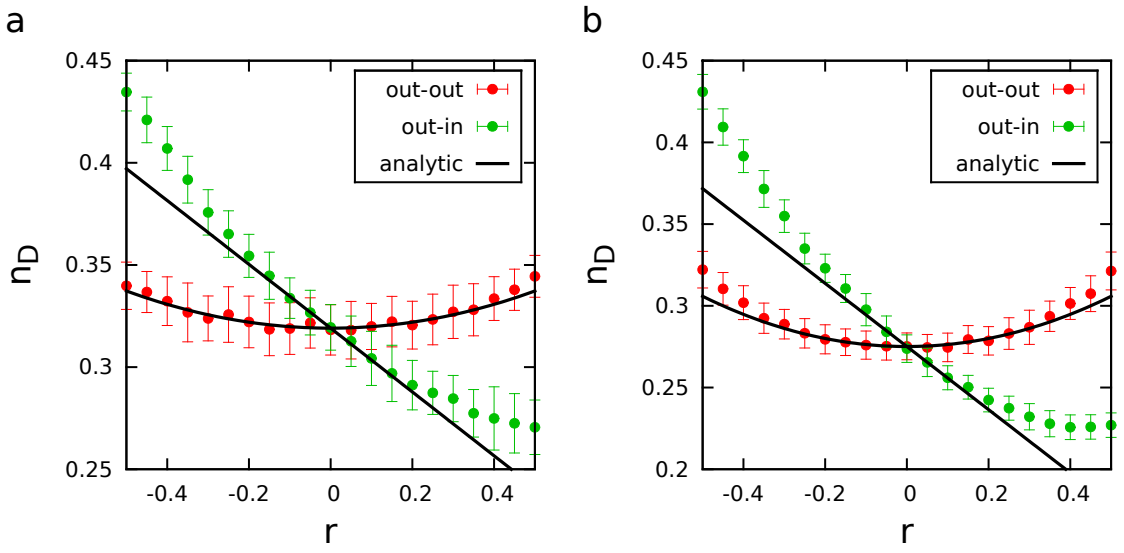


Figure 3.9: **The analytical formulas compared to simulations.** The density of driver nodes n_D is plotted in function of degree-degree correlation coefficient r for **(a)** Erdős-Rényi networks and for **(b)** scale-free networks. The algorithm proposed in Ref. [79] is used to set $e_{ab}(j_i, j_o; k_i, k_o)$. For **(a)** $N = 1,000$ and $\langle k \rangle = 1.5$; for **(b)** $N = 1,000$, $\gamma = 2.5$ and $\langle k \rangle = 2$. Each data point is an average over 100 independent runs; the errors represent by the standard deviation of the measurements.

the original algorithm and the range of accessible r values is more restricted. However, since the results are based on perturbation scheme they are only expected to be correct for small r values. Indeed, it is found that the predictions quantitatively reproduce the numerical results in a fair interval of r_{ab} (Fig. 3.9).

3.1.3 Real networks

The predictions provided by the developed analytical and numerical tools are tested on a set of publicly available network datasets (Table 3.1). To test our predictions on real networks the following measure is calculated

$$\Delta = \frac{N_D - N_D^{rand}}{N}, \tag{3.28}$$

where N_D^{rand} represents the number of driver nodes for the degree-preserved randomized version of the original network. Hence if $\Delta = 0$ then $p_{in}(k_i)$ and $p_{out}(k_o)$ accurately determines N_D; if $\Delta \neq 0$ then the structural properties not captured by the degree sequence influence its controllability. The correlations are measured in several real networks and based on the numerical and analytical results, the sign of Δ is predicted (Fig. 3.10). The networks are grouped according to the predictions.

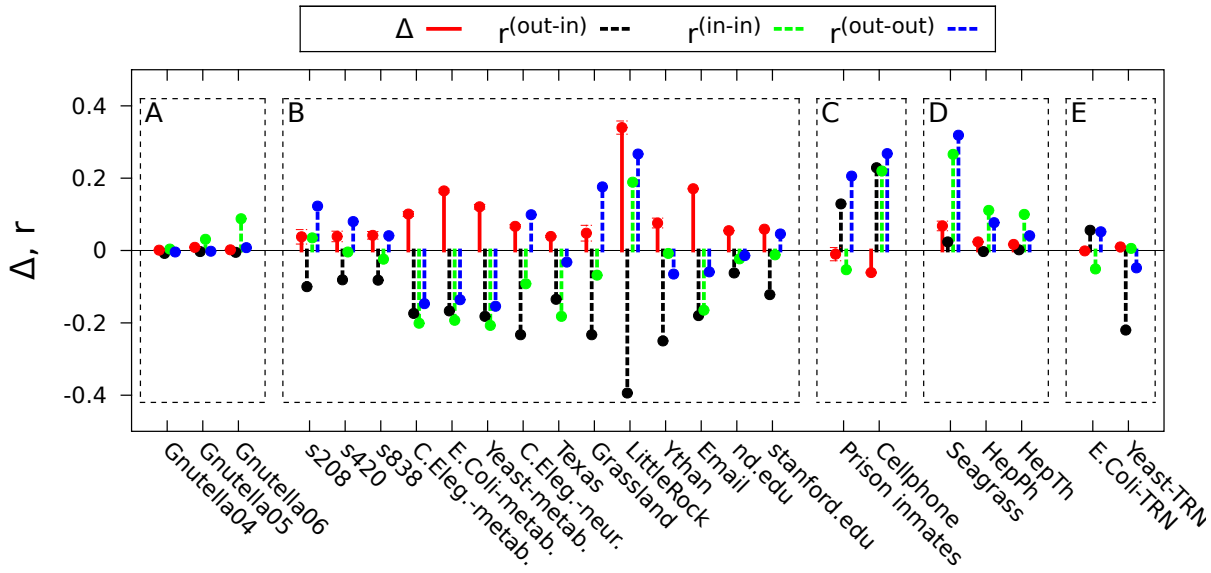


Figure 3.10: **The observed and predicted deviation between N_{D} and $N_{\mathrm{D}}^{\mathrm{rand}}$.** Red line: $\Delta = \left(N_{\mathrm{D}} - N_{\mathrm{D}}^{\mathrm{rand}}\right)/N$, the prediction error based on the degree sequence. Dashed lines: correlations relevant to controllability. For each network Δ is calculated by averaging over 50 independent randomized configurations.

Group A. The networks of p2p Internet (Gnutella filesharing clients) do not have strong correlations, therefore n_{D} is expected to be correctly approximated by the prediction based on $p_{\mathrm{in}}(k_{\mathrm{i}})$ and $p_{\mathrm{out}}(k_{\mathrm{o}})$ (i.e. $\Delta \approx 0$), in line with the empirical observations.

Group B. As in most networks the three relevant correlations coexist to some degree, it is impossible to isolate their individual role. Yet, the networks in this group (electric circuits, metabolic networks, neural networks, power grids and food webs with exception of the Seagrass network) all have negative out-in and nonzero in-in and out-out correlations, each of which individually increase n_{D}. Therefore $\Delta > 0$ is predicted, in line with the empirical observations.

Group C. Only the prison social-trust and the cell phone network feature significant positive out-in correlations. These networks also display nonzero in-in and out-out correlation, leading to the coexistence of two competing effects: out-in correlations decrease n_{D} and the out-out and in-in correlations increase n_{D}. Since the out-in correlation is a first order effect (Eq. (3.20)), while out-out and in-in correlations are only of second order (Eq. (3.22)), a decrease in n_{D} is expected (i.e. $\Delta < 0$), consistent with the empirical results.

Group D. The Seagrass food web and citation networks do not feature significant out-in correlations, only the secondary in-in and out-out correlations, hence n_{D} is expected to increase ($\Delta > 0$), consistent with the observations.

Group E. Only the transcriptional regulatory networks are somewhat puzzling in that they show degree correlations, yet the degree sequence still correctly gives n_{D}. However, the simulations indicated that the effect of correlations is negligible for high n_{D}. And the analytical results showed that the value of the correction depends on details of $e(j_{\mathrm{i}}, j_{\mathrm{o}}; k_{\mathrm{i}}, k_{\mathrm{o}})$, not captured by the Pearson coefficient r. These observations highlight that even though in most cases our qualitative predictions based on r are valid, in some cases further investigation is required.

If these observations would stand on there own, they would not provide enough information to describe the role of correlations. However, they provide auxiliary support for the analytical and numerical results.

Note that when complex systems are mapped to networks, the links connecting the nodes represent interactions between them. In this context self-loops represent self-interactions, with a strong, well understood impact on controllability [26, 46]. The purpose here is to test the effect of correlations, hence we rely on datasets that capture the wiring diagram of various complex systems with different correlation properties. Therefore, even if in some of these maps self-loops are missing, it is beyond the scope of this work to complete these networks. However, when studying controllability of a particular system, careful thought has to be put into whether self-loops are present or not.

Network	N	L	r_{oi}	r_{oo}	r_{ii}	n_{D}	$n_{\mathrm{D}}^{\mathrm{rand}}$	Ref.
Electric circuits								
s208	122	189	−0.100	0.035	0.123	0.238	0.200 ± 0.0200	[98]
s420	252	399	−0.081	−0.004	0.080	0.234	0.195 ± 0.0145	[98]
s838	512	819	−0.082	−0.024	0.041	0.232	0.190 ± 0.0105	[98]
Food webs								
Grassland	88	137	−0.233	−0.068	0.176	0.522	0.474 ± 0.0217	[99]
Little Rock	183	2494	−0.394	0.189	0.267	0.541	0.201 ± 0.0183	[100]
Seagrass	49	226	0.024	0.266	0.319	0.265	0.197 ± 0.0133	[101]
Ythan	135	601	−0.250	−0.008	−0.065	0.511	0.435 ± 0.0135	[99]
Metabolic								
C. Elegans	1,173	2,864	−0.174	−0.201	−0.147	0.302	0.201 ± 0.0066	[102]
E. Coli	2,275	5,763	−0.167	−0.193	−0.136	0.382	0.217 ± 0.0051	[102]
Yeast	1,511	3,833	−0.182	−0.207	−0.154	0.329	0.208 ± 0.0065	[102]
Neural								
C. Elegans	297	2,345	−0.233	−0.092	0.099	0.165	0.098 ± 0.0048	[12]
Social								
Prison inmates	67	182	0.129	−0.053	0.206	0.134	0.144 ± 0.0183	[103, 104]
Cellphone	36,595	91,826	0.229	0.220	0.268	0.151	0.212 ± 0.0012	[105]
Email	57,194	103,731	−0.180	−0.165	−0.059	0.893	0.722 ± 0.0011	[106]
Power grid								
Texas	4,889	5,855	−0.135	−0.182	−0.032	0.325	0.286 ± 0.0024	[107]
Transcriptional regulatory								
E. Coli	418	519	0.056	−0.051	0.052	0.751	0.752 ± 0.0031	[98]
Yeast	688	1,079	−0.220	0.006	−0.048	0.821	0.811 ± 0.0014	[98]
WWW								
Notre Dame	325,729	1,497,134	−0.062	−0.023	−0.014	0.677	0.622 ± 0.0003	[7]
Stanford	281,903	2,312,497	−0.122	−0.012	0.046	0.317	0.258 ± 0.0006	[108]
p2p Internet								
Gnutella-04	10,876	39,994	−0.008	0.004	−0.004	0.552	0.551 ± 0.0006	[109]
Gnutella-05	8,846	31,839	−0.003	0.031	−0.002	0.578	0.569 ± 0.0006	[109]
Gnutella-06	8,717	31,525	−0.005	0.088	0.008	0.577	0.575 ± 0.0006	[109]
Citation								
ArXiv-HepPh	34,546	421,578	−0.003	0.111	0.077	0.232	0.208 ± 0.0008	[110]
ArXiv-HepTh	27,770	352,807	0.002	0.100	0.041	0.216	0.199 ± 0.0009	[110]

Table 3.1: **Real networks.** For each network: number of nodes (N) and links (L); the out-in (r_{oi}), the out-out (r_{oo}) and the in-in (r_{ii}) correlation coefficient; the density of driver nodes (n_{D}); the density of driver nodes after degree-preserving randomization ($n_{\mathrm{D}}^{\mathrm{rand}}$) averaged over 100 independent randomizations, the error is estimated by the standard deviation of the sample; and reference.

3.2 Classification of driver nodes

In this section, the nodes are categorized according to their role in control based on the observation that multiple possible maximum matching configurations can exist. In Sec. 3.2.1, the node categories are defined, and a polynomial time algorithm is described to classify each node in a given directed network. In Sec. 3.2.2 using this algorithm, model networks are systematically studied, revealing that in certain cases the different node categories are distributed very differently even if the networks are generated using the same parameters. More specifically, it is found that above a critical density of links, two control modes emerge: either majority of nodes can be used as driver nodes or driver nodes can only be selected from a small subset of nodes. In Sec. 3.2.3, analytical solution based on the greedy leaf removal unveils the relation between core percolation and the emergence of two control modes. It is shown that the structure of the core determines the control mode of uncorrelated model networks.

3.2.1 Driver node categories

Let us start with the small example on Fig. 3.11. In this small network, the size of the maximum matching is two, and the maximum cardinality matching can be conceived three ways: link $(v_1 \to v_2)$ is always in the matching, and either one of links $(v_2 \to v_3)$, $(v_2 \to v_4)$ and $(v_2 \to v_5)$ is included. According to the minimum input theorem, the driver nodes are the unmatched nodes, therefore three possible minimum driver node sets (MDS) exist. This observation prompts us to categorize the nodes based on the likelihood that they are included in a MDS:

(i) *Critical nodes* are always driver nodes, that is they are in all MDSs or equivalently the are matched in all possible maximum matching sets.

(ii) *Redundant nodes* are the nodes that are never required to be driver nodes, that is they are never included in the MDS or equivalently the are never matched.

(iii) *Intermittent nodes* are the nodes that are in some of the MDSs but not in all. That is intermittent nodes are sometimes matched, sometimes remain unmatched.

For example, on Fig. 3.11, node v_1 is critical, node v_2 is redundant and nodes v_3, v_4 and v_5 are intermittent.

The next question is how to determine the category of a node v in a directed network $\mathcal{G}$? It is proven in the Supplementary Information of Ref. [21] that a node is critical if it has no incoming links. Here, an algorithm is provided to identify the redundant nodes. The nodes that are nor critical, nor redundant are necessarily intermittent.

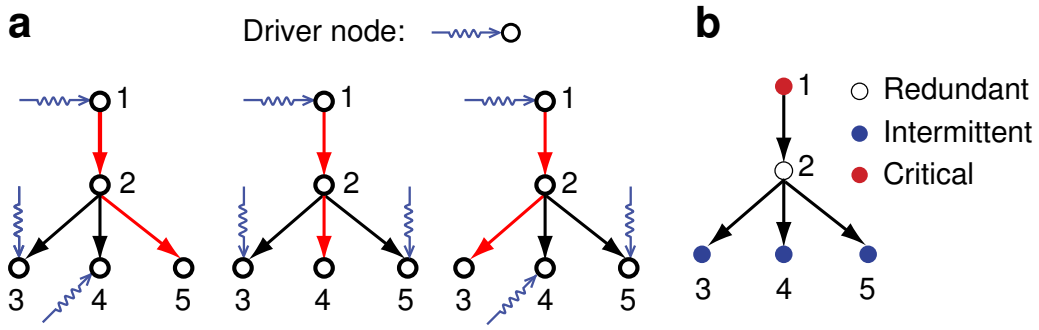


Figure 3.11: **Possible maximum matchings in a network. (a)** A network with five nodes with maximum matching size two. There are three possible maximum matching configurations (red links), therefore there are three possible MDSs. **(b)** Node v_1 is critical as it is part of all MDSs. Node v_2 is redundant as it does not participate in any MDSs. And nodes v_3, v_4 and v_5 are intermittent, beacause they are included in some but not all MDSs.

To determine if node v is redundant, the bipartite representation $\mathcal{B}$ of directed network $\mathcal{G}$ is used. A redundant node v is always matched in $\mathcal{G}$, that is v^- is always matched in $\mathcal{B}$. Therefore if v^- is forced to be unmatched, the cardinality of the maximum matching decreases by one. In the first step, a possible maximum matching in $\mathcal{B}$ is identified using the Hopcroft-Karp algorithm [34]. Then node v^- is removed (v^+ is not removed), and the Hopcroft-Karp algorithm is used again to calculate the size of the maximum matching in $\mathcal{B} \setminus v^-$. If the size of the maximum matching decreased, v is redundant. This is repeated for all nodes $v^- \in V^-$.

The complexity of the Hopcroft-Karp algorithm is $O(N^{1/2}L)$, where N is the number of nodes and L is the number of links [34, 35]. Therefore naïv implementation of the algorithm has complexity $O(N^{3/2}L)$. However, it is possible to update a maximum matching configuration after removing a node in $O(L)$ steps using augmenting paths [34, 35], making it possible to find all redundant nodes in $O(NL)$ steps.

3.2.2 Emergence of bimodality

To explore the role of network topology, Poisson distributed networks and scale-free networks generated by the static model are systematically studied (for model definitions see Sec. 2.1). The fraction of redundant nodes n_r, the fraction of critical nodes n_c and the fraction of intermittent nodes n_i are measured for different average degree c.

Critical nodes are nodes that do not have incoming links, that is $n_\mathrm{c} = p_\mathrm{in}(k = 0)$. This prediction is fully supported by the numerical simulations and nothing unexpected happens (Fig. 3.12b).

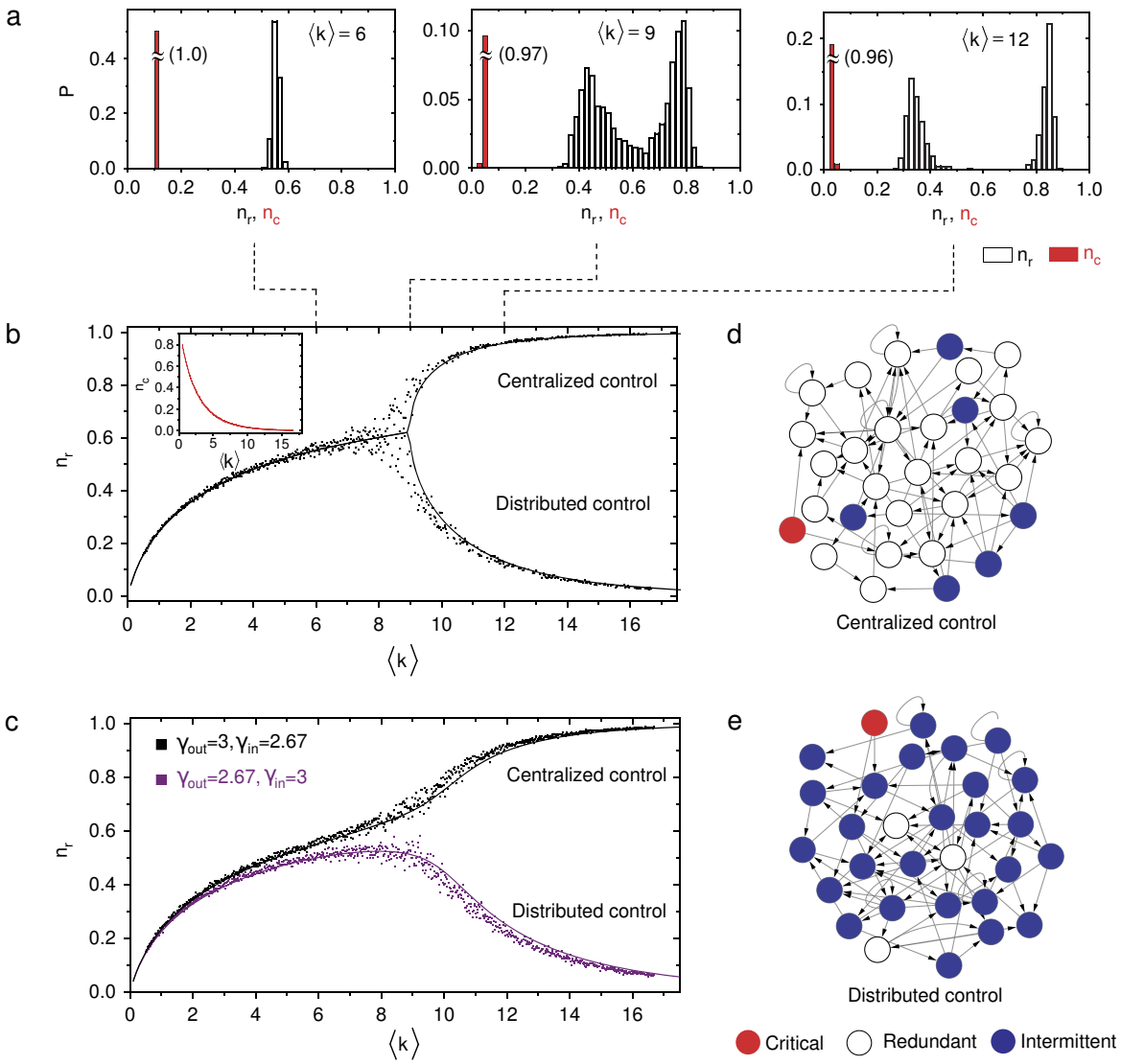


Figure 3.12: **Emergence of bimodality. (a)** Distribution of the fraction of redundant ($n_{\rm r}$) and critical ($n_{\rm c}$) nodes for scale-free networks with $\gamma_{\rm out} = \gamma_{\rm in} = 3$, documenting the emergence of bimodal behavior for high $\langle k \rangle$. **(b)** $n_{\rm r}$ and $n_{\rm c}$ (inset) in function of $\langle k \rangle$ for scale-free networks with $\gamma_{\rm out} = \gamma_{\rm in} = 3$, illustrating the emergence of the two control modes for high $\langle k \rangle$. **(c)** $n_{\rm r}$ and $n_{\rm c}$ (inset) in function of $\langle k \rangle$ for scale-free networks either with $\gamma_{\rm out} = 3$ and $\gamma_{\rm in} = 2.67$ or with $\gamma_{\rm out} = 2.67$ and $\gamma_{\rm in} = 3$, showing that the asymmetry of the degree distributions forces the network in one of the control modes. The continuous line in **b** and **c** correspond to the analytic solution provided by Eq. (3.31), and each data point corresponds to one independent measurement. **(d,e)** Networks displaying centralized and distributed control mode. For both networks $N_{\rm D} = 4$ and $N_{\rm C} = 1$, yet they have very different number of redundant nodes $N_{\rm r} = 23$ in **d** and $N_{\rm r} = 3$ in **e**. Note that $\langle k \rangle = \langle k_{\rm in} \rangle + \langle k_{\rm in} \rangle = 2c$ is the total average degree.

The fraction of redundant nodes n_r exhibits more interesting behavior. For small c average degree, n_r monotonically increases as c increases for both Poisson distributed and scale-free networks with identical in- and out-degree distributions ($p_{in}(k) \equiv p_{out}(k)$). Surprisingly, when c exceeds a critical value c^*, half the time n_r further increases and half the time n_r decreases as c increases. That is plotting n_r versus c, two distinct branches appear for $c > c^*$ (Fig. 3.12b). To better illustrate this, n_r is measured for a large number of independent realizations of networks generated with the same c, and the distribution $P(n_r)$ is calculated. It was found that for $c < c^*$, $P(n_r)$ is unimodal and it is sharply peaked around its average value. And above the critical point $c > c^*$, $P(n_r)$ becomes bimodal, sharply peaked around the average value of the two branches of n_r (Fig.3.12a). Numerical investigations indicate that c^* coincides with the critical point of core percolation.

The two modes of the bimodal distribution are strikingly different:

(i) *Centralized control:* For networks that are on the upper branch of the diagram most nodes are redundant. This means that the driver nodes are selected from an exceedingly small fraction of nodes n_c+n_i. Hence, this scenario is dubbed centralized control mode. For an example see Fig. 3.12d.

(ii) *Distributed control:* On the lower branch, most nodes are either critical or intermittent, meaning that most nodes can act as driver nodes in some MDSs. For an example see Fig. 3.12e.

For networks with different in- and out-degree distributions ($p_{in}(k) \neq p_{out}(k)$) the two branches no longer coexist, either one or the other branch is populated. Flipping the direction of the links, that is exchanging the degree distributions $p_{in}(k_i)$ and $p_{out}(k_o)$, changes the mode of the network. This is illustrated on Fig. 3.12c for networks with $p_{in}(k_i) \sim k_i^{-\gamma_{in}}$ and $p_{out}(k_o) \sim k_o^{-\gamma_{out}}$ for large k_i and k_o, where $\gamma_{in} \neq \gamma_{out}$. The degree asymmetry forces the network to be in either centralized or distributed control mode.

3.2.3 Relation with core structure

To understand the origin of the two control modes, an analytical solution for n_r is developed based on the greedy leaf removal (GLR) procedure. The solution reveals the connection between control modes and core structure. As demonstrated in Sec. 2.3, the GLR procedure can be used to construct a maximum matching in large random bipartite networks. To enhance readability, some of the arguments are repeated. The goal is to calculate n_r, which is equivalent to counting the nodes v^- that are always matched on the V^- side.

During GLR, a leaf is removed in each step that consist of node v_1 with degree 1 and node v_2 with possibly higher degree. To construct a maximum matching, the link $(v_1 - v_2)$

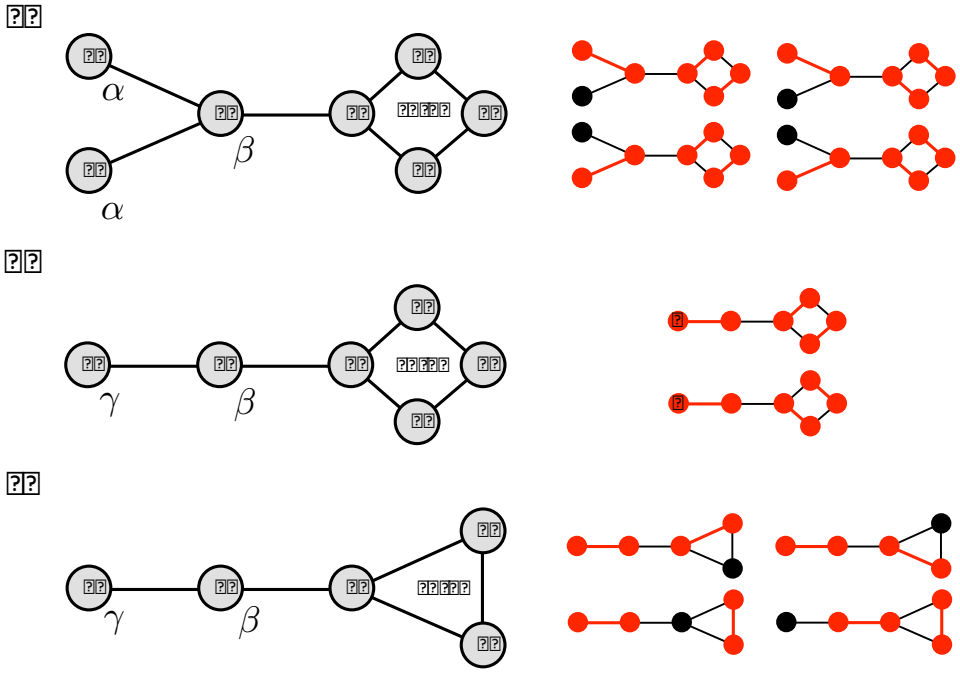


Figure 3.13: **Possible maximum matchings.** On the left, three small exmaple networks are shown. And on the right, all possible maximum matchings are listed. For simplicity unipartite undirected examples are shown, however, the conclusions are easily extended to bipartite undirected networks. **(a)** Example to illustrate that β-removable nodes are always matched. Node v_1 and v_2 are α-removable, and v_3 is β-removable. In the first step of the GLR procedure, there are two options: either $(v_1 - v_3)$ or $(v_2 - v_3)$ is removed directly and added to the matching. In either case, v_3 is matched. **(b)** Example to illustrate that γ-removable nodes are always matched if the nodes in the core are all always matched. Node v_1 is γ-removable, because it is a leaf, but cannot become isolated. In the first step of the GLR procedure, there is only one option: $(v_1 - v_2)$ is removed directly and added to the matching. Therefore, v_3 is matched. **(c)** Example to illustrate that γ-removable nodes are *not* always matched if the nodes in the core are *not* all matched. If the core is perfectly matched, all possible maximum matchings can be constructed by different realizations of the GLR procedure. However, this is not true, if not all nodes are matched in the core: instead of placing the directly removed link $(v_1 - v_2)$ into the matching, one can opt to include the indirectly removed link, see the fourth maximum matching.

is added to the matching. Now all links adjacent to v_2 are not allowed in the matching, and therefore they are removed too. This is repeated, until there are no leaves left, i.e. only the core remains. It was shown that in large unipartite random networks the core can be asymptotically matched, i.e. the probability of randomly choosing an unmatched node is zero [68]. However, in bipartite networks a new limitation arises. Let C^+ and C^- denote the set of nodes that are in the core and are in V^+ or V^-, respectively. The core size is denoted as $N^{\pm}_{\text{core}} = |C^{\pm}|$. If $N^{+}_{\text{core}} \neq N^{-}_{\text{core}}$, the matching cannot be larger than the smaller side of the core.

As a reminder, α-removable nodes are nodes that can become isolated during the GLR procedure, and β-removable nodes are nodes that can be removed as a neighbor of a leaf. On Fig. 3.13a, v_3 is a β-removable node connected to two α-removable leaves v_1 and v_2. Preforming GLR, link $(v_1 - v_3)$ or $(v_2 - v_3)$ is added to the matching, either way v_3 is matched. Indeed, it is generally true that β-removable nodes are always matched, and α-removable nodes are not always matched. If there is no core, a node is either α-removable or β-removable, therefore $n_\mathrm{r} = n_\beta^-$, where n_β^- is the fraction of β-removable nodes on the V^- side. To calculate n_β^-, recall the local rule stating that node v is β-removable if it has at least one neighbor that is α-removable in the absence of v, which translates to

$$n_\beta^- = 1 - G^-(1 - \alpha^+), \tag{3.29}$$

where α^+ is calculated using Eq. (2.23).

If a core exists, there are nodes that are neither α- or β-removable. These nodes are either in the core or belong to a category called γ-removable. A node is γ-removable if it can be removed as a leaf, but cannot become isolated (e.g. node v_1 on Fig. 3.13b,c). There are two possibilities either (i) $N_\mathrm{core}^+ < N_\mathrm{core}^-$ or (ii) $N_\mathrm{core}^+ \geq N_\mathrm{core}^-$.

In case (i) $N_\mathrm{core}^+ < N_\mathrm{core}^-$, almost all of the nodes in C^- are not always matched, consequently γ-removable nodes in V^- are also not always matched (for explanation see Fig. 3.13c). Therefore only β-removable nodes are always matched in V^-, meaning that $n_\mathrm{r} = n_\beta^-$.

In case (ii) $N_\mathrm{core}^+ \geq N_\mathrm{core}^-$, all of the nodes in C^- are always matched, consequently γ-removable nodes in V^- are also always matched (for explanation see Fig. 3.13b). Therefore only α-removable nodes are not always matched in V^-, meaning that $n_\mathrm{r} = 1 - n_\alpha^-$, where

$$n_\alpha^- = G^-(\beta^+), \tag{3.30}$$

and β^+ is calculated using Eq. (2.23).

Stiching these results together, it is found that

$$n_\mathrm{r} = \begin{cases} n_\beta^-, & \text{if } c < c^* \text{ or } N_\mathrm{core}^+ < N_\mathrm{core}^- \\ 1 - n_\alpha^-, & \text{if } c \geq c^* \text{ or } N_\mathrm{core}^+ \geq N_\mathrm{core}^- \end{cases}. \tag{3.31}$$

The argument presented above is by no means rigorous proof, therefore the results are validated by numerical simulations (Fig. 3.12b,c). Furthermore, the results are confirmed by a more rigorous, but less insightful, analytical solution of Ref. [21].

Equation (3.31) provides means to explain the emergence of the two control modes. The centralized control mode corresponds to the case when $N_\mathrm{core}^+ < N_\mathrm{core}^-$, and the dis-

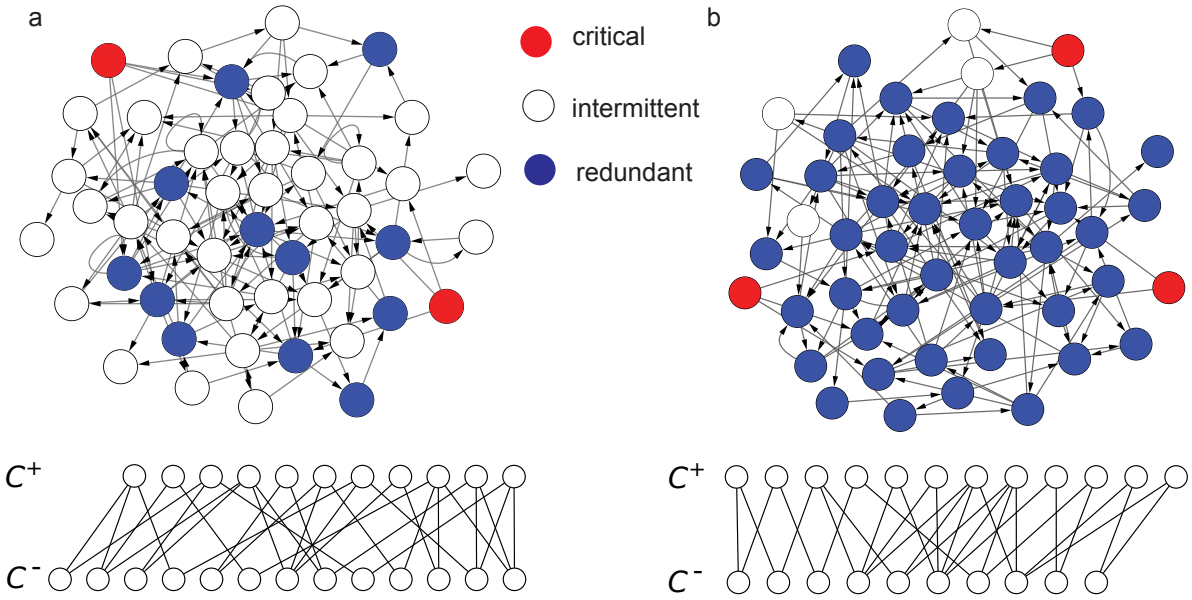


Figure 3.14: **Control modes and core structure. (a)** A network in centralized control mode and its core in the bipartite representation. As $N^+_{\text{core}} < N^-_{\text{core}}$, almost all nodes in C^- are not always matched, meaning that almost all nodes in C^- are intermittent. **(b)** A network in distributed control mode and its core in the bipartite representation. As $N^+_{\text{core}} \geq N^-_{\text{core}}$, almost all nodes in C^- are always matched, meaning that almost all nodes in C^- are redundant.

tributed control mode corresponds to $N^+_{\text{core}} \geq N^-_{\text{core}}$ (Fig. 3.14). If the degree distribution is symmetric ($p_{\text{in}}(k) = p_{\text{out}}(k)$), the expected core size on the two sides is equal, small fluctuations of core sizes $N^{\pm}_{\text{core}}$ cause one of them to be greater than the other with equal probability. Therefore, networks generated with the same parameters will be in centralized or distributed mode with equal chance, in line with the numerical observations. For networks with asymmetric degree distribution ($p_{\text{in}}(k) \neq p_{\text{out}}(k)$), the expected core size on the two sides is not equal, forcing the network to be in one of the control modes depending on the degree distribution. However, if the network size is small the fluctuations of the core size may exceed the expected difference, making both control modes possible, but this time the probability of the two modes are not equal. And indeed, this predicted effect can be observed (Fig. 3.15).

3.3 Summary

In this chapter, two distinct problems were studied. Both investigations relied heavily on the mapping between the minimum driver node problem and the maximum matching problem which was established by the minimum input theorem. The connection between

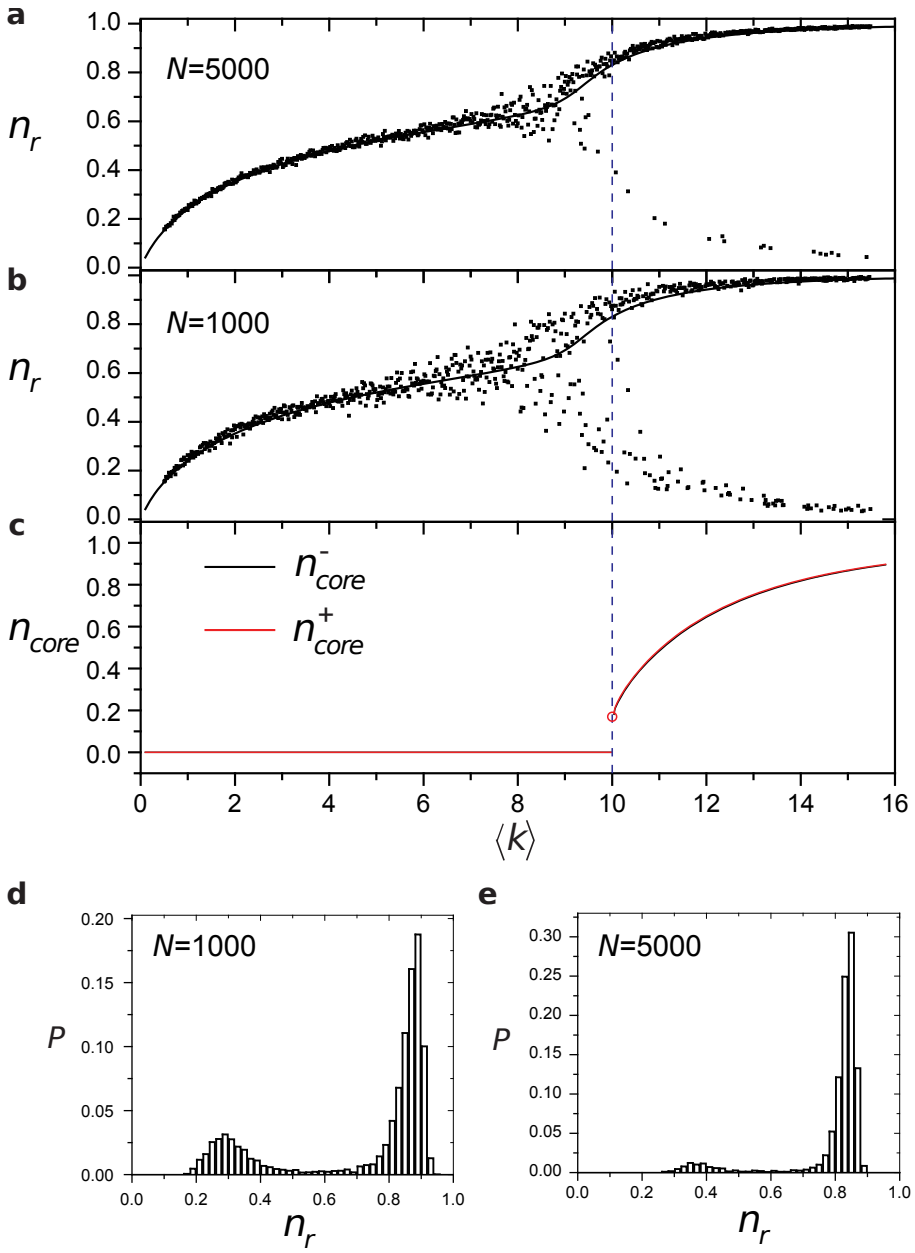


Figure 3.15: **Finite size effect and the coexists of control modes.** Finite size effect allows the coexistence of the two different control modes in scale-free networks with $\gamma_{\text{out}} = 3$ and $\gamma_{\text{in}} = 2.85$. **(a,b)** n_{r} versus the average total degree $\langle k \rangle$ for network sizes $N = 5,000$ and $N = 1,000$. Each data point is one independent measurement, the continuous line is the analytic solution valid in the large network limit $N \to \infty$. Most finite size networks are in centralized control mode, following the analytical solution. However a small number of networks are in distributed control mode. **(c)** The relative size of C^+ (n^+_{core}) and C^- (n^-_{core}). The difference between n^+_{core} and n^-_{core} is small, allowing random fluctuations to change the control mode. **(d-e)** The distribution $P(n_{\text{r}})$ for network sizes $N = 1,000$ and $N = 5,000$. For the larger network the weight of the peak corresponding to the lower branch in **a** and **b** diminishes.

core percolation and maximum matching was exploited to derive analytical solutions.

In Sec. 3.1, the effect of higher order correlations on n_{D} was explored. First, the effect of clustering, community structure, degree-degree correlations, single node degree correlations and self-loop density have been numerically investigated, reveling that – apart from degree distribution – degree-degree correlations are the most important network characteristics that influences n_{D}. In directed networks each node has an in- and an out-degree, therefore four types of correlations are possible: in-in, in-out, out-in and out-out. Both numerical simulations and the analytical solution identified three different classes of behavior depending on the type of correlation present in the network: (i) out-in correlations have the strongest effect, negative (positive) r_{oi} increases (decreases) n_{D}; (ii) in-in and out-out correlations have the same effect, and both negative and positive $r_{\mathrm{ii/oo}}$ increases n_{D} in a symmetric fashion; (iii) in-out correlations has no effect. These results were found to be coherent with measurements in various real networks.

In Sec. 3.2, node categories were introduced based on their role in network control. Generally, there are multiple possible minimum driver node sets (MDS) which prompts the following classification: (i) a node is critical if it is in all MDSs; (ii) redundant nodes are not included in any MDSs; and (iii) intermittent nodes are included only in some MDSs. It was found that above a critical average degree c^* two control modes exist: (i) in centralized control mode most nodes are redundant, meaning that the driver nodes are selected from a small set of nodes; (ii) in distributed control mode the number of redundant nodes is low, meaning that most nodes are drivers in some MDSs. I showed that the control mode in model networks is determined by the core structure: if $N^{+}_{\mathrm{core}} \geq N^{-}_{\mathrm{core}}$ the network is in centralized control mode, and if $N^{+}_{\mathrm{core}} < N^{-}_{\mathrm{core}}$ the network is in distributed control mode.

Chapter 4

Controllability of temporal networks

In this chapter, structural controllability is extended to systems for which the timescale of the dynamics and the timescale of changes in the network topology are comparable [111]. In particular, it is necessary to take into account temporal information of the connections when the interaction events are not evenly distributed over time, but have nontrivial temporal correlations [112, 113]. Such systems include communication, trade or transportation networks [114, 115, 116, 117, 118]. Furthermore, the temporal sequence of interactions governs spreading processes [114, 115]. Consider an example, a small communication network of three individuals A, B and C (Fig. 4.1). Assume that B sends an email to C at time $t = 1$, and A sends an email to B at time $t = 2$. Neglecting the temporal sequence, one finds that information may spread from A to C. However, taking the order of the messages into account, this is obviously not possible, which has a clear consequence for control: we cannot influence C using A. Therefore, one must include the temporal aspect of the interactions, when studying the controllability of networks with time-varying topologies.

In static networks, a driver node is a node that is directly controlled and requires an independent control signal, i.e. it is the first internal node of a stem (see Sec. 1.3). In temporal networks, the underlying network changes during the control process. Therefore we do not have a control structure similar to static networks, rendering the concept of driver nodes impractical. So instead of driver nodes, input nodes are investigated. Further difference is that while the minimum number of driver nodes in static networks is efficiently found, identifying the minimum number of input nodes in temporal networks is an NP-complete problem. Therefore alternative approach has to be used to quantify the overall controllability of a temporal network. Here, the average size of the maximum controllable subnetwork N_{C} is chosen as such quantity. It is determined by calculating the dimension of the maximum controllable subspace using a single node v as input node, and this is averaged over all possible vs. Values of $n_{\mathrm{C}} = N_{\mathrm{C}}/N$ close to 1 indicate networks

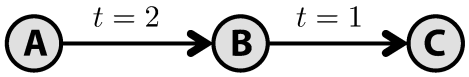


Figure 4.1: **Information propagation in a small temporal network.** In the aggregated network, information can spread from node A to C, however, it is forbidden if the temporal order of the interactions is taken into account. As a consequence, one cannot control C by imposing a control signal on A.

that are easy to control, while n_C close to 0 suggest that imposing control on the system is hard in some sense. To uncover the role of various temporal network properties, this new measure is utilized to investigate a simple class of temporal network models and a real system.

In Sec. 4.1, the necessary definitions and concepts are introduced. In Sec. 4.1.4, the independent path theorem is proved which provides the link between controllability and temporal network features. In Sec. 4.1.5, I prove that the minimum input node problem is NP-complete. In Sec. 4.1.6, the maximum controllable subspace problem is formalized and an efficient algorithm is devised to solve the problem. In Sec. 4.2, analytical solution for a simple class of model networks is derived. And in Sec. 4.3, I use various randomization procedures to investigate the role of temporal correlations in a real system.

This chapter is based on Ref. [22].

4.1 Structural controllability of temporal networks

4.1.1 Temporal networks

A directed *temporal network* $\mathcal{T}$ is defined as a set nodes $V = \{v_1, v_2, \ldots, v_N\}$ and a set temporal links $E = \{e_1, e_2, \ldots, e_L\}$ [111]. Each temporal link $(v_i, v_j, t) \in E$ consists of an ordered node pair and a time stamp, representing that there is a link pointing from node v_i to node v_j at time t. Time is measured in discrete steps $t = 0, 1, 2 \ldots$, the choice of the unit may depend on the resolution of the available dataset or modeling purposes. Furthermore, it is assumed that each link $e \in E$ has a weight $w_e \in \mathbb{R}$ associated to it, although the weight is not necessarily known.

If there exists a link $(v_i, v_j, t) \in E$, then v_i is the *in-neighbor* of v_j, and v_j is the *out-neighbor* of v_i. The temporal links (v_i, v_j, t) and (v_k, v_l, s) are *consecutive*, if $j = k$ and $t + 1 = s$. A *temporal path* P connecting node v_i and v_j from t_0 to t_1 is a sequence

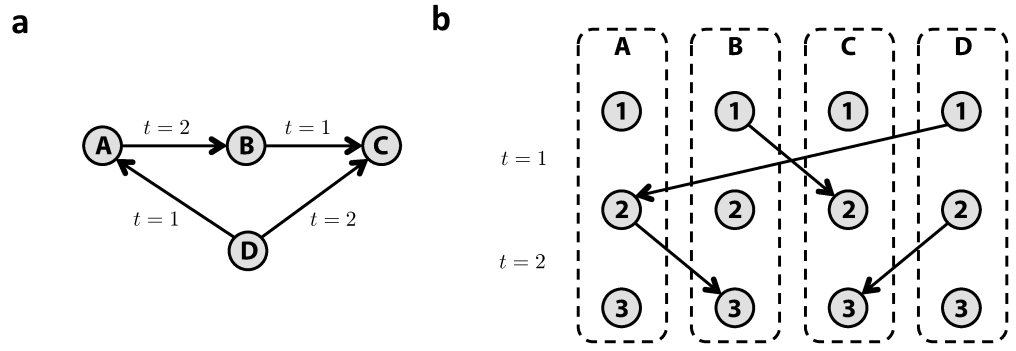


Figure 4.2: **Layered network example. (a)** A small temporal network of 4 nodes. In the static representation of the network, information can spread from node A to node C, however, due to the temporal sequence of the interactions, this is not possible. There is a time respecting path from D to B (consisting of links $(D, A, 1)$ and $(A, B, 2)$), and from D to C (consisting of link $(D, C, 2)$). The two paths do not pass the same node at the same time, therefore they are independent. **(b)** The layered network representation $\mathcal{L}(t_0 = 0, t_1 = 3)$. A copy $\hat{v}_{(i,t)}$ is made of each node v_i for each time step $t \in (t_0, t_1]$. Two nodes $\hat{v}_{(i,t)}$ and $\hat{v}_{(j,t+1)}$ are connected, if there exist a temporal link connecting nodes v_i and v_j at time t. There are altogether $|\hat{V}| = |V|(t_1 - t_0) = 12$ nodes in $\mathcal{L}(0, 3)$.

of consecutive temporal links such that the first link originates from node v_i at time $t_0 + 1$, and the last link in the sequence points at node v_j at time t_1. The path consist of $\Delta t = t_1 - t_0$ consecutive links and $\Delta t + 1$ nodes. A node by itself is a path of length 0. Two paths are *independent* if they do not pass through the same node at the same time. For a small example see Fig. 4.2a. Note that this definition of time-respecting path does not exclude waiting at nodes, it can be realized by adding self-loops.

It will be useful to represent the temporal network defined above as a *layered network* $\mathcal{L}(t_0, t_1)$ consisting of a set of nodes $\hat{V}$ and a set of static links $\hat{E}$. A copy $\hat{v}_{(i,t)} \in \hat{V}$ is made of each node $v_i \in V$ for each time step $t \in (t_0, t_1]$. Two nodes $\hat{v}_{(i,t)}$ and $\hat{v}_{(j,t+1)}$ are connected, if there exist a temporal link (v_i, v_j, t) (Fig. 4.2b). Therefore, the layered representation is a static directed acyclic network with $|\hat{V}| = |V|(t_1 - t_0)$ nodes. As a consequence, temporal paths appear as static paths in the layered representation, and independent temporal paths are simply node-disjoint paths.

4.1.2 Time-varying dynamics

In this chapter, discrete time-varying linear dynamics are studied [24]

$$\mathbf{x}(t + 1) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t), \tag{4.1}$$

where the vector $\mathbf{x}(t) \in \mathbb{R}^N$ represents the state variables, $x_i(t)$ corresponding to the state of node v_i at time t. The matrix $\mathbf{A}(t) \in \mathbb{R}^{N\times N}$ provides information about the interactions at time t: if there exists a link (v_i, v_j, t) with associated weight w, the matrix element $a_{ji}(t) = w$, otherwise $a_{ji}(t) = 0$. The vector $\mathbf{u}(t) \in \mathbb{R}^{N_\mathrm{I}(t)}$ is the control signal, we call each element of $\mathbf{u}(t)$ an *intervention*, and $N_\mathrm{I}(t)$ is the number of interventions at time t. The matrix $\mathbf{B}(t) \in \mathbb{R}^{N\times N_\mathrm{I}(t)}$ identifies the nodes we intervene at: $b_{ij}(t) \neq 0$ means that we shift $x_i(t+1)$ by $b_{ij}(t)u_j$. The total number of interventions is $N_\mathrm{I} = \sum_t N_\mathrm{I}(t)$. If we intervene at a node at any time, the node is referred to as an input.

The state $x_i(t+1)$ of node v_i is completely determined by the state of its in-neighbors at time t. If $x_i(t+1)$ is not independent from $x_i(t)$, self-interactions (e.g. diagonal entries in $\mathbf{A}(t)$) have to be added. Information about self-interactions is not always explicitly provided in network datasets.

4.1.3 Controllability

To proceed, the original definition of controllability and structural controllability has to be extended to time-varying dynamics.

A system $(\mathbf{A}(t), \mathbf{B}(t))$ is *controllable* at target time t_1 in Δt time steps, if the system can be driven to any final state $x(t_1)$ at time t_1 from any initial state $x(t_0)$ in at most $\Delta t = t_1 - t_0$ time steps. Since the system is linear, we can transform $x(t)$ to set $x(t_0) = 0$ without loss of generality.

By successively applying (4.1), the state of the system at time t_1 is

$$\mathbf{x}(t_1) = \mathbf{A}(t_1-1)\mathbf{A}(t_1-2)\cdots\mathbf{A}(t_0+1)\mathbf{B}(t_0)\mathbf{u}(t_0) + \ldots \\ + \mathbf{A}(t_1-1)\mathbf{B}(t_1-2)\mathbf{u}(t_1-2) + \mathbf{B}(t_1-1)\mathbf{u}(t_1-1). \quad (4.2)$$

The *temporal contollability* matrix is defined as [24, 119]

$$\mathbf{C}(t_0,t_1) = \left[\mathbf{A}(t_1-1)\mathbf{A}(t_1-2)\cdots\mathbf{A}(t_0+1)\mathbf{B}(t_0); \ldots; \mathbf{A}(t_1-1)\mathbf{B}(t_1-2); \mathbf{B}(t_1-1)\right], \quad (4.3)$$

where $[\mathbf{X};\mathbf{Y}]$ is the concatenation of matrices $\mathbf{X}$ and $\mathbf{Y}$, therefore $\mathbf{C}(t_0,t_1) \in \mathbb{R}^{N\times N_\mathrm{I}}$ with the total number of interventions $N_\mathrm{I} = \sum_t N_\mathrm{I}(t)$. Using this definition, the following equation is obtained:

$$\mathbf{x}(t_1) = \mathbf{C}(t_0,t_1)\mathbf{u}, \quad (4.4)$$

where $\mathbf{u} = [\mathbf{u}(t_0)^T; \mathbf{u}(t_0+1)^T; \ldots; \mathbf{u}(t_1-1)^T]^T \in \mathbb{R}^{N_\mathrm{I}}$. It is now clear that the linear rank of $\mathbf{C}(t_0,t_1)$ is the number of variables that can be set independently by the proper choice

of $\mathbf{u}$, that is $(\mathbf{A}(t), \mathbf{B}(t))$ is controllable if

$$\operatorname{rank} \mathbf{C}(t - \Delta t, t) = N. \tag{4.5}$$

The system $(\mathbf{A}(t), \mathbf{B}(t))$ is *structurally controllable* at target time t in Δt time steps, if we can set the free parameters of $(\mathbf{A}(t), \mathbf{B}(t))$ such that the system is controllable in the original sense.

A subset of state variables $C \subseteq V$ is a *controllable subspace* at target time t_1 in Δt time steps, if the state variables $x_i \in C$ can be driven to any final state at target time t_1 from any initial state in at most $\Delta t = t_1 - t_0$ time steps.

A subset of state variables $C \subseteq V$ is *a structurally controllable subspace* at target time t in Δt steps, if we can set the free parameters of $(\mathbf{A}(t), \mathbf{B}(t))$ such that C is a controllable subspace in the original sense.

4.1.4 Independent path theorem

In this section, I prove the independent path theorem which connects structural controllability to temporal network features, making it possible to formalize control related questions without detailed knowledge of link weights.

The independent path theorem is stated as follows: $C \subseteq V$ is a controllable subspace of $\mathcal{T}$ with dynamics $(\mathbf{A}(t), \mathbf{B}(t))$ at target time t_1 in Δt time steps, iff there exists $|C|$ independent paths starting from intervention points within $(t_1 - \Delta t, t_1]$ and ending at nodes $v_i \in C$ at time t_1.

To provide proof, the time-dependent controllability problem is reduced to a larger time-independent problem

$$\hat{\mathbf{x}}(\hat{t} + 1) = \hat{\mathbf{A}}\hat{\mathbf{x}}(\hat{t}) + \hat{\mathbf{B}}\hat{\mathbf{u}}. \tag{4.6}$$

The $\hat{\mathbf{x}} \in \mathbb{R}^{N \cdot \Delta t}$ state vector is created, such that $\hat{x}_{(i,t)}(\hat{t})$ corresponds to $x_i(t)$, $t \in (t_1 - \Delta t, t_1]$. Note that the index pair (i, t) is used to identify the elements of vector $\hat{x}$. To construct $\hat{\mathbf{A}}$, $\hat{a}_{(i,t);(j,t+1)}$ is set to $a_{ij}(t)$, all other elements of $\hat{\mathbf{A}}$ are set to 0. The input nodes in the time-independent system correspond to the intervention points of the time-dependent system, that is $\hat{\mathbf{u}} = [\mathbf{u}(t_0)^T; \mathbf{u}(t_0 + 1)^T; \ldots; \mathbf{u}(t_1 - 1)^T]^T = \mathbf{u}$, and $\hat{b}_{(i,t);j}$ is $b_{ij}(t)$. The network representation of the time-independent system is equivalent to the layered graph representation $\mathcal{L}(t_1 - \Delta t, t_1)$ of the temporal network $\mathcal{T}$.

It can be checked by simple multiplication that $\hat{x}_{(i,t_1)}(\hat{t} = \Delta t) = x_i(t_1)$ for all i. Therefore for every C controllable subspace of the system $(\mathbf{A}(t), \mathbf{B}(t))$, there exist a $\hat{C}$ controllable subspace of $(\hat{\mathbf{A}}, \hat{\mathbf{B}})$ such that $\hat{C} \supset \{\hat{v}_{(i,t_1)} : v_i \in C\}$. It has been previously shown [120, 44] that a subspace $\hat{C}$ of a static network is structurally controllable, iff

there exists a stem-cycle disjoint subgraph that contains all nodes in $\hat{C}$. A stem is a path starting from a node that is directly coupled to an input signal, in our case these are the intervention points. A stem-cycle disjoint subgraph is a subgraph composed of stems and cycles, such that all nodes are contained by exactly one stem or one cycle. The network representation of the time-independent system is acyclic, hence a stem-cycle disjoint subgraph in our case is simply a set of independent paths. Therefore, C is a structurally controllable subspace of $(\mathbf{A}(t), \mathbf{B}(t))$, iff there exists $|C|$ independent paths starting from intervention points and leading to each node $v_i \in C$ at time t_1.

For a small example see Fig. 4.3.

4.1.5 NP-completeness of the minimum input problem

In this section, the minimum cardinality control problem is investigated, that is we search for the minimum number of input nodes needed to control the full network at target time t in at most Δt time steps. Full control means that all N nodes are controlled in the target layer, therefore N interventions are needed. The question is how to place these interventions, such that the number of input nodes is minimized.

To formally state the problem, the set of input nodes is denoted by $D \subseteq V$. All input nodes can be used as intervention points during control, therefore the set of possible intervention points is $I = \{v_{(i,t)} : v_i \in D, t \in (t_1 - \Delta t, t_1]\}$. The set of nodes that are aimed to be controlled is $T = \{v_{(i,t_1)} : \forall v_i \in V\}$. We search for D^*, such that all nodes in T are controlled and $|D^*|$ is minimal.

For $\Delta t = 1$, the problem is trivial, all nodes have to be controlled directly. It can be shown for $\Delta t = 2$ that the problem can be mapped to an undirected maximum matching problem, thus it can be solved in polynomial time. Here, I provide proof that for $\Delta t \geq 3$ the minimum input node problem is NP-complete by reducing it to the exact 3-dimensional matching problem.

The exact 3-dimensional matching problem is stated as follows. Consider three disjunct sets X, Y and Z, such that $|X| = |Y| = |Z| = n$, and a set of triplets $T \subset X \times Y \times Z$. A three dimensional matching M is a subset of T, such that for all pairs of triplets (x_1, y_1, z_1) and (x_2, y_2, z_2), it holds that $x_1 \neq x_2$, $y_1 \neq y_2$ and $z_1 \neq z_2$. The question is that is there a perfect matching, i.e. a matching of size $|M| = n$? This is one of the original NP-complete problems of Karp [55].

A layered network $\mathcal{L}$ is constructed with 3 layers labeled $t = 1, 2, 3$, such that finding the minimum set of input nodes in $\mathcal{L}$ reveals if there is a perfect matching in T. First, a column x (three nodes x_1, x_2 and x_3, one for each layer) is created for each element of X, and links (x_1, x_2) and (x_2, x_3) are added. This is repeated for Y and Z. Then a block of 3 columns $\{u, v, w\}$ is created for each triplet $t = (x, y, z)$, and links (u_1, v_2), (u_2, v_3),

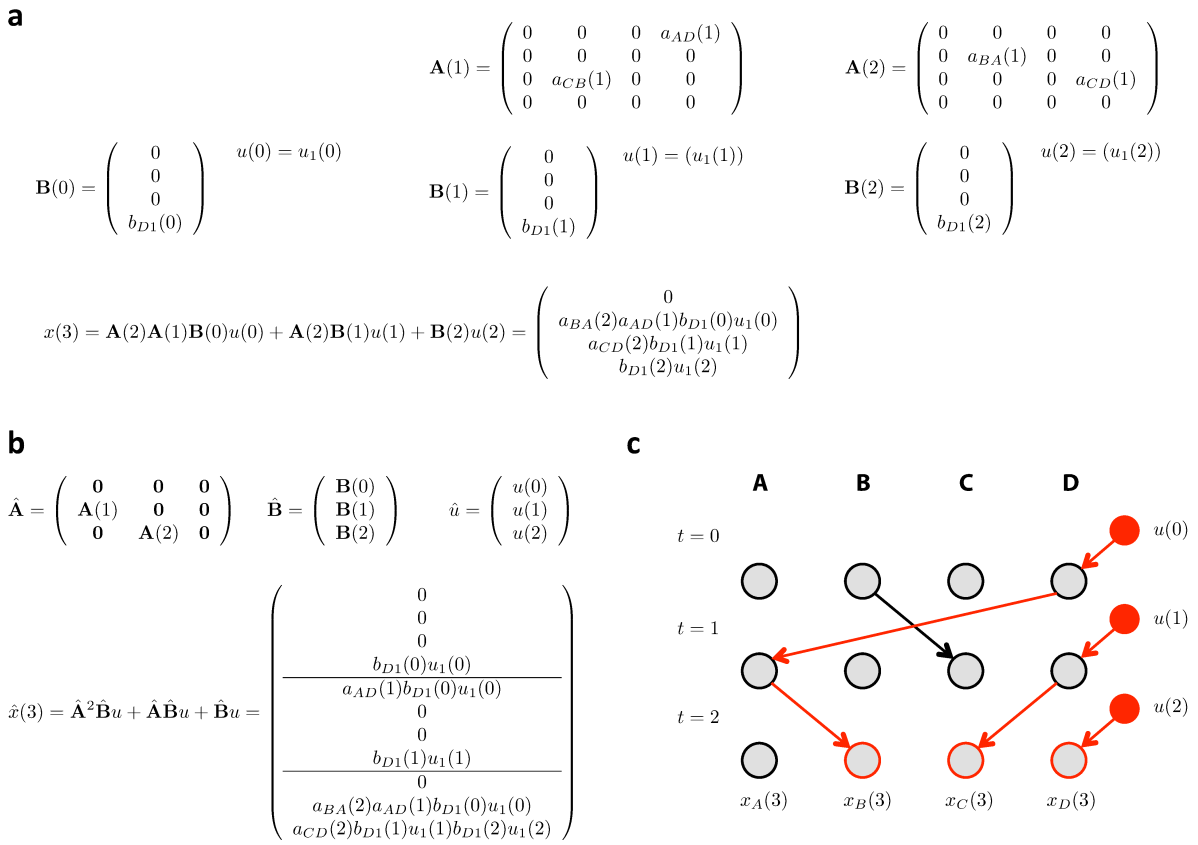


Figure 4.3: **Converting time dependent dynamics to time invariant. (a)** The matrices describing the linear dynamics corresponding to the example network in figure 4.2a. The goal is to control the system at target time $t = 3$ in $\Delta t = 3$ time steps, and node D is the only input node. The value of $x(3)$ is computed by successfully applying (4.1). **(b)** The linear time-varying dynamics are converted to a larger time-independent system the following way: Create $\hat{x} \in \mathbb{R}^{N \cdot \Delta t}$ state vector, such that $\hat{x}_{(i,t)}(\hat{t})$ corresponds to $x_i(t)$, $t \in (t_1 - \Delta t, t_1]$. Then construct $\hat{\mathbf{A}}$ by setting $\hat{a}_{(i,t);(j,t+1)}$ to $a_{ij}(t)$, all other elements of $\hat{\mathbf{A}}$ are set to 0. The input nodes $(D, 1)$, $(D, 2)$ and $(D, 3)$ correspond to the intervention points in the time-varying system. The state vector of the time-independent system $\hat{x}(t)$ at time $t = 3$ is equal to the state vector of the time-varying system $x(t)$. (c) The corresponding network of the time-independent system is equivalent to the layered graph representation $\mathcal{L}(0, 3)$ of the temporal network $\mathcal{T}$. According to the independent path theorem, nodes B, C, and D are controlled.

(v_2, w_3) and (v_2, u_3) are added. Then the block is connected to the columns representing the elements of X, Y and Z by adding links (v_1, x_2), (w_1, y_2) and (w_2, z_3).

Now, the minimum input node set is identified to control the full network $\mathcal{L}$. Let us examine possible input node choices in a block representing the triplet $t = (x, y, z)$. If we select the first node u to be an input, we can control the whole block, but we cannot control anything else. Selecting v or w by itself to be an input is not sufficient to control the block. If we select v and w both to be inputs, we control the whole block and nodes x, y and z in addition. In the case of selecting u and either v or w as inputs we do control the whole block, but we do not control all three other nodes. Finally, by selecting all three nodes u, v and w as inputs we again control the whole block and nodes x, y and z, however, we use one more input, therefore this option is not optimal.

If it is possible, it is optimal not to select any of the nodes x, y and z to be inputs, but control all of them through blocks. The minimum number of blocks necessary to control all such nodes is n. So in the best case, we use $2n$ inputs to control the x, y and z nodes and the corresponding blocks, and we use $|T| - n$ inputs to control the remaining blocks. Therefore, we need minimum $n + |T|$ inputs to control $\mathcal{L}$. If such input node set exists, a perfect matching M also exists in T: choosing v and w as input in a block corresponds to adding the triplet t to the matching M; choosing u as input node corresponds to not adding the triplet t to the matching M.

For a small example see Fig. 4.4.

4.1.6 Maximum controllable subspace

The NP-completeness of the minimum input problem renders it as an impractical way to quantify the overall controllability of temporal networks, therefore an alternative method is pursued.

Given a temporal network $\mathcal{T}$, a set of nodes $D \subseteq V$ are selected to be inputs, meaning that interventions are allowed only at these nodes. The goal is to determine the dimension of the maximum controllable subspace $N_{\mathrm{C}}(t_1, \Delta t)$, where t_1 is the target time, and Δt is the number of time steps available to reach the desired state. To formalize the problem, the layered representation $\mathcal{L}(t_1 - \Delta t, t_1)$ is used. The set of potential intervention points is $I = \{\hat{v}_{(i,t)}, \forall i, t : v_i \in D\}$ and the set of potential target nodes is $T = \{\hat{v}_{(i,t_1)}, \forall i\}$. A controllable subspace is given by a subset of T for which all nodes can be reached via independent paths from potential intervention points. Therefore, the dimension of the maximum controllable subspace $N_{\mathrm{C}}(t_1, \Delta t)$ is the maximum number of independent paths originating from I and terminating in T.

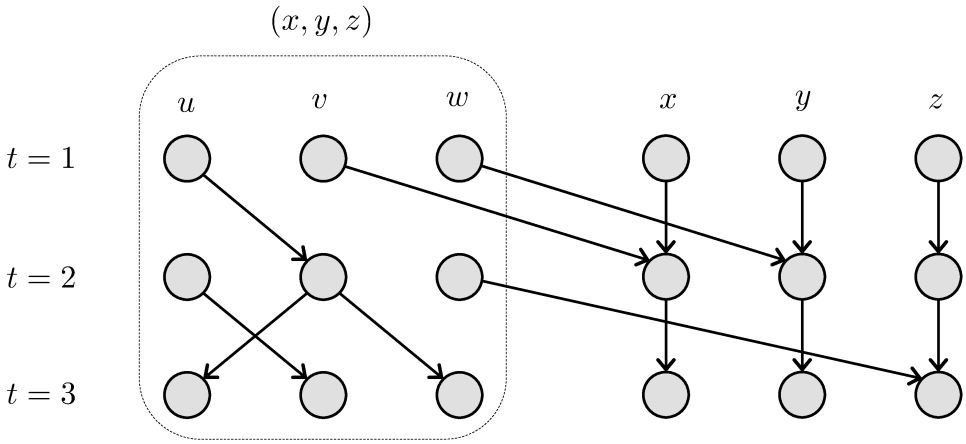
The problem of finding the maximum number of independent paths in directed networks is equivalent to solving a maximum flow problem. If the nodes in set I are sources,

a

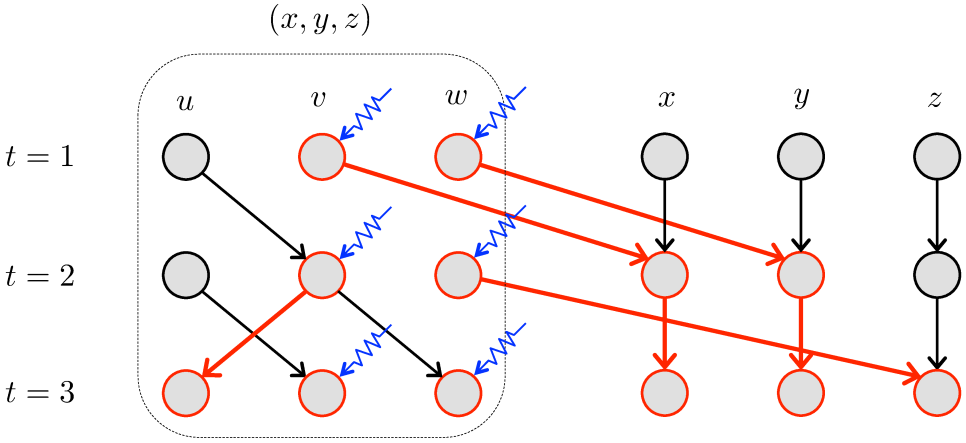
$X = \{x\}$ $Y = \{y\}$ $Z = \{z\}$ $n = |X| = |Y| = |Z| = 1$

$T = \{(x, y, z)\}$ $|T| = 1$

b



c



$N_{\mathrm{I}} = 2 = n + |T|$

Figure 4.4: **Reducing 3 dimensional matching to the minimum input node set problem for $\Delta t = 3$. (a)** A small example for the three dimensional exact matching problem. **(b)** First, a column x (three nodes x_1, x_2 and x_3, one for each layer) is created for each element of X, and links (x_1, x_2) and (x_2, x_3) are added. Similar procedure is repeated for Y and Z. Then a block of 3 columns $\{u, v, w\}$ is created for each triplet $t = (x, y, z)$, and links (u_1, v_2), (u_2, v_3), (v_2, w_3) and (v_2, u_3) are added. Then the block is connected to the columns representing the elements of X, Y and Z by adding links (v_1, x_2), (w_1, y_2) and (w_2, z_3). **(c)** The minimum number of input nodes N_{I} is determined. If $N_{\mathrm{I}} = n + |T|$ there exists an exact three dimensional matching in the original problem, if $N_{\mathrm{I}} > n + |T|$, there is no exact matching.

the nodes in T are sinks, and the capacity of each link and node is set to 1, the maximum flow is equal to the maximum number of independent paths. This problem can be solved in polynomial time, e.g. using the Ford-Fulkerson algorithm with complexity $O(|\hat{E}| \cdot N_{\mathrm{C}})$ [121, 18].

Now let $N_{\mathrm{C}}(v, t, \Delta t)$ be the dimension of the maximum controllable subspace using a single input node v, i.e. $D = \{v\}$. The overall controllability of a temporal network is characterized by the average maximum controllable subspace

$$N_{\mathrm{C}}(t, \Delta t) = \frac{1}{N} \sum_{v \in V} N_{\mathrm{C}}(v, t, \Delta t). \tag{4.7}$$

Using the procedure described above $N_{\mathrm{C}}(t, \Delta t)$ can be calculated efficiently.

4.2 Analytical solution for a simple class of networks

4.2.1 Temporal network model definition

A simple uncorrelated temporal network model is studied that can be considered as the temporal counterpart of the static uncorrelated model used in previous chapters. Initially there are N unconnected nodes, and for each time step a directed network is generated independently using either the Erdős-Rényi model or the static model as defined in Sec. 2.1. For the static model, the weights $w_{\mathrm{in}}(i)$ and $w_{\mathrm{out}}(i)$ are shuffled to eliminate any correlations persisting over time.

The generating function of the degree distribution of the Erdős-Rényi model is

$$G^{\mathrm{ER}}_{\mathrm{in/out}}(x) = \exp\left[-c(1-x)\right], \tag{4.8}$$

where $c = \langle k \rangle$.

For to asymptotically scale-free networks generated by the static model, the degree distribution of the model is given by

$$p_{\mathrm{in/out}}(k) = \sum_{i=1}^{N} \exp\left[-\frac{w_{\mathrm{in/out}}(i)}{\sum_i w_{\mathrm{in/out}}(i)} L\right] \frac{(L w_{\mathrm{in/out}}(i) / \sum_i w_{\mathrm{in/out}}(i))^k}{k!}, \tag{4.9}$$

where L is the number of links at the given time step. The corresponding generating function of this distribution is

$$G^{\mathrm{SF}}_{\mathrm{in/out}}(x) = \sum_{k=0}^{\infty} p_{\mathrm{in/out}}(k) x^k = \sum_{i=1}^{N} \exp\left[-\frac{w_{\mathrm{in/out}}(i) L}{\sum_i w_{\mathrm{in/out}}(i)} (1-x)\right]. \tag{4.10}$$

Substituting $w_{\text{in/out}}(i) = i^{-\xi_{\text{in/out}}}$, the degree distribution is

$$p_{\text{in/out}}(k) = \frac{\left[(1-\xi_{\text{in/out}})c/2\right]^{1/\xi_{\text{in/out}}}}{\xi_{\text{in/out}}} \frac{\Gamma(k-1/\xi_{\text{in/out}}, [1-\xi_{\text{in/out}}]c/2)}{\Gamma(k+1)}, \tag{4.11}$$

for large k $p_{\text{in/out}}(k) \sim k^{-(1+1/\xi_{\text{in/out}})} = k^{-\gamma}$ holds. And the generating function in the $N \to \infty$ limit is

$$G^{\text{SF}}_{\text{in/out}}(x) = \frac{1}{\xi_{\text{in/out}}} E_{1+\frac{1}{\xi_{\text{in/out}}}} \left[c(1-\xi_{\text{in/out}})(1-x)\right], \tag{4.12}$$

where $E_n(x) = \int_1^\infty dt e^{-xt} t^{-n}$ is the exponential integral function. However, for scale-free networks it is often found that finite size effects are not negligible for system sizes accessible for simulation. In these cases we have to take the finite size into account by using Eq. (4.10) explicitly.

This model is suitable to study three different aspects of temporal networks: (i) The effect of scale-free degree distribution within a time step, for static networks the degree heterogeneity is a crucial property affecting controllability, it is expected to be important in the temporal case too. (ii) The control time Δt is set by the number of layers generated. (iii) And by setting the average degree of the layers $\langle k \rangle$ the effect of the time unit selection is studied, if the dynamics controlled is slow, the time window that is aggregated into one layer contains more interactions, resulting a denser network. Other properties of real temporal networks are not captured by the model, e.g. correlations persisting over time or inhomogeneous interevent time distribution. These properties are studied in Sec. 4.3 by randomizing a temporal network abstracted from a real system.

4.2.2 Percolation in the temporal network model

Consider the case when there is only one intervention point $\hat{v}_{(i,t)}$ in the layered network. This intervention can be used to control one of the accessible nodes in a lower layer, i.e. any node that can be reached via a path originating from $\hat{v}_{(i,t)}$. The cluster of accessible nodes can be described as a cluster generated by the Galton-Watson branching process [122]: the node $\hat{v}_{(i,t)}$ has k_{out} out-neighbors, where k_{out} is drawn from the distribution $p_{\text{out}}(k)$. Each of these out-neighbors will have k_{out} out-neighbors also drawn from $p_{\text{out}}(k)$, and so forth.

The process is studied in the $N \to \infty$ limit and S_{out} denotes the probability that the branching process continues forever. Probability S_{out} is computed using the self-consistent equation

$$1 - S_{\text{out}} = G_{\text{out}}(1 - S_{\text{out}}). \tag{4.13}$$

The equation simply means that the probability that the branching process rooted at node $\hat{v}_{(i,t)}$ stops in finite steps $(1 - S_{\text{out}})$ is equal to the probability that all branching processes rooted at each out-neighbor of node $\hat{v}_{(i,t)}$ also terminate in finite steps. The equation has a trivial solution $S_{\text{out}} = 0$, at the critical point this solution loses stability:

$$1 = G'_{\text{out}}(1) = \langle k \rangle, \tag{4.14}$$

meaning that the critical point is simply determined by the average degree independent from other parameters of the degree distribution. Nodes that are roots of infinite trees form the *giant out-component*.

In the subcritical phase ($\langle k \rangle < 1$) the branching process will halt in finite steps, meaning that only finite number of nodes can be accessed. In the critical point ($\langle k \rangle = 1$) the size of the largest cluster diverges, however, the relative size is still zero. In the supercritical phase ($\langle k \rangle > 1$) the branching process will continue forever with probability S_{out}.

Similarly, the probability that a randomly selected node is an offspring of an infinite cluster is calculated:

$$1 - S_{\text{in}} = G_{\text{in}}(1 - S_{\text{in}}). \tag{4.15}$$

Nodes that are offspring of infinite trees form the *giant in-component*.

The intersection of the giant in- and out-component is referred to as the *giant component*.

4.2.3 $N_{\text{C}}(\Delta t)$ in the subcritical phase

The goal of this section is to determine the average $N_{\text{C}}(\Delta t)$ using a randomly selected node v_i as input. It is an important observation that if the network is uncorrelated, each intervention point can be treated as independently and randomly selected. In the subcritical phase the size of the accessible cluster rooted at a random node is finite. Therefore, the probability that two such clusters rooted at two randomly selected nodes overlap is 0. The probability that an intervention at $\hat{v}_{(i,t)}$ can be used to control a node at the target time is equal to the probability that a sufficiently long path is rooted at the intervention point. Hence, first the goal is to determine the cumulative distribution function of the maximum path length originating from a randomly selected point, i.e. $P(d)$ is the probability that the maximum length path originating from a node is at most d. The maximum path length from node $\hat{v}_{(i,t)}$ is 1 larger than the maximum path length originating from its out-neighbors, averaging over $p_{\text{out}}(k)$ yields

$$P(d) = G_{\text{out}}(P(d-1)). \tag{4.16}$$

The equation can be solved recursively starting from $P(d=0) = p_{\text{out}}(k=0)$.

Since the model is invariant to time shifts, the control target time is set to Δt without loss of generality. The probability that an intervention at time t can be used is given by $1 - P(\Delta t - t - 1)$ if $t \neq \Delta t$, and 1 if $t = \Delta t$. Therefore

$$N_{\text{C}}(\Delta t) = 1 + \sum_{t=1}^{\Delta t-1} 1 - P(\Delta t - t - 1) = 1 + \sum_{t=0}^{\Delta t-2} 1 - P(t). \tag{4.17}$$

For $\Delta t \to \infty$:

$$N_{\text{C}} = \lim_{\Delta t \to \infty} N_{\text{C}}(\Delta t) = 1 + \langle d \rangle, \tag{4.18}$$

which is simply one larger than the average maximum path length. For the same average degree, $\langle d \rangle_{\text{ER}}$ of Poisson distributed networks is always larger than $\langle d \rangle_{\text{SF}}$ in scale-free networks. Furthermore, in the scale-free case $\langle d \rangle_{\text{SF}}$ decreases as the degree exponent γ_{out} decreases, evincing that the presence of hubs makes control increasingly difficult (Fig. 4.5a,b).

Further insight is gained by studying the asymptotic solution of (4.16) for models with Poisson and power-law degree distribution. For the Poisson case $p_{\text{out}}(k)$ (or any distribution with finite variance σ^2_{out}), we can expand the generating function $G_{\text{out}}(x)$ around $x = 1$:

$$P(d) = G_{\text{out}}(1) - G'_{\text{out}}(1)(1 - P(d-1)) = 1 - \langle k \rangle (1 - P(d-1)). \tag{4.19}$$

Solving the recursion we get for large d

$$1 - P(d) \sim \langle k \rangle^d. \tag{4.20}$$

In other words, $P(d)$ has an exponential tail, e.g. large d values add little to $\langle d \rangle$. This means that $N_{\text{c}}(\Delta t)$ approximates its infinite solution $N_{\text{c}}(\infty)$ exponentially fast with characteristic time $\Delta t^* = 1/|\log\langle k \rangle|$, and there is little benefit from further increasing Δt. (Fig. 4.5a,b).

At the critical point $\langle k \rangle = 1$, and thus the second-order term in the expansion of the generating function is needed to extract the asymptotic behavior:

$$\begin{aligned} P(d) &= G_{\text{out}}(1) - G'_{\text{out}}(1)(1 - P(d-1)) + \frac{1}{2} G''_{\text{out}}(1)(1 - P(d-1))^2 \\ &= 1 - (1 - P(d-1)) + \frac{\sigma^2_{\text{out}}}{2} (1 - P(d-1))^2 . \end{aligned} \tag{4.21}$$

For large d, this yields

$$1 - P(d) \sim \frac{2}{\sigma^2_{\text{out}}} \frac{1}{d}. \tag{4.22}$$

From this it follows that for large Δt, $N_{\text{c}}(\Delta t) \sim \log \Delta t$, meaning that increasing Δt will increase the number of nodes that can be controlled. However, the fraction of the network that is controlled still remains 0 in the large network limit.

For scale-free networks with $\gamma < 3$, the σ^2_{out} is infinite, and therefore the simple Taylor series expansion of the generating function in (4.21) is not sufficient. To understand the effect of a power-law distribution, the generating function provided in (4.12) is transformed:

$$\begin{aligned} G_{\text{out}}(x) &= \frac{1}{\alpha} \left[c(1-\alpha)(1-x)\right]^{1/\alpha} \Gamma\left(-1/\alpha, c[1-\alpha][1-x]\right) \\ &= \frac{1}{\alpha} \left[c(1-\alpha)(1-x)\right]^{1/\alpha} \left[\Gamma(-1/\alpha) - \sum_{k=0}^{\infty} \frac{(-1)^k \left[c(1-\alpha)(1-x)\right]^{-1/\alpha+k}}{k!(-1/\alpha+k)}\right]. \end{aligned} \tag{4.23}$$

Using this in (4.16) and only keeping the first two terms:

$$\begin{aligned} P(d+1) &= \frac{1}{\alpha} \left[c(1-\alpha)P(d)\right]^{1/\alpha} \left[\Gamma(-1/\alpha) + \frac{\left[c(1-\alpha)P(d)\right]^{-1/\alpha}}{1/\alpha} + \frac{\left[c(1-\alpha)P(d)\right]^{-1/\alpha+1}}{1/\alpha - 1}\right] \\ &= 1 + cP(d) + \frac{1}{\alpha}\Gamma(-1/\alpha) \left[c(1-\alpha)P(d)\right]^{1/\alpha}. \end{aligned} \tag{4.24}$$

Consider $2 < \gamma < 3$, or equivalently $1/2 < \alpha < 1$. If $c = \langle k \rangle < 1$, the asymptotic behavior of the solution is determined by the second term, and the same solution as Eq. (4.20) is obtained, thus $N_{\text{c}}(\Delta t)$ approximates its infinite solution $N_{\text{c}}(\infty)$ exponentially fast with the same characteristic time $\Delta t^* = 1/|\log\langle k \rangle|$ (Fig. 4.5a,b).

For the solution in the critical point $\langle k \rangle = 1$, we keep the third term, and we find

$$1 - P(d) \sim d^{-\alpha/(1-\alpha)} = d^{-1/(\gamma-2)}. \tag{4.25}$$

For $2 < \gamma < 3$ this means that even in the critical point $N_{\text{c}}(\infty)$ will remain finite. However, $N_{\text{c}}(\Delta t)$ will approach its stationary value slowly, that is $N_{\text{c}}(\infty) - N_{\text{c}}(\Delta t) \sim \Delta t^{-(3-\gamma)/(\gamma-2)}$.

4.2.4 $n_{\mathbf{C}}(\Delta\tau)$ in the supercritical phase

Above the critical point, the probability that an intervention point is a root of an infinite tree is $S_{\text{out}} > 0$, meaning that there exists infinite length paths originating from the node. As a consequence, by choosing $\Delta t = \Delta\tau N$ ($\Delta\tau > 0$) it is possible to control finite fraction of the network $n_{\text{c}}(\Delta\tau) = N_{\text{c}}(\Delta\tau N)/N$ using infinite length paths, and the contribution

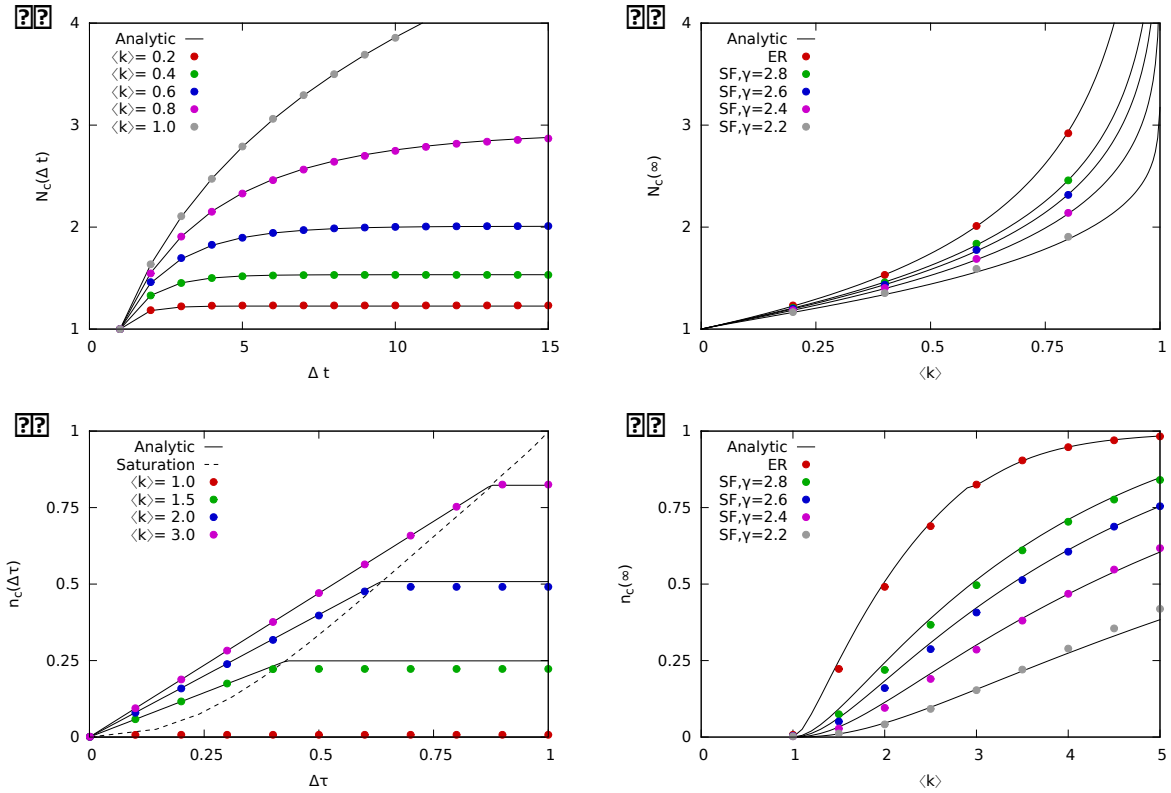


Figure 4.5: **Temporal controllability of model networks. (a)** The average maximum controllable subspace $N_C(\Delta t)$ for Poisson distributed networks in the subcritical phase. The $N_C(\Delta t)$ reaches its stationary value exponentially fast, and increasing the average degree increases both $N_C(\infty)$ and the time necessary for saturation. In the critical point $\langle k \rangle = 1$, $N_C(\Delta t)$ does not saturate for finite Δt, but increases logarithmically. **(b)** The stationary value of the average maximum controllable subspace $N_C(\infty)$ is shown in function of the average degree for Poisson and scale-free networks in the subcritical phase. The existence of hubs makes control increasingly difficult, the difference between Poisson and scale-free networks is the most prominent in the critical point, where $N_C(\infty)$ diverges for Poisson distributed networks, but remains finite for scale-free networks. **(c)** In the supercritical phase, finite fraction of the network can be controlled. The controllable fraction of the network $n_C(\Delta\tau) = N_C(\Delta\tau N)/N$ is shown for Poisson networks. For small $\Delta\tau$, $n_C(\Delta\tau)$ increases linearly, saturates at a characteristic $\Delta\tau^*$ (dashed line), and remains constant for larger $\Delta\tau$. **(d)** The plot shows $n_C(\infty)$ for Poisson and scale-free networks, and it is found again that heterogeneity of the degree distribution lowers $n_C(\infty)$. The dots are results of simulations for networks of size $N = 1,000$, each data point is an average of $10,000$ input node measurements in the subcritical phase, and $1,000$ in the supercritical phase. The continuous line shows the analytic solution, for the scale-free networks the finite size effect into account.

of finite size clusters is negligible.

Consider the case when v_i is the input node, and $v_{(i,t)}$ and $v_{(i,s)}$ are two intervention points such that both are roots of infinite trees. Since these trees cover finite fraction of the network, it can no longer be assumed that the overlap of accessible clusters has zero probability. However, being the root of an infinite tree also means that starting from the node a finite fraction S_{out} of the nodes in the target layer can be reached, and many possible paths exist. Therefore, for small $\Delta\tau$, it is assumed that whenever an intervention point is a root of an infinite tree, it can be used as an intervention point to control one node in the target layer. This means that

$$n_{\text{c}}(\Delta\tau) = S_{\text{out}}\Delta\tau. \tag{4.26}$$

Note that the result does not depend on the in-degree distribution $p_{\text{in}}(k)$.

Examining Eq. (4.26), one might think that by allowing sufficiently large $\Delta\tau$, the entire network can be controlled. However, above a characteristic $\Delta\tau^*$, a new limitation arises, and $n_{\text{C}}(\Delta\tau)$ saturates (Fig. 4.5c), since it is limited by the maximum throughput of the giant component, i.e. the maximum number of independent infinite paths. The giant component consists of nodes in each layer that are both in the giant in-component and the giant out-component. To calculate $n_{\text{c}}(\Delta\tau)$ the first step is to determine the degree distributions $\tilde{p}_{\text{in/out}}(k)$ within the giant component. Consider two adjacent layers of nodes at $t = 1$ and $t = 2$ (Fig. 4.6). The two layers are connected by links that are active at time $t = 1$. We aim to determine $\tilde{p}_{\text{out}}(k)$ for $t = 1$. The nodes that are in the giant component in layer $t = 1$ are the nodes that are in the giant in-component and have at least one connection to nodes in layer $t = 2$ that are in the giant out-component. First, the links connecting the in-component with nodes not in the out-component are removed, this is equivalent to randomly removing $1 - S_{\text{in}}$ fraction of links. Now all nodes that have at least one connection left are members of the giant component. Therefore, to obtain $\tilde{p}_{\text{out}}(k)$ nodes with 0 connections are removed. This leads to

$$\tilde{p}_{\text{out}}(k) = (1 - \delta_{k,0}) \sum_{j=k}^{\infty} \binom{j}{k} S_{\text{in}}^k (1 - S_{\text{in}})^{j-k}, \tag{4.27}$$

and the corresponding generating function is

$$\tilde{G}_{\text{out}}(k) = \frac{G_{\text{out}}(1 - S_{\text{in}} + S_{\text{in}}x) - G_{\text{out}}(1 - S_{\text{in}})}{1 - G_{\text{out}}(1 - S_{\text{in}})}. \tag{4.28}$$

The in-degree distribution $\tilde{p}_{\text{in}}(k)$ is determined similarly.

To calculate a first approximation $n_{\text{C}}^{(1)}(\infty)$, the maximum number of independent

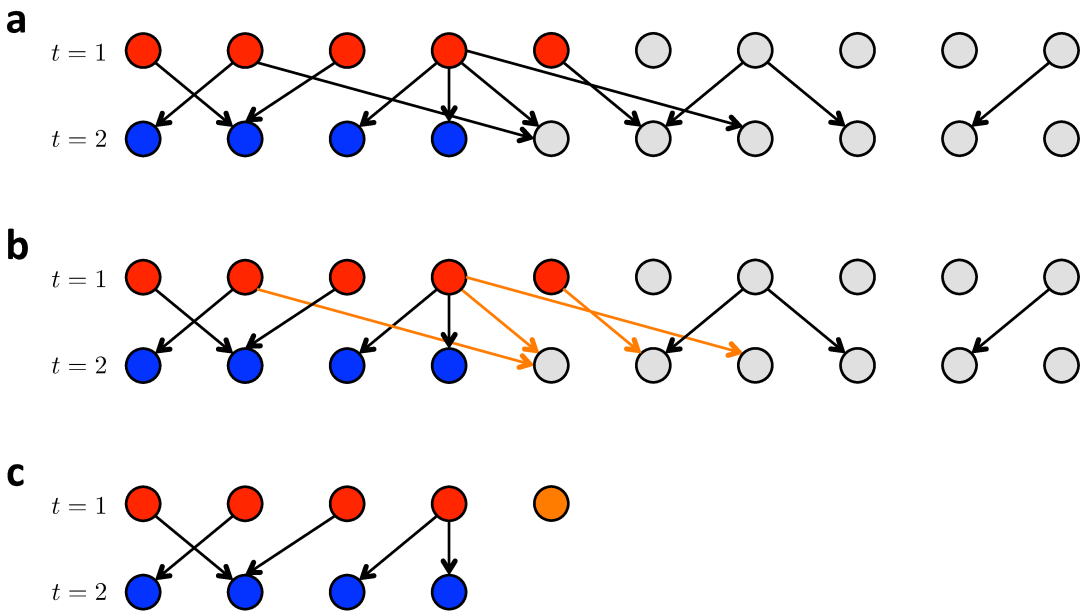


Figure 4.6: **Determining the degree distribution within the giant component. (a)** The goal is to determine the degree distribution within the giant component, i.e. nodes that belong to both the giant in-component and the giant out-component. Consider two consecutive layers $t = 1$ and $t = 2$. Assume that the red nodes are the giant in-component in layer $t = 1$, and the blue nodes are the giant out-component in layer $t = 2$. The giant component in layer $t = 1$ consists of the red nodes that are connected to at least one blue node. **(b)** First, all links are removed that lead to nodes outside the giant component, i.e. orange links. The network is uncorrelated, therefore the link removal can be treated as random. **(c)** Next, the now isolated nodes (orange) are removed, i.e. nodes that are in the in-component, but not in the out-component.

paths in the giant component connecting two subsequent layers $t = 1$ and $t = 2$ is determined, which is equivalent to finding the maximum matching in a bipartite network formed by the two layers. A matching in a network is defined as a set of links that do not share endpoints, therefore in the case of the network of two layers, the links in the matching are independent paths of length one. A node is called matched, if they are adjacent to a link in the matching. This way $n_{\mathrm{C}}^{(1)}(\infty)$ is equal to the maximum matching in a bipartite network with $S_{\mathrm{in}}S_{\mathrm{out}}N$ nodes in each layer, and degree distributions $\tilde{p}_{\mathrm{out}}(k)$ and $\tilde{p}_{\mathrm{in}}(k)$. For uncorrelated networks the size of the maximum matching can be determined analytically, the detailed calculations are provided in Sec. 4.2.5. This approximation yields an upper bound for $n_{\mathrm{C}}(\infty)$ (Fig. 4.7), because it assumes that the endpoints of the paths in layer $t = 1$ and the starting points of the paths in layer $t = 2$ can be chosen without constraint. However, when constructing a maximum matching there is no such freedom: some nodes always have to be matched [21], and in other cases some nodes cannot be included at the same time, for a small example see Fig. 4.8.

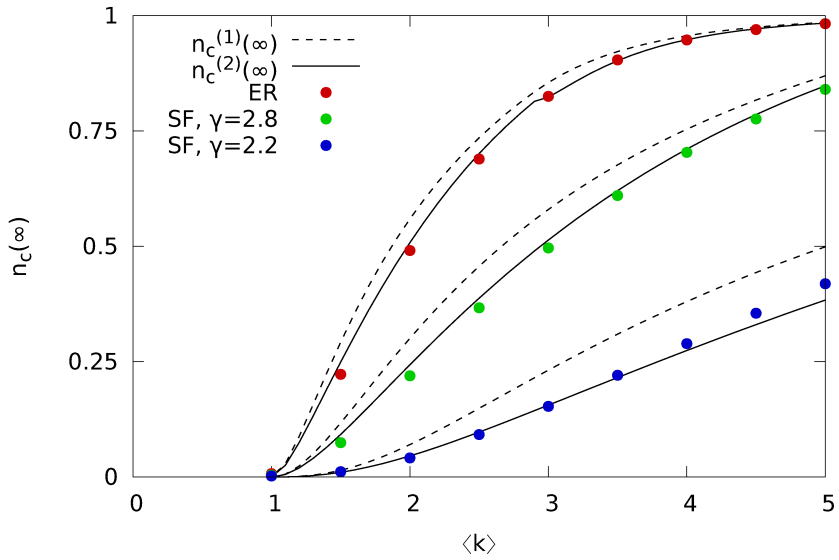


Figure 4.7: **Comparing the approximations.** The average controllable fraction of the network in function of the average degree is plotted for Poisson (ER) and scale-free (SF) degree distributions and size $N = 1,000$, each data point is an average of $1,000$ input node measurements. The dashed line is the first approximation $n_C^{(1)}(\infty)$, and the solid line is the second analytical approximation $n_C^{(2)}(\infty)$. The small break in $n_C^{(2)}(\infty)$ for the Erdős-Rényi network is a consequence of the core percolation transition described in Chapter 2, the transition point for scale-free networks is not shown on the plot.

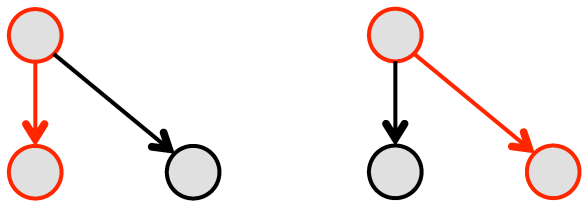


Figure 4.8: **Role of nodes in possible maximum matchings.** A small example network of 3 nodes is shown. In this network the size of the maximum matching is one, and there are two possible configurations highlighted in red. The top node is always matched, and the two bottom nodes cannot be matched in the same configuration.

For the next approximation $n_{\mathrm{C}}^{(2)}(\infty)$, consider three subsequent layers $t = 0, 1, 2$. Each layer contains $S_{\mathrm{in}}S_{\mathrm{out}}N$ nodes, and has degree distributions $\tilde{p}_{\mathrm{out}}(k)$ and $\tilde{p}_{\mathrm{in}}(k)$. First, the maximum matching between layers $t = 0$ and $t = 1$ is examined, and the set of nodes A in layer $t = 1$ is identified such that nodes in A are matched in all possible maximum matchings. In the first approximation, these nodes will always be endpoints of independent paths. However, if cannot be matched in the next layer, they will become dead ends. Therefore, the number of nodes in A that cannot be matched at the same time will be the next correction to $n_{\mathrm{C}}^{(1)}(\infty)$ (Fig. 4.9). To calculate the correction, we find the maximum matching in the bipartite network formed by nodes in A in layer $t = 0$, and all nodes in layer $t = 1$. The degree distribution of nodes in A is $\tilde{p}_{\mathrm{out}}(k)$, and the degree distribution of nodes in layer $t = 2$ can be calculated by randomly removing $1 - |A|/S_{\mathrm{in}}S_{\mathrm{out}}N$ fraction of links from $\tilde{p}_{\mathrm{in}}(k)$, similarly to (4.27). The number of nodes in A is determined using the equations developed in Sec RRR.

Similar correction can be computed for the set of nodes B in layer $t = 1$ that are always matched from layer $t = 2$, but cannot be matched at the same time from layer $t = 0$. It is found that $n_{\mathrm{C}}^{(2)}(\infty)$ approximates the numerical simulations well (Fig. 4.7).

Note that as described in Sec. RRR, for dense networks above the core percolation threshold, the number of nodes that are always matched can be drastically different depending on specific realization of the network model, e.g. two Erdős-Rényi networks generated with the same parameters can be different. This is due to a special case, when a finite fraction of nodes are "almost always" matched, meaning that we have a set of nodes A such that in each possible matching only a finite number of nodes in A are not matched. Therefore, for current purposes these nodes can be treated as always matched.

The saturation point $\Delta\tau^*$ is provided by the intersection of the two solutions, Eq. (4.26) for small $\Delta\tau$ and the approximation $n_{\mathrm{C}}^{(2)}(\infty)$ for large $\Delta\tau$.

It is now possible to compare networks with specific degree distribution. It is found trhat Poisson distributed networks are easier to control both below and above the saturation point $\Delta\tau^*$, in line with our observation in the subcritical regime (Fig. 4.5b,d).

4.2.5 Matching in bipartite networks

In this section, the relative size of the maximum matching is calculated in uncorrelated bipartite networks with arbitrary degree distribution. The difference compared to Sec. 2.3 that the number of nodes in the two sides of the bipartite network are allowed to be different. The results described here are essential to calculate the approximation $n_{\mathrm{C}}^{(2)}(\infty)$. For sake of readabilility, I repeat some arguments of Sec. 2.3.

Let $\mathcal{B}$ be a bipartite network, with two sets of nodes V^- (lower) and V^+ (upper) and a set of links E, such that each link connects one upper node $v^+ \in V^+$ and one lower

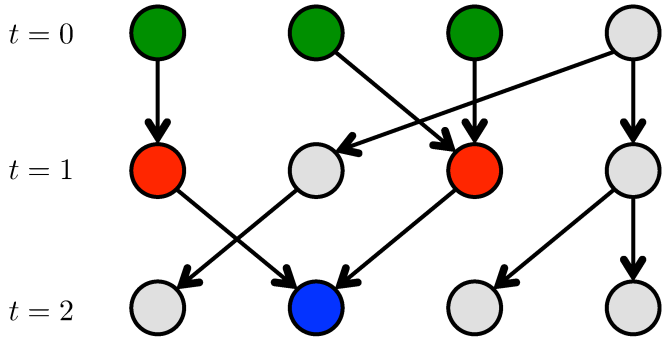


Figure 4.9: **Second approximation of $n_C(\infty)$.** Consider the giant component in three consecutive layers $t = 0, 1, 2$. Examining the maximum matching in layers $t = 0$ and $t = 1$, it is found that the red nodes are matched in all possible maximum matchings. However, in the next layer the red nodes are connected to the same node, hence they cannot be matched at the same time. Therefore, the two red nodes can only be used in one independent path. Counting such configurations provides the second approximation.

node $v^- \in V^-$. $p^+(k)$ and $p^-(k)$ are the degree distributions of the upper and lower sides, respectively. For simplicity, notations $N^+ = |V^+|$, $N^- = |V^-|$ and $L = |E|$ are used. The average degree of each layer is $c^\pm = L/N^\pm$. A matching M is the set of links that do not share endpoints, the relative size of the matching is $m^\pm = |M|/N^\pm$.

The formalism developed for core percolation in Chapter 2 is used. As a reminder, the greedy leaf removal (GLR) process is defined as follows. A leaf is selected randomly (a node with degree 1), and the leaf and its neighbor is removed together with all links adjacent to that neighbor, this step is repeated until no leaves are left; then remove all isolated nodes are removed too. The core is defined as the remainder of the network after the GLR.

The GLR process can be used to construct a maximum matching in the class of bipartite networks that are studied here. In each step, a leaf is removed that consist of node v_1 with degree 1, and node v_2 with possibly higher degree. To construct the maximum matching, the link (v_1, v_2) is added to the matching. Now all links adjacent to v_2 are not allowed in the matching, and therefore they are removed too. This is repeated, until there are no leaves left, i.e. only the core remains. It was shown that in large non-bipartite random networks the core can be asymptotically matched, i.e. the probability of randomly choosing an unmatched node is zero [68]. However, in bipartite networks there is another limiting factor: if the size of the core is different on the two sides, the size of the matching in the core cannot be larger then the smaller side.

As stated above, $m^\pm$ is the sum of the contribution of the leaf removal and the core. First, the contribution of leaf removal is calculated. For each leaf removal, one link is

added to the matching, increasing the number of matched nodes by 2, one on both sides. For each β-node there is one leaf removal. Therefore, to calculate the contribution of the leaf removal, the β-nodes are counted on both sides:

$$N_\beta^\pm = N^\pm[1 - G^\pm(1 - \alpha^\mp)]. \tag{4.29}$$

However, by doing this the case is double counted when two β-nodes are removed together. This can only happen, if in the absence of the link connecting the two nodes, both nodes are α-nodes, the probability of this event is $\alpha^+\alpha^-$ for each link. Therefore, the overall contribution is

$$N^+[1 - G^+(1 - \alpha^-)] + N^-[1 - G^-(1 - \alpha^+)] - L\alpha^+\alpha^-. \tag{4.30}$$

To determine the contribution of the core, the size of the core is calculated on both sides:

$$N_{\text{core}}^\pm = N^\pm \left[G^\pm(1 - \alpha^\mp) - G^\pm(\beta^\mp) - c^\pm\alpha^\pm(1 - \beta^\mp - \alpha^\mp)\right], \tag{4.31}$$

and select the smaller side. Therefore, all together:

$$m^- = \frac{1}{N^-}\left(N^-[1 - G^-(1 - \alpha^+)] + N^+[1 - G^+(1 - \alpha^-)] - L\alpha^+\alpha^- + \min_\pm N_{\text{core}}^\pm\right) \tag{4.32a}$$

$$= [1 - G^-(1 - \alpha^+)] + \frac{N^+}{N^-}[1 - G^+(1 - \alpha^-)] - c^-\alpha^+\alpha^- + \frac{1}{N^-}\min_\pm N_{\text{core}}^\pm. \tag{4.32b}$$

4.3 Effect of correlations in a real system

In the previous section, a simple temporal network model was studied. While the simplicity of the model allows analytical treatment, it also renders the model insufficient to capture important correlations present in real systems. To understand the role of these correlations, a real temporal network is analyzed by applying various randomization procedures. Each randomization eliminates certain correlations, and comparing $N_{\text{C}}(t, \Delta t)$ for different randomization procedures to each other and the original network reveals their effect.

A publicly available temporal network is studied that represent the email communication of a mid-size company [123, 124]. The data set contains the sender, the recipient, and the time each email has been sent. All together there are 82,927 emails between 167 employees covering a 9 month period.

The necessary temporal resolution of the network depends on the time scale of the

dynamical process we aim to control. To highlight different features of the dataset, two different temporal resolutions are used with one hour and one day time steps. These networks are obtain via coarse-graining: for each time step $t_0 \leq t < t_1$ an aggregated network is created, i.e. node v_i to v_j are connected in the coarse-grained network, if at least one email has been sent from v_i to v_j between t_0 and t_1.

The one hour coarse-grained network corresponds to a scenario when the goal is to influence the dynamics within a day. An important feature of the dataset is that it follows strong daily and weekly patterns. The bulk of the email traffic happens during the 9 hour period of the regular office hours on workdays (Fig. 4.10a-b). The average degree of the network outside the working hours is approximately 0, while during the office hours $\langle k \rangle_{\mathrm{h}} \approx 0.23$. This means that control on the hourly time scale is only possible within one day, that is each day can be considered separately. It is found that the average degree distribution is highly heterogeneous (Fig. 4.10c-d), the second moment ($\langle k^2_{\mathrm{out}} \rangle_{\mathrm{h}} \approx 0.99$, $\langle k^2_{\mathrm{in}} \rangle_{\mathrm{h}} \approx 0.36$) is much larger than the second moment of a Poisson distribution with the same average degree ($\langle k^2 \rangle_{\mathrm{ER}} \approx 0.28$).

By choosing one day time steps, slower dynamics on the network are assumed. The coarse-graining removes the daily activity patterns. To study control spanning over multiple weeks, weekends and holidays are removed, i.e. the time is measured in workdays. The average degree within a time step is $\langle k \rangle_{\mathrm{d}} \approx 1.76$ (Fig. 4.11a), which predicts that the system is in the supercritical phase, meaning that the characteristic control time is in the order of the system size. Therefore, the length of the available time period does not allow multiple independent measurements of the control process, hence focus is on controlling the system at the end of the last workday at $t = 190$. Similarly to the one hour case, the average degree distribution within a time step is heterogeneous (Fig. 4.11b-c), with second moments $\langle k^2_{\mathrm{out}} \rangle_{\mathrm{d}} \approx 18.07$ and $\langle k^2_{\mathrm{in}} \rangle_{\mathrm{d}} \approx 11.12$) compared to the second moment assuming a Poisson distribution with the same average degree $\langle k^2 \rangle_{\mathrm{ER}} \approx 4.86$.

Next, four different randomization techniques are used to identify which temporal or network characteristics of the system influence controllability.

(i) *Random time (RT)*: This randomization assigns random time steps to each link, thereby removing all temporal correlations, including both overall fluctuations in the average degree, and local correlations such as consequent and simultaneous events (Fig. 4.10a and 4.11a). This randomization does not change who interacts with whom, that is it does not change the aggregated network. However, by separating simultaneous events, the randomization changes the degree distribution within a time step indirectly (Fig. 4.10c-d and 4.11b-c). For the one hour coarse-grained network, the randomization is preformed only within the working hours of each workday.

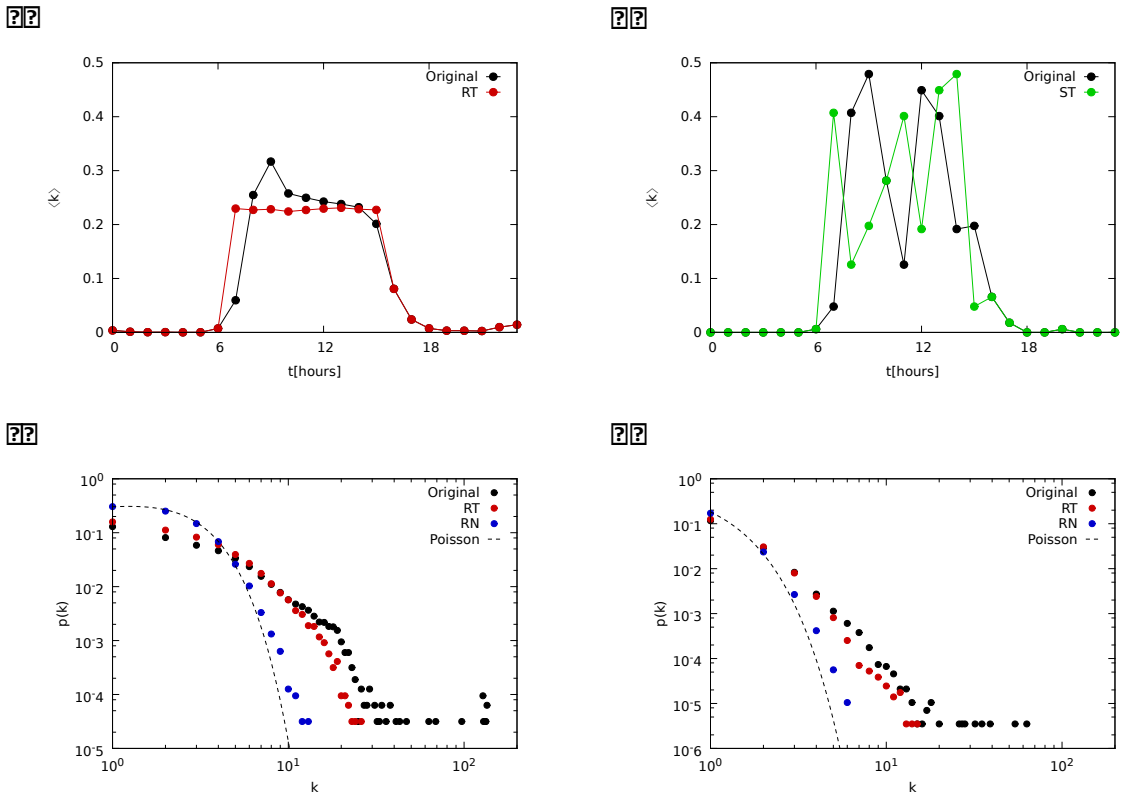


Figure 4.10: **The temporal network of email communication with one hour time resolution. (a)** The average degree of the network at different hours of a day, the plot shows the average of all workdays. The bulk of the email traffic happens during a 9 hour period corresponding to the office hours (black). The fluctuations of the average degree are eliminated by assigning random times to the links within the 9 hour active period (RT, red). **(b)** The average degree of the network at different hours of a typical workday. The average degree is characterized by large fluctuations. Shuffling the time steps keeps the fluctuations, but eliminates correlations between consecutive layers (ST, green). **(c-d)** The in- and out-degree distribution within a time step, the plot shows the average of all working hours. The original distributions are highly heterogeneous (black) compared to the Poisson distribution with the same average. Assigning random link times indirectly changes the distributions, however, the distributions remain heterogeneous (RT, red). Randomizing network topology within a time step eliminates the heterogeneity (RN, blue). Note that the RN randomization keeps the fluctuations in the average degree, hence the difference from the Poisson distribution. To obtain the Poisson distribution, both RT and RN randomization has to be applied. DPN and ST randomization do not change the degree distribution.

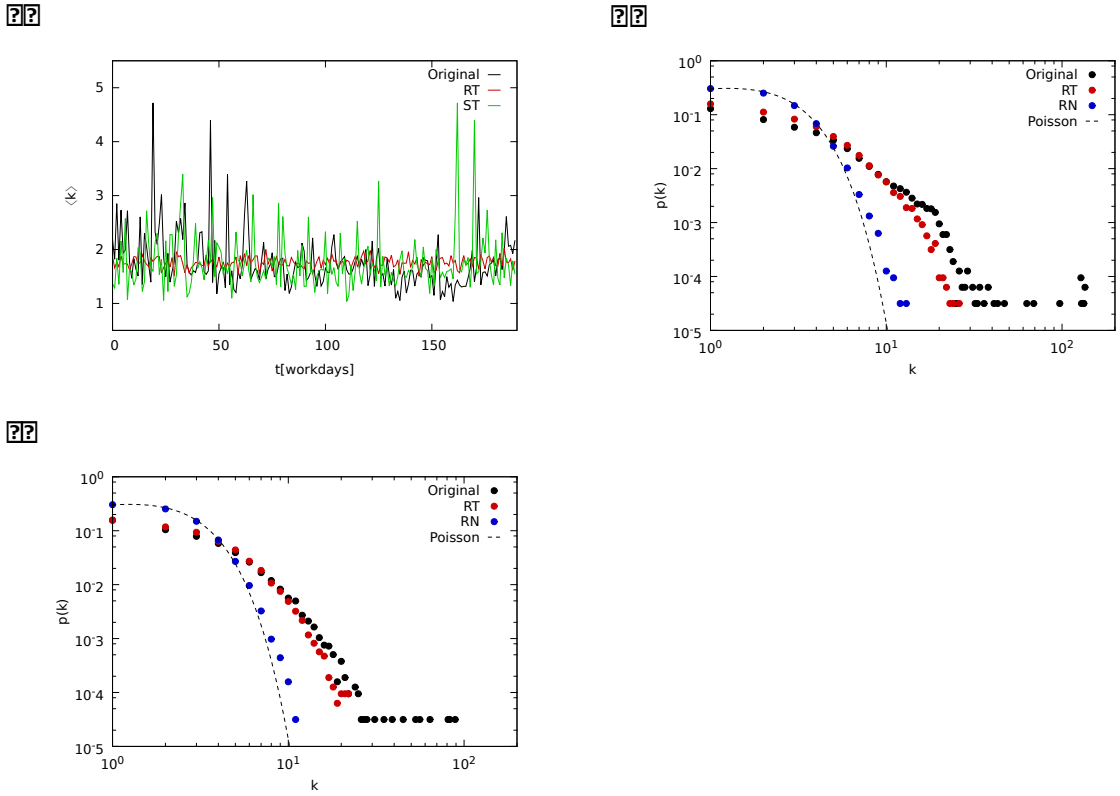


Figure 4.11: **The temporal network of email communication with one day time resolution. (a)** The average degree of the network at different days. Similarly to the one hour case, the average degree has large fluctuations. Randomly assigning link times eliminates the fluctuations (RT, red), and shuffling the time steps only eliminates correlations between consecutive time steps (ST, green). **(b-c)** The in- and out-degree distribution within a time step. The original distributions are highly heterogeneous (black) compared to the Poisson distribution with the same average. Assigning random link times indirectly changes the distributions, however, the distributions remain heterogeneous (RT, red). Randomizing the network topology within a time step eliminates the heterogeneity (RN, blue). DPN and ST randomization do not change the degree distribution.

(ii) *Shuffled time (ST)*: The order of entire time steps is shuffled. This removes all correlations between subsequent time steps, such as casual chain of events. The structure within a time steps remains unchanged (Fig. 4.10b and 4.11a). For the one hour coarse grained network, the time steps are shuffled only within the working hours of each workday.

(iii) *Random network (RN)*: In this randomization, the network for each time step is replaced by an Erdős-Rényi network with the same number of links, thereby removing all network structure, including the heterogeneity from the degree distribution (Fig. 4.10c-d and 4.11b-c). All interaction times are retained, preserving the fluctu-

ations in the average degree.

(iv) *Degree preserved network (DPN)*: For this randomization, all connections are braked and are randomly rewired within a time step. This way only the degree distribution is preserved, but all other correlations in the network structure are eliminated. Similarly to RN, we do not change the interaction times.

For further details on randomizing temporal networks see Ref. [111] and references within.

Applying RT randomization significantly increases $N_{\mathrm{C}}(t, \Delta t)$. This shows that fluctuations in the average degree of the time steps decreases $N_{\mathrm{C}}(t, \Delta t)$: a drop in the average degree acts as a bottleneck, letting through fewer independent paths. ST randomization slightly decreases $N_{\mathrm{C}}(t, \Delta t)$ suggesting that temporal correlations enhance the number of available paths, such as casual chain of events.

To investigate the effect of the underlying network, the temporal information is kept unchanged, and only the network within a time step is randomized. First, RN randomization is applied, the controllability of the resulting network dramatically increases, showing that the existence of hubs makes control difficult. Next, it is found that after preforming DPN randomization controllability of the networks is very close to the original, meaning that the degree sequence of the nodes is the main factor in determining controllability, correlations within a time step are only secondary. See Fig. 4.12 for details.

4.4 Summary

In this chapter, structural controllability was extended to systems for which the timescale of the dynamics and the timescale of changes in the network topology are comparable. Discrete time dynamics were considered on directed networks. Two modifications were added to the definition of controllability: the target time t is specified, indicating the time we aim to set the state of the system and control time; and control time Δt is also specified, indicating the number of time steps available to achieve control.

The concept of driver nodes cannot be meaningfully adopted to the case of temporal networks, so instead I focused on input nodes. The key result of the chapter is the independent path theorem which links controllability to temporal network structure. Thus it plays a similar role in case of temporal networks as Lin's structural controllability theorem and the minimum input theorem in case of static networks. In static networks, the minimum number of driver nodes served as a measure to quantify how hard is a network to control. For temporal networks, I showed that the problem of identifying the minimum set of input nodes is NP-complete for $\Delta t > 2$. Therefore the minimum number of input

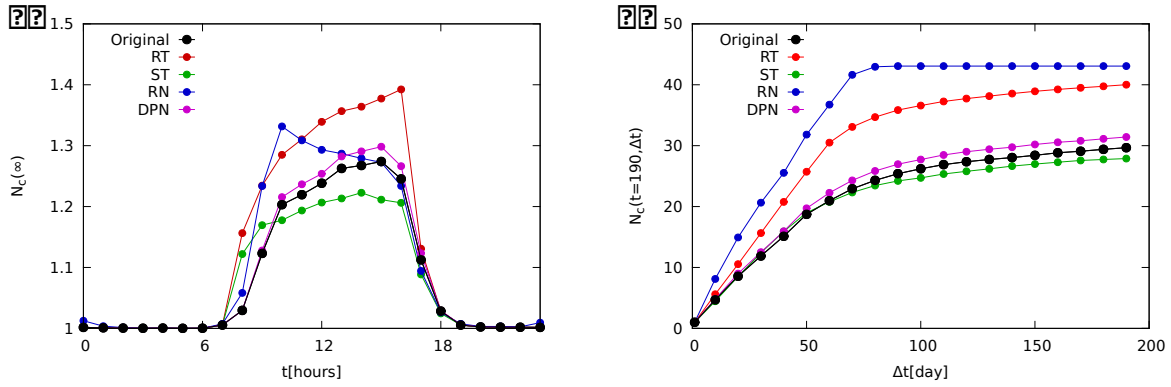


Figure 4.12: **Controllability of a real system (a)** In the first scenario, the goal is to control the company within a workday, thus the temporal resolution is set to 1 hour. The average degree of the network within a time step is $\langle k \rangle_{\mathrm{h}} \approx 0.23 < 1$, that is the system is in the subcritical phase. The plot shows $N_{\mathrm{C}}(t, \Delta t = \infty)$, i.e. the average maximum controllable subspace in function of the target time t in the $\Delta t \to \infty$ limit. The data points are the average of 189 workdays covered by the dataset. The bulk of the email traffic happens during the working hours. Therefore, the calculations are restricted to the regular office hours spanning the period $9 \leq t \leq 17$ when randomizing the time of the links. Completely removing temporal patterns by assigning random times to the links (RT), and completely removing the network structure by randomly placing the links within a time step (RN) both increase $N_{\mathrm{C}}(t, \infty)$. Shuffling the time steps (ST) only slightly increases, and randomizing the links while keeping the degree sequence (DPN) slightly decreases $N_{\mathrm{C}}(t, \infty)$. This shows that the controllability is mainly determined by the degree distribution and the overall activity pattern, correlations have smaller impact. **(b)** In the second scenario, the goal is to control the system on a longer time scale, the unit of time is set to 1 workday. In this case the average degree within a time step is $\langle k \rangle_{\mathrm{d}} \approx 1.76 > 1$, predicting that the system is above the critical point, and that the characteristic time to control the system will be in the order of the system size. Therefore, multiple independent measurements are impossible, and only control target time set to the end of the last workday is considered, meaning $t = 190$. The figure shows the $N_{\mathrm{C}}(t = 190, \Delta t)$ in function of Δt. Linear growth is observed for small Δt and saturation for large Δt, although the saturation does not happen completely in the available time period. Randomizations yield similar conclusion as in the short-term control scenario.

nodes is an impractical measure to quantify the overall controllability of time-varying networks. Instead the average maximum controllable subspace with a single input node ($N_{\mathrm{C}}(\Delta t)$) is used.

In Sec. 4.2, I developed analytical solution for a simple uncorrelated network model. The network model is able to capture the degree distribution of real systems, the effect of Δt and effect of the time unit of the dynamics. However, it does not contain temporal correlations which are present in real systems, e.g. burstiness. Using the network model, I have shown that if there is no percolating giant component $N_{\mathrm{C}}(\Delta t)$ remains finite, and the value of $N_{\mathrm{C}}(\infty)$ is reached exponentially fast. If there is a giant component, finite fraction $n_{\mathrm{C}}(\Delta\tau) = N_{\mathrm{C}}(\Delta\tau)/N$ of the network can be controlled, where $\Delta\tau = \Delta t/N$. For small $\Delta\tau$, $n_{\mathrm{C}}(\Delta\tau)$ linearly increases in function of $\Delta\tau$, however, for high $\Delta\tau$, $n_{\mathrm{C}}(\Delta\tau)$ saturates, and it is determined by the maximum number of independent paths with infinite length.

In Sec. 4.3, I studied the controllability of a real email network. I showed that the two phases observed for network models exists in real networks depending on the choice of time unit. Using various randomization procedures, I showed that degree distribution and fluctuations in the average activity are the factors that mainly determine $N_{\mathrm{C}}(\Delta\tau)$.

Chapter 5

Concluding remarks

In the last few years, control of complex networks became one of the central topics of network science, aiming to unravel how properties of complex self-organized systems affect our ability to control them and ultimately to provide methods to influence their behavior. The task is difficult for a number of reasons: (i) Complex systems are composed of many interacting entities, and these interactions form a complex network that has nontrivial effects on the dynamical behavior of the system. (ii) Most complex systems are governed by nonlinear dynamics, and nonlinear control is a challenging problem even for small systems. (iii) Typically limited information is available about both the dynamical rules governing the system and the state of the system. Facing these difficulties, various frameworks have been proposed [50, 51, 53, 56, 57], however, there is no universal solution to the problem to date.

My thesis relies on the framework of structured systems and structural controllability. The methodology was originally proposed by Lin in 1974 [28] and first employed to study complex networks by Liu et al. in 2011 [26]. Structural controllability connects linear controllability to a purely graph combinatorial problem, thus making it possible to investigate control related problems with the full arsenal of network science. Liu et al. showed that finding the minimum set of driver nodes in a network that have to be directly controlled for full control, is equivalent to finding the maximum matching in the network.

In Chapter 2, I analyzed core percolation which is a structural phase transition in complex networks [70], and I demonstrated that core percolation is deeply related to the maximum matching problem, and therefore to controllability. This connection provided means to analytically solve the minimum driver node problem for networks with arbitrary degree distribution. This was put to use in Chapter 3 to investigate two distinct questions. First, I explored how typical characteristics of real networks affect the systems controllability, finding that degree-degree correlations of connected node pairs are the

most important network features – apart from degree distribution – that influence the number of driver nodes needed for control, and high clustering and community structure have little or no effect. And second, I showed that depending on network topology two control modes can exist: centralized and distributed mode. In a given network, typically multiple equivalent minimum driver node sets (MDS) exist. In centralized mode, most nodes are never included in any of the possible MDSs, meaning that the driver nodes are selected from a small subset of nodes. In distributed mode, most nodes are included in some MDSs, therefore driver nodes are selected from a large group of nodes. I showed that in uncorrelated networks, the emergence of the two control modes coincides with the critical point of core percolation, and that the control mode of a given network is determined by the structure of the core.

In Chapter 4, I extended structural controllability to systems for which the timescale of the dynamics and the timescale of changes in the network topology are comparable. I have introduced the independent path theorem which links controllability to temporal network structure. Thus it plays a similar role in case of temporal networks as Lin's structural controllability theorem and the minimum input theorem in case of static networks. I then employed the theorem to systematically investigate the role of temporal network structure by calculating the maximum controllable subspace both in temporal network models and a real temporal network. I found that if the interaction density is low, only finite number of nodes can be controlled using a single input, but above a critical density a giant controllable subnetwork emerges spanning finite fraction of the network. I also showed that degree distribution and fluctuations in the average activity are the factors that mainly determine the size of the controllable subnetwork.

Through out my thesis, I relied on structural controllability, and using this framework comes with a number of advantages and trade-offs. For example, it assumes linear dynamics and focuses on the effect of the underlying complex network. While this provides a principled method to investigate the role of networks, it completely neglects nonlinear effects. Also, studying structural controllability requires very limited information, only the network structure of the system has to be known. Therefore it can be used to investigate any weighted directed network without detailed knowledge of the link weights, making it possible to systematically study and compare a range of complex systems. However, relying on such limited information also means that using structural controllability we can only determine where we have to intervene with the system, but it does not tell us exactly how. In summary, I believe that the main asset of structural controllability is that it provides a principled approach that only requires information that is actually available. This allows us to understand the simple case of linear dynamics, and the insights gained can serve as a foundation for work that adds further layers of complexity to the analysis.

Summary

My thesis aims to investigate how the network structure of a system influences our ability to control it. I rely on the framework of structured systems and structural controllability [28] which connects linear controllability to a purely graph combinatorial problem, thus making it possible to investigate control related problems with the full arsenal of network science. Liu et al. showed that finding the minimum set of driver nodes in a network that have to be directly controlled for full control, is equivalent to finding the maximum matching in the network [26].

First, I analyzed core percolation which is a structural phase transition in complex networks [70], and I demonstrated that core percolation is deeply related to the maximum matching problem, and therefore to controllability. This connection provided means to analytically solve the minimum driver node problem for networks with arbitrary degree distribution. Then, this was put to use to investigate two distinct questions. First, I explored how typical characteristics of real networks affect the systems controllability, finding that degree-degree correlations of connected node pairs are the most important network features – apart from degree distribution – that influence the number of driver nodes needed for control, and high clustering and community structure have little or no effect. And second, I showed that depending on network topology two control modes can exist: centralized and distributed mode. In a given network, typically multiple equivalent minimum driver node sets (MDS) exist. In centralized mode, most nodes are never included in any of the possible MDSs, meaning that the driver nodes are selected from a small subset of nodes. In distributed mode, most nodes are included in some MDSs, therefore driver nodes are selected from a large group of nodes. I showed that in uncorrelated networks, the emergence of the two control modes coincides with the critical point of core percolation, and that the control mode of a given network is determined by the structure of the core.

Finally, I extended structural controllability to systems for which the timescale of the dynamics and the timescale of changes in the network topology are comparable. I have introduced the independent path theorem which links controllability to temporal network structure. I then employed the theorem to systematically investigate the role of temporal network structure by calculating the maximum controllable subspace both in temporal network models and a real temporal network. I found that if the interaction density is low, only finite number of nodes can be controlled using a single input, but above a critical density a giant controllable subnetwork emerges spanning finite fraction of the network. I also showed that the degree distribution and the fluctuation in the average activity in the network are the factors that mainly determine the size of the controllable subnetwork.

Összefoglalás

A doktori értekezésem témája a komplex hálózatok irányíthatóságának vizsgálata. Munkám során a strukturális irányíthatóság (structural controllability) elméletére támaszkodtam [28]. A strukturális irányíthatóság kapcsolatot teremt rendszerek lineáris irányíthatósága és bizonyos gráfelméleti problémák között, így lehetővé téve a hálózatkutatás eszközeinek használatát a kérdéskör vizsgálatára. Vezérlő csúcsoknak (driver nodes) nevezzük a hálózat azon csúcsait, amelyeket közvetlenül és egymástól függetlenül befolyásolva irányíthatjuk a teljes rendszert. Liu és munkatársai megmutatták, hogy a vezérlő csúcsok legkisebb halmazának azonosítása ekvivalens a hálózat maximális párosításának megtalálásával [26].

Az értekezés első szakaszában a magperkoláció (core percolation) folyamatát vizsgáltam, amely egy strukturális fázisátalakulás komplex hálózatokban [70]. Rámutattam a magperkoláció és a maximális párosítás – és ezzel együtt az irányíthatóság – kapcsolatára. Ez a kapcsolat lehetővé tette, hogy analitikusan kiszámoljam a vezérlő csúcsok minimálisan szükséges számát egyszerű modell hálózatokban. Erre a megoldásra alapozva két kérdéskört vizsgáltam meg. Először azt vizsgáltam, hogy valódi hálózatok tipikus tulajdonságai közül melyik van a hálózat irányíthatóságára leginkább befolyással. Megmutattam, hogy irányíthatóság szempontjából – a fokszámeloszláson túl – az összekötött csúcspárok fokszámainak korrelációja a legfontosabb tulajdonság, a magas klaszterezettségnek és a sűrű alhálózatoknak nincs, vagy elenyésző a szerepe. Ezt követően megmutattam, hogy a modell hálózatok paramétereinek függvényében két, egymástól drasztikusan különböző kontroll módus jelenhet meg: a központi módus (centralized mode) és az osztott módus (distributed mode). Egy hálózatban több, egymással ekvivalens minimális vezérlő csúcs halmaz (MVH) kiválasztása is lehetséges. Ha egy hálózat központi módusban van, akkor a csúcsok nagy része egyik MVH-ban sem szerepel, míg ha osztott módusban van, akkor a csúcsok nagy része legalább egy lehetséges MVH-ban megtalálható. Megmutattam, hogy a két módus megjelenése egybeesik a magperkoláció kritikus pontjával, és hogy a mag (core) szerkezetétől függ, hogy egy adott hálózat melyik módusban van.

Végül a strukturális irányíthatóság elméletét kiterjesztettem olyan rendszerekre, amelyekben a hálózat szerkezete megváltozhat a szabályozni kívánt dinamika időskáláján. Bemutattam, hogy ilyen rendszerek strukturális irányíthatósága pusztán a rendszert leíró időfüggő hálózat szerkezetétől függ. A feltárt kapcsolatra támaszkodva megmutattam, hogy ha a kölcsönhatások sűrűsége alacsony, egy bemeneti jellel csak véges számú csúcsot tudunk irányítani. Azonban ha a kölcsönhatások sűrűsége meghalad egy kritikus értéket, a csúcsok véges hányada irányíthatóvá válik. Az irányítható részhálózat mérete elsősorban az átlagos fokszám időbeli változásaitól és a hálózat fokszámeloszlásától függ.

Bibliography

[1] Anderson, P. W. *et al.* More is different. *Science* **177**, 393–396 (1972).

[2] May, R. M. *et al.* Simple mathematical models with very complicated dynamics. *Nature* **261**, 459–467 (1976).

[3] Smith, J. M. Evolution and the theory of games: In situations characterized by conflict of interest, the best strategy to adopt depends on what others are doing. *American Scientist* 41–45 (1976).

[4] Simon, H. A. The organization of complex systems. In *Models of Discovery*, 245–261 (Springer, 1977).

[5] Wolfram, S. Universality and complexity in cellular automata. *Physica D: Nonlinear Phenomena* **10**, 1–35 (1984).

[6] Vicsek, T. *Fractal growth phenomena*, vol. 4 (World Scientific, 1989).

[7] Albert, R., Jeong, H. & Barabási, A.-L. Internet: Diameter of the world-wide web. *Nature* **401**, 130–131 (1999).

[8] Aiello, W., Chung, F. & Lu, L. A random graph model for massive graphs. In *Proceedings of the thirty-second annual ACM symposium on Theory of Computing*, 171–180 (Acm, 2000).

[9] Kwak, H., Lee, C., Park, H. & Moon, S. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World Wide Web*, 591–600 (ACM, 2010).

[10] Barabasi, A.-L. & Oltvai, Z. N. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics* **5**, 101–113 (2004).

[11] Barabási, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286**, 509–512 (1999).

[12] Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world'networks. *Nature* **393**, 440–442 (1998).

[13] Cohen, R. & Havlin, S. Scale-free networks are ultrasmall. *Physical Review Letters* **90**, 058701 (2003).

[14] Albert, R. & Barabási, A.-L. Statistical mechanics of complex networks. *Reviews of Modern Physics* **74**, 47–97 (2002).

[15] Newman, M. E. The structure and function of complex networks. *SIAM Review* **45**, 167–256 (2003).

[16] Barrat, A., Barthelemy, M. & Vespignani, A. *Dynamical processes on complex networks*, vol. 1 (Cambridge University Press Cambridge, 2008).

[17] Cohen, R. & Havlin, S. *Complex networks: structure, robustness and function* (Cambridge University Press, 2010).

[18] Newman, M. *Networks: an introduction* (Oxford University Press, 2010).

[19] Liu, Y.-Y., Csóka, E., Zhou, H. & Pósfai, M. Core percolation on complex networks. *Physical Review Letters* **109**, 205703 (2012).

[20] Pósfai, M., Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Effect of correlations on network controllability. *Scientific Reports* **3** (2013).

[21] Jia, T. *et al.* Emergence of bimodality in controlling complex networks. *Nature Communications* **4** (2013).

[22] Pósfai, M. & Hövel, P. Phase transition in the controllability of temporal networks. *arXiv preprint arXiv:1312.7595* (2013).

[23] Jia, T. & Pósfai, M. Connecting core percolation and controllability of complex networks. *Scientific Reports* **4** (2014).

[24] Kwakernaak, H. & Sivan, R. *Linear optimal control systems*, vol. 1 (Wiley-interscience New York, 1972).

[25] Luenberger, D. Introduction to dynamic systems: theory, models, and applications (1979).

[26] Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Controllability of complex networks. *Nature* **473**, 167–173 (2011).

[27] Kalman, R. E. Mathematical description of linear dynamical systems. *Journal of the Society for Industrial & Applied Mathematics, Series A: Control* **1**, 152–192 (1963).

[28] Lin, C.-T. Structural controllability. *Automatic Control, IEEE Transactions on* **19**, 201–208 (1974).

[29] Davison, E. J. & Wang, S.-H. Properties of linear time-invariant multivariable systems subject to arbitrary output and state feedback. *Automatic Control, IEEE Transactions on* **18**, 24–32 (1973).

[30] Dion, J.-M., Commault, C. & Van Der Woude, J. Generic properties and control of linear structured systems: a survey. *Automatica* **39**, 1125–1144 (2003).

[31] Shields, R. & Pearson, J. Structural controllability of multiinput linear systems. *Rice University ECE Technical Report* (1975).

[32] Lovász, L. & Plummer, M. D. Matching theory. *New York* (1986).

[33] Ford, L. R. & Fulkerson, D. R. Maximal flow through a network. *Canadian Journal of Mathematics* **8**, 399–404 (1956).

[34] Hopcroft, J. E. & Karp, R. M. An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing* **2**, 225–231 (1973).

[35] Karp, R. M. & Sipser, M. Maximum matching in sparse random graphs. In *Proceedings of the 22nd Annual IEEE Symposium on Foundations of Computer Science*, 364–375 (IEEE, 1981).

[36] Gilbert, E. N. Random graphs. *The Annals of Mathematical Statistics* 1141–1144 (1959).

[37] Erd6s, P. & Rényi, A. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci* **5**, 17–61 (1960).

[38] Clauset, A., Shalizi, C. R. & Newman, M. E. Power-law distributions in empirical data. *SIAM Review* **51**, 661–703 (2009).

[39] Cohen, R., Erez, K., Ben-Avraham, D. & Havlin, S. Resilience of the internet to random breakdowns. *Physical Review Letters* **85**, 4626 (2000).

[40] Pastor-Satorras, R. & Vespignani, A. Epidemic spreading in scale-free networks. *Physical Review Letters* **86**, 3200 (2001).

[41] Maslov, S. & Sneppen, K. Specificity and stability in topology of protein networks. *Science* **296**, 910–913 (2002).

[42] Menichetti, G., Dall'Asta, L. & Bianconi, G. Network controllability is determined by the density of low in-degree and out-degree nodes. *Physical Review Letters* **113**, 078701 (2014).

[43] Goh, K.-I., Kahng, B. & Kim, D. Universal behavior of load distribution in scale-free networks. *Physical Review Letters* **87**, 278701 (2001).

[44] Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Control centrality and hierarchical structure in complex networks. *PLoS ONE* **7**, e44459 (2012).

[45] Ruths, J. & Ruths, D. Control profiles of complex networks. *Science* **343**, 1373–1376 (2014).

[46] Cowan, N. J., Chastain, E. J., Vilhena, D. A., Freudenberg, J. S. & Bergstrom, C. T. Nodal dynamics, not degree distributions, determine the structural controllability of complex networks. *PLoS ONE* **7**, e38398 (2012).

[47] Pequito, S., Kar, S. & Aguiar, A. P. A structured systems approach for optimal actuator-sensor placement in linear time-invariant systems. In *American Control Conference (ACC), 2013*, 6108–6113 (IEEE, 2013).

[48] Nepusz, T. & Vicsek, T. Controlling edge dynamics in complex networks. *Nature Physics* **8**, 568–573 (2012).

[49] Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Observability of complex systems. *Proceedings of the National Academy of Sciences* **110**, 2460–2465 (2013).

[50] Yuan, Z., Zhao, C., Di, Z., Wang, W.-X. & Lai, Y.-C. Exact controllability of complex networks. *Nature Communications* **4** (2013).

[51] Wang, X. F. & Chen, G. Pinning control of scale-free dynamical networks. *Physica A: Statistical Mechanics and its Applications* **310**, 521–531 (2002).

[52] Sorrentino, F., di Bernardo, M., Garofalo, F. & Chen, G. Controllability of complex networks via pinning. *Physical Review E* **75**, 046103 (2007).

[53] Fiedler, B., Mochizuki, A., Kurosawa, G. & Saito, D. Dynamics and control at feedback vertex sets. i: Informative and determining nodes in regulatory networks. *Journal of Dynamics and Differential Equations* **25**, 563–604 (2013).

[54] Mochizuki, A., Fiedler, B., Kurosawa, G. & Saito, D. Dynamics and control at feedback vertex sets. ii: A faithful monitor to determine the diversity of molecular activities in regulatory networks. *Journal of Theoretical Biology* **335**, 130–146 (2013).

[55] Karp, R. M. *Reducibility among combinatorial problems* (Springer, 1972).

[56] Cornelius, S. P., Kath, W. L. & Motter, A. E. Realistic control of network dynamics. *Nature Communications* **4** (2013).

[57] Noël, P.-A., Brummitt, C. D. & D'Souza, R. M. Controlling self-organizing dynamics on networks using models that self-organize. *Physical Review Letters* **111**, 078701 (2013).

[58] Dorogovtsev, S. N., Goltsev, A. V. & Mendes, J. F. Critical phenomena in complex networks. *Reviews of Modern Physics* **80**, 1275 (2008).

[59] Bollobás, B. *Random graphs* (Springer, 1998).

[60] Callaway, D. S., Newman, M. E., Strogatz, S. H. & Watts, D. J. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters* **85**, 5468 (2000).

[61] Pittel, B., Spencer, J. & Wormald, N. Sudden emergence of a giant¡ i¿ k¡/i¿-core in a random graph. *Journal of Combinatorial Theory, Series B* **67**, 111–151 (1996).

[62] Dorogovtsev, S. N., Goltsev, A. V. & Mendes, J. F. F. K-core organization of complex networks. *Physical Review Letters* **96**, 040601 (2006).

[63] Palla, G., Derényi, I., Farkas, I. & Vicsek, T. Uncovering the overlapping community structure of complex networks in nature and society. *Nature* **435**, 814–818 (2005).

[64] Derényi, I., Palla, G. & Vicsek, T. Clique percolation in random networks. *Physical Review Letters* **94**, 160202 (2005).

[65] Achlioptas, D., D'Souza, R. M. & Spencer, J. Explosive percolation in random networks. *Science* **323**, 1453–1455 (2009).

[66] Riordan, O. & Warnke, L. Explosive percolation is continuous. *Science* **333**, 322–324 (2011).

[67] Vespignani, A. Modelling dynamical processes in complex socio-technical systems. *Nature Physics* **8**, 32–39 (2012).

[68] Zdeborová, L. & Mézard, M. The number of matchings in random graphs. *Journal of Statistical Mechanics: Theory and Experiment* **2006**, P05003 (2006).

[69] Bauer, M. & Golinelli, O. Exactly solvable model with two conductor-insulator transitions driven by impurities. *Physical Review Letters* **86**, 2621 (2001).

[70] Bauer, M. & Golinelli, O. Core percolation in random graphs: a critical phenomena analysis. *The European Physical Journal B-Condensed Matter and Complex Systems* **24**, 339–352 (2001).

[71] Weigt, M. & Hartmann, A. K. Number of guards needed by a museum: A phase transition in vertex covering of random graphs. *Physical Review Letters* **84**, 6118 (2000).

[72] Newman, M. E., Strogatz, S. H. & Watts, D. J. Random graphs with arbitrary degree distributions and their applications. *Physical Review E* **64**, 026118 (2001).

[73] Békéssy, A., Bekessy, P. & Komlós, J. Asymptotic enumeration of regular matrices. *Stud. Sci. Math. Hungar.* **7**, 343–353 (1972).

[74] Molloy, M. & Reed, B. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing* **7**, 295–305 (1998).

[75] Caldarelli, G., Capocci, A., De Los Rios, P. & Muñoz, M. A. Scale-free networks from varying vertex intrinsic fitness. *Physical Review Letters* **89**, 258702 (2002).

[76] Söderberg, B. General formalism for inhomogeneous random graphs. *Physical Review E* **66**, 066121 (2002).

[77] Lee, J.-S., Goh, K.-I., Kahng, B. & Kim, D. Intrinsic degree-correlations in the static model of scale-free networks. *The European Physical Journal B-Condensed Matter and Complex Systems* **49**, 231–238 (2006).

[78] Pastor-Satorras, R., Vázquez, A. & Vespignani, A. Dynamical and correlation properties of the internet. *Physical Review Letters* **87**, 258701 (2001).

[79] Newman, M. E. Mixing patterns in networks. *Physical Review E* **67**, 026126 (2003).

[80] Fortunato, S. Community detection in graphs. *Physics Reports* **486**, 75–174 (2010).

[81] Barrat, A. & Weigt, M. On the properties of small-world network models. *The European Physical Journal B-Condensed Matter and Complex Systems* **13**, 547–560 (2000).

[82] Leicht, E. A. & Newman, M. E. Community structure in directed networks. *Physical Review Letters* **100**, 118703 (2008).

[83] Foster, J. G., Foster, D. V., Grassberger, P. & Paczuski, M. Edge direction and the structure of networks. *Proceedings of the National Academy of Sciences* **107**, 10815–10820 (2010).

[84] Boguná, M., Pastor-Satorras, R. & Vespignani, A. Cut-offs and finite size effects in scale-free networks. *The European Physical Journal B-Condensed Matter and Complex Systems* **38**, 205–209 (2004).

[85] Menche, J., Valleriani, A. & Lipowsky, R. Asymptotic properties of degree-correlated scale-free networks. *Physical Review E* **81**, 046103 (2010).

[86] Chung, F. & Lu, L. Connected components in random graphs with given expected degree sequences. *Annals of Combinatorics* **6**, 125–145 (2002).

[87] Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. *Numerical recipes: The art of scientific computing* (Cambridge university press, 2007).

[88] Fagiolo, G. Clustering in complex directed networks. *Physical Review E* **76**, 026107 (2007).

[89] Newman, M. E. & Girvan, M. Finding and evaluating community structure in networks. *Physical Review E* **69**, 026113 (2004).

[90] Clauset, A., Newman, M. E. & Moore, C. Finding community structure in very large networks. *Physical Review E* **70**, 066111 (2004).

[91] Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008**, P10008 (2008).

[92] Rosvall, M. & Bergstrom, C. T. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* **105**, 1118–1123 (2008).

[93] Ahn, Y.-Y., Bagrow, J. P. & Lehmann, S. Link communities reveal multiscale complexity in networks. *Nature* **466**, 761–764 (2010).

[94] Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N. & Barabási, A.-L. Hierarchical organization of modularity in metabolic networks. *Science* **297**, 1551–1555 (2002).

[95] Newman, M. E. Assortative mixing in networks. *Physical Review Letters* **89**, 208701 (2002).

[96] Newman, M. E. & Park, J. Why social networks are different from other types of networks. *Physical Review E* **68**, 036122 (2003).

[97] Dorogovtsev, S. N., Mendes, J. F. F. & Samukhin, A. N. Giant strongly connected component of directed networks. *Physical Review E* **64**, 025101 (2001).

[98] Milo, R. *et al.* Network motifs: simple building blocks of complex networks. *Science* **298**, 824–827 (2002).

[99] Dunne, J. A., Williams, R. J. & Martinez, N. D. Food-web structure and network theory: the role of connectance and size. *Proceedings of the National Academy of Sciences* **99**, 12917–12922 (2002).

[100] Martinez, N. D. Artifacts or attributes? effects of resolution on the little rock lake food web. *Ecological Monographs* 367–392 (1991).

[101] Christian, R. R. & Luczkovich, J. J. Organizing and understanding a winter's seagrass foodweb network through effective trophic levels. *Ecological Modelling* **117**, 99–124 (1999).

[102] Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N. & Barabási, A.-L. The large-scale organization of metabolic networks. *Nature* **407**, 651–654 (2000).

[103] Van Duijn, M. A., Zeggelink, E. P., Huisman, M., Stokman, F. N. & Wasseur, F. W. Evolution of sociology freshmen into a friendship network. *Journal of Mathematical Sociology* **27**, 153–191 (2003).

[104] Milo, R. *et al.* Superfamilies of evolved and designed networks. *Science* **303**, 1538–1542 (2004).

[105] Song, C., Qu, Z., Blumm, N. & Barabási, A.-L. Limits of predictability in human mobility. *Science* **327**, 1018–1021 (2010).

[106] Ebel, H., Mielsch, L.-I. & Bornholdt, S. Scale-free topology of e-mail networks. *Physical Review E* **66**, 035103 (2002).

[107] Bianconi, G., Gulbahce, N. & Motter, A. E. Local structure of directed networks. *Physical Review Letters* **100**, 118701 (2008).

[108] Leskovec, J., Lang, K. J., Dasgupta, A. & Mahoney, M. W. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics* **6**, 29–123 (2009).

[109] Leskovec, J., Kleinberg, J. & Faloutsos, C. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)* **1**, 2 (2007).

[110] Leskovec, J., Kleinberg, J. & Faloutsos, C. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge Discovery in Data Mining*, 177–187 (ACM, 2005).

[111] Holme, P. & Saramäki, J. Temporal networks. *Physics Reports* **519**, 97–125 (2012).

[112] Barabasi, A.-L. The origin of bursts and heavy tails in human dynamics. *Nature* **435**, 207–211 (2005).

[113] Malmgren, R. D., Stouffer, D. B., Motter, A. E. & Amaral, L. A. A poissonian explanation for heavy tails in e-mail communication. *Proceedings of the National Academy of Sciences* **105**, 18153–18158 (2008).

[114] Vazquez, A., Racz, B., Lukacs, A. & Barabasi, A.-L. Impact of non-poissonian activity patterns on spreading processes. *Physical Review Letters* **98**, 158702 (2007).

[115] Iribarren, J. L. & Moro, E. Impact of human activity patterns on the dynamics of information diffusion. *Physical Review Letters* **103**, 038702 (2009).

[116] Jo, H.-H., Karsai, M., Kertész, J. & Kaski, K. Circadian pattern and burstiness in mobile phone communication. *New Journal of Physics* **14**, 013055 (2012).

[117] Pan, R. K. & Saramäki, J. Path lengths, correlations, and centrality in temporal networks. *Physical Review E* **84**, 016105 (2011).

[118] Konschake, M., Lentz, H. H., Conraths, F. J., Hövel, P. & Selhorst, T. On the robustness of in-and out-components in a temporal network. *PLoS ONE* **8**, e55223 (2013).

[119] Hartung, C., Reißig, G. & Svaricek, F. Sufficient conditions for strong structural controllability of uncertain linear time-varying systems. In *American Control Conference (ACC), 2013*, 5875–5880 (IEEE, 2013).

[120] Hosoe, S. Determination of generic dimensions of controllable subspaces and its application. *Automatic Control, IEEE Transactions on* **25**, 1192–1196 (1980).

[121] Ford, L. R. & Fulkerson, D. R. *Flows in networks* (Princeton University Press, 1962).

[122] Harris, T. E. *The theory of branching processes* (Springer-Verlag, Berlin, 1963).

[123] Michalski, R., Palus, S. & Kazienko, P. Matching organizational structure and social network extracted from email communication. In *Lecture Notes in Business Information Processing*, vol. 87, 197–206 (Springer Berlin Heidelberg, 2011).

[124] konect network dataset - konect (2013). URL `http://konect.uni-koblenz.de/networks/konect`.

[38]**ADATLAP**

a doktori értekezés nyilvánosságra hozatalához

I. A doktori értekezés adatai

A szerző neve: Pósfai Márton

MTMT-azonosító: 10040795

A doktori értekezés címe és alcíme:

Structure and controllability of complex networks

DOI-azonosító[39]: 10.15476/ELTE.2014.017

A doktori iskola neve: ELTE TTK, Fizika Doktori Iskola

A doktori iskolán belüli doktori program neve:

Statisztikus fizika, biológiai fizika és kvantumrendszerek fizikája Doktori Program

A témavezető neve és tudományos fokozata: Prof. Vattay Gábor, DSc

A témavezető munkahelye: ELTE TTK, Komplex Rendszerek Fizikája Tanszék

II. Nyilatkozatok

1. A doktori értekezés szerzőjeként[40]

a) hozzájárulok, hogy a doktori fokozat megszerzését követően a doktori értekezésem és a tézisek nyilvánosságra kerüljenek az ELTE Digitális Intézményi Tudástárban. Felhatalmazom a Természettudományi Kar Tudományszervezési és Egyetemközi Kapcsolatok Osztályának ügyintézőjét Bíró Évát, hogy az értekezést és a téziseket feltöltse az ELTE Digitális Intézményi Tudástárba, és ennek során kitöltse a feltöltéshez szükséges nyilatkozatokat.

b) kérem, hogy a mellékelt kérelemben részletezett szabadalmi, illetőleg oltalmi bejelentés közzétételéig a doktori értekezést ne bocsássák nyilvánosságra az Egyetemi Könyvtárban és az ELTE Digitális Intézményi Tudástárban;[41]

c) kérem, hogy a nemzetbiztonsági okból minősített adatot tartalmazó doktori értekezést a minősítés *(dátum)*-ig tartó időtartama alatt ne bocsássák nyilvánosságra az Egyetemi Könyvtárban és az ELTE Digitális Intézményi Tudástárban;[42]

d) kérem, hogy a mű kiadására vonatkozó mellékelt kiadó szerződésre tekintettel a doktori értekezést a könyv megjelenéséig ne bocsássák nyilvánosságra az Egyetemi Könyvtárban, és az ELTE Digitális Intézményi Tudástárban csak a könyv bibliográfiai adatait tegyék közzé. Ha a könyv a fokozatszerzést követőn egy évig nem jelenik meg, hozzájárulok, hogy a doktori értekezésem és a tézisek nyilvánosságra kerüljenek az Egyetemi Könyvtárban és az ELTE Digitális Intézményi Tudástárban.[43]

2. A doktori értekezés szerzőjeként kijelentem, hogy

a) az ELTE Digitális Intézményi Tudástárba feltöltendő doktori értekezés és a tézisek saját eredeti , önálló szellemi munkám és legjobb tudomásom szerint nem sértem vele senki szerzői jogait;

b) a doktori értekezés és a tézisek nyomtatott változatai és az elektronikus adathordozón benyújtott tartalmak (szöveg és ábrák) mindenben megegyeznek.

3. A doktori értekezés szerzőjeként hozzájárulok a doktori értekezés és a tézisek szövegének plágiumkereső adatbázisba helyezéséhez és plágiumellenőrző vizsgálatok lefuttatásához.

Kelt: Budapest, 2014. október 18.

.....

a doktori értekezés szerzőjének aláírása

[38] Beiktatta az Egyetemi Doktori Szabályzat módosításáról szóló CXXXIX/2014. (VI. 30.) Szen. sz. határozat. Hatályos: 2014. VII.1. napjától.

[39] A kari hivatal ügyintézője tölti ki.

[40] A megfelelő szöveg aláhúzandó.

[41] A doktori értekezés benyújtásával egyidejűleg be kell adni a tudományági doktori tanácshoz a szabadalmi, illetőleg oltalmi bejelentést tanúsító okiratot és a nyilvánosságra hozatal elhalasztása iránti kérelmet.

[42] A doktori értekezés benyújtásával egyidejűleg be kell nyújtani a minősített adatra vonatkozó közokiratot.

[43] A doktori értekezés benyújtásával egyidejűleg be kell nyújtani a mű kiadásáról szóló kiadói szerződést.