Dynamics and Structure of Complex Networks

Inaugural-Dissertation

zur Erlangung des Doktorgrades der
Mathematisch-Naturwissenschaftlichen Fakultät
der Universität zu Köln

vorgelegt von

Steffen Karalus

aus Bad Soden-Salmünster

Köln 2015

Berichterstatter: (Gutachter)	Prof. Dr. Joachim Krug
	Prof. Dr. Johannes Berg

Tag der mündlichen Prüfung: 22. Oktober 2015

Kurzzusammenfassung

Diese Arbeit untersucht die Beziehung zwischen strukturellen Eigenschaften und dem Verhalten von dynamischen Prozessen in komplexen Netzwerken. Aufbauend auf Ideen aus der biologischen Evolution oder Monte Carlo-Simulationen wird die Methode der Netzwerkevolution entwickelt. Sie dient zur Generierung von Netzwerken mit vorgegebenem dynamischen Verhalten ohne explizite Vorkenntnis, wie strukturelle Eigenschaften die Dynamik beeinflussen. Insbesondere werden Laplace-Dynamiken als Klasse von grundlegenden – einfachen jedoch hochgradig relevanten – Prozessen betrachtet. Ziel der Evolution ist ein Potenzgesetz im Spektrum der Laplace-Matrix, das die spektrale Dimension als charakteristische Größe für die Beschreibung von Diffusionsprozessen definiert. Die erfolgreich evolvierten Netzwerke weisen heterogene Strukturen, bestehend aus dicht verbundenen Kernen und dünnen Randbereichen, auf. Durch Rekonstruktion der evolvierten Netzwerke wird der Einfluss von Gradverteilung und Grad-Korrelationen auf die spektralen Eigenschaften untersucht. Hierfür werden Zufallsnetzwerke mit der Gradverteilung und den Grad-Korrelationen der evolvierten Netzwerke generiert und die Laplace-Spektren verglichen. Schließlich werden homogene Netzwerke mit dem vorgegebenen Potenzgesetz im Laplace-Spektrum durch Beschränkung der Evolution auf reguläre Netzwerke erzeugt. Die so evolvierten Netzwerke sind gekennzeichnet durch gehäuft auftretende symmetrische Motive. Dies wird für ein systematisches Coarse-Graining der Netzwerkstrukturen genutzt. Dadurch lassen sich die Laplace-Spektren eindeutig in Beiträge der symmetrischen Motive einerseits und der unterliegenden großskaligen Strukturen andererseits aufspalten.

Abstract

In this thesis the relationship between the structure of complex networks and the behavior of dynamical processes on these networks is studied by network evolution. Adopting ideas from biological evolution or Monte Carlo simulations the method of network evolution is developed as strategy to generate networks with a prescribed dynamical behavior without any prior knowledge about the structure-dynamics relation. Here, Laplacian dynamics are considered forming a basic—simple but highly relevant—class of dynamics. Networks are successfully evolved towards a power-law scaling in the spectrum of the graph Laplacian which defines the spectral dimension as important characteristic of diffusion behavior. The resulting evolved networks exhibit heterogeneous structures with densely connected cores and sparse peripheries. The influence of the degree distribution and degree correlations on the spectral behavior is investigated. To this end random networks with the degree distribution and degree correlations of the evolved networks are generated and the Laplacian spectra are compared. Finally, homogeneous networks with the prescribed power law in the Laplacian spectrum are generated by restricting the evolution to regular networks. The resulting evolved networks are found to be highly symmetric. The symmetry is exploited to construct quotient networks as systematic coarse-graining separating the Laplacian spectra into contributions from the repeating symmetric motifs on small scales and the underlying large-scale structures, which are particularly relevant in determining the spectral dimension.

Contents

List of Figures

List of Tables

Symbols

N	Number of vertices in a network.
M	Number of edges in a network.
A_{ij}	Elements of the adjacency matrix.
L_{ij}	Elements of the graph Laplacian.
k_i	Degree of vertex i, the number of edges connected to it.
$P(k)$	Degree distribution, the probability that a randomly chosen vertex has degree k.
$P_{\mathrm{e}}(k)$	Edge end distribution, the probability that a vertex at the end of a randomly chosen edge has degree k.
$P(j,k)$	Joint degree distribution, the probability that a randomly chosen edge connects vertices of with degrees j and k.
r	Assortativity coefficient measuring the degree-degree correlations.
C	Global clustering coefficient measuring the overall transitivity.
C_i	Local clustering coefficient measuring the transitivity at vertex i.
$C(k)$	Degree-dependent clustering coefficient measuring the transitivity of vertices in the degree class k.
$P_0(t)$	Average return probability, probability that a random walker is located at its origin at time t, averaged over all starting vertices.
$I(\lambda)$	Integrated Laplacian eigenvalue density.
d_{s}	Spectral dimension, describing the scaling of the Laplacian spectral density.
$\mathcal{D}(\tilde{I},\tilde{I}')$	Spectral distance between the two logarithmically integrated densities $\tilde{I}$ and $\tilde{I}'$.

1. Introduction

> Everything is simple and neat—except, of course, the world.
> *Goldenfeld and Kadanoff*

Complex Systems

Undoubtedly, the world we are living in with the phenomena we are observing and, not only as physicists, trying to understand is a complex world. It consists of a vast amount of entities of extremely different sizes, masses, and many other properties with an even larger number of interactions between them. Describing the whole complexity of this world would not only be cumbersome but is truly impossible.—To mention just two reasons for this, consider the fact that we will never be able attain full information about the state of each and every atom in the universe due to quantum mechanical uncertainty and the relativistic event horizon. Additionally, and in a certain sense more fundamentally, the device to describe the world would have to be part of and interacting with the rest of the world, leaving us without any material to store information and perform computations.—The sciences in general and in particular physics have made much progress in the description and the understanding of the world by the use of reductionism. The idea to separate the essential aspects from the less important has led to the discovery of again and again more elementary entities and more fundamental interactions. The importance of these discoveries like quantum mechanics and elementary particle physics as well as general relativity and cosmology is, of course, beyond any doubt. Nevertheless, it is a naive fallacy to conclude that other fields of science can be simply deduced from more the fundamental ones. The emergence of collective phenomena and the shifts in which aspects are considered as essential, depending on the level of description, are subsumed in the manifestation "More Is Different" by Anderson (1972):

> The behavior of large and complex aggregates of elementary particles, it turns out, is not to be understood in terms of a simple extrapolation of the properties of a few particles. Instead, at each level of complexity entirely new properties appear, and the understanding of the new behaviors requires research which I think is as fundamental in its nature as any other.

In this hierarchy of science the elementary entities of one level of description are governed by the laws of the more fundamental level, but in the transition from one to the other a completely new field emerges. These transitions are studied in statistical physics and complex systems science. The basic entities of a system and their local interactions

are considered to be known. From the non-trivial interplay of many of those entities, however, the complexity of the new field arises.

But what actually is complexity? A strict definition of which systems and phenomena are considered as "complex" and which as "simple" does not exist so far. However, there seems to be an agreement on a number of aspects a complex system has. The emergence of collective phenomena as a central feature of complex systems was already mentioned above. On the level of observations important features are structure formation and intermittency or "structure with variations" (Goldenfeld and Kadanoff, 1999). Also self-organization is considered as an important feature of many complex systems. This idea was very successfully driven forward in the concept of *self-organized criticality* (Bak *et al.*, 1987). In a systematic analysis, Chu *et al.* (2003) reason that necessary features of complexity are internal inhomogeneity, adaptivity of agents, nonlinear interactions, and a net-like causal structure as well as radical openness and contextuality. While the first four form the constituents of *complex adaptive systems* which can be described by *agent-based models* they conclude that together with the latter two a general unified theory of complexity might not be possible to achieve. For the study of complex systems, Amaral and Ottino (2004) consider nonlinear dynamics, statistical physics, and network theory as the most important tools. Out of these three, network theory is the youngest and most rapidly developing field. As an integral component of complex systems it has become a very active research topic over the last fifteen years.

Networks

A network model describes binary relations between elements of a discrete set of entities. This includes traditional arrangements such as regular lattices (as studied, e.g., in solid state physics), all-to-all or randomly chosen interactions as well as more complex structures with very distinct features. Since interactions in many natural, technological, or socio-economic systems are indeed pairwise, or can at least be approximated as such, networks provide a quite general description of interaction patterns. In the following, some important examples of systems which were successfully modeled as networks are presented. More detailed information on these systems and references can be found in several review articles (Albert and Barabási, 2002; Dorogovtsev and Mendes, 2002; Newman, 2003b) and a book (Newman, 2010).

In biology, a number of systems over all relevant length scales can be usefully represented as networks. On the sub-cellular level, *protein-protein interaction networks* describe the interactions between different proteins in a biological cell. Two types of proteins are connected in this description if they can, in their native folded state, interlock to form protein complexes. These interactions primarily do not involve chemical reactions. In a *metabolic network* the pathways of cell metabolism are recorded, i.e., the chemical substances and their reactions involved in the process of breaking down nutrients and making energy and useful biomolecules available for the functioning of the cell. This is an example of a system which is most effectively modeled as a directed network. In order to have well defined metabolic pathways the chemical reactions are

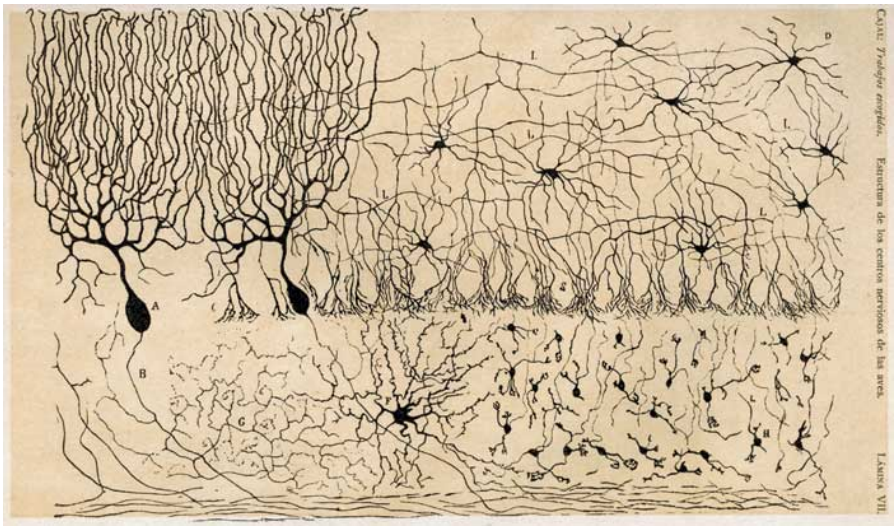

Figure 1.1. Illustration of a neural network. Shown is a network of nerve cells from the chick cerebellum in an original drawing by the Nobel laureate Santiago Ramón y Cajal. (Ramón y Cajal, Estructura de los centros nerviosos de las aves, Madrid, 1905)

best described by asymmetric relations between the different metabolites. The same holds true for *gene regulation networks* describing how the expression of different genes in a cell regulate each other. If a certain gene is expressed, i.e., DNA is transcribed into messenger RNA which is then translated into gene products such as proteins, its products may promote (excitatory interaction) or suppress (inhibitory interaction) the expression of other genes building up a complex regulatory system. Similar regulatory mechanisms with excitatory and inhibitory connections which are, however, of completely different origin are found in *neural networks*. They consist of specialized excitable nerve cells, so-called neurons, which are electrically connected by synapses or gap junctions. While there are about 10^{11} neurons found in a human brain the neural network of the nematode worm *Caenorhabditis elegans* consist of only 302 neurons making it an ideal model system which has been very well studied. A historic illustration of a neural network in the cerebellum of a chick is displayed in figure 1.1. On the scale of a whole organism, *vascular networks* describe the connectivity structure of blood vessels or equivalent transport systems in plants. *Physiological networks* are used as models for the interactions of different organs of the human body. On even larger, ecological scales *food webs* describe the hunting relations between different species in an ecosystem. As predator-prey relation are in general asymmetric, these are also most naturally modeled as directed networks.

There is also a number of technological systems which can be very effectively described as networks. Although these systems are man-made they are often not completely under control of a central organization unit and exhibit features of complex systems. A very famous and well-studied example is the *Internet*, a network of physical data connections between computers and other telecommunication devices such as routers. It has no

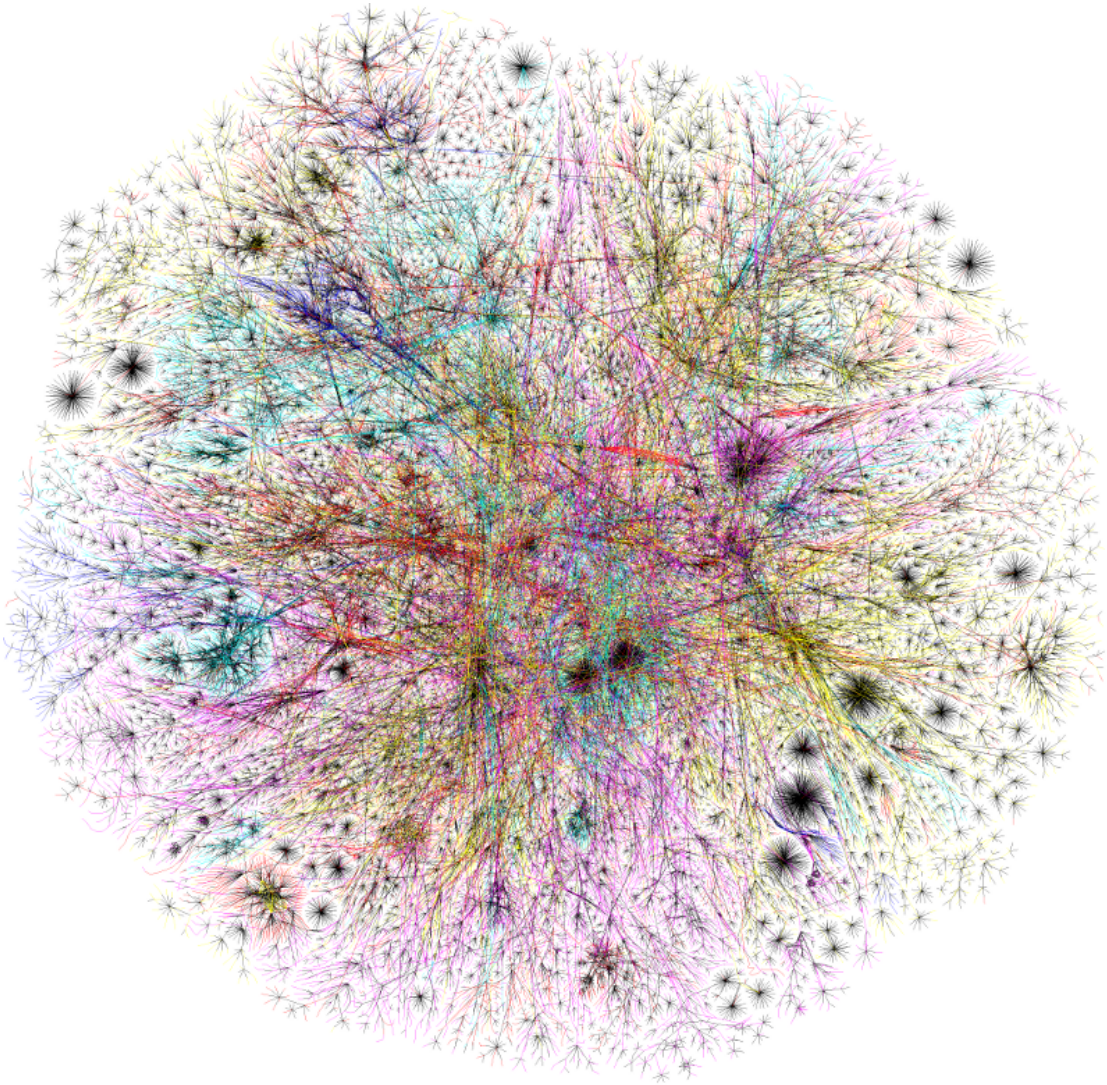

Figure 1.2. The structure of the Internet as connections between the "class A subnets", groups of computers with similar IP addresses, in 2003. The connections indicate the observed routes taken by data packages sent through the Internet and the colors indicate different groups of domain names. (Image: The Opte Project / Barrett Lyon)

superordinated organization or central authority and there is no "official map" of the Internet. Instead, its topology has to be explored by experimental measurement just like in any natural system. Figure 1.2 shows the connectivity structure of the Internet derived from such a measurement. Not to be confused with the Internet must be the *World Wide Web*, a network of web pages connecting to one another via hyperlinks without any physical structure. Just as in the case of the Internet, the structure of the World Wide Web has to be measured experimentally. Another class of network systems our daily live depends on crucially are *supply networks* like power grids and the supply infrastructure for gas, water, oil, etc. In this context, mostly the large-scale distribution networks such as the high-voltage transmission lines for the long-distance transport of electrical energy are considered. These grids are usually managed and supervised by single authorities and complete maps are available. In this case, determining the topology is not a difficult task. Nevertheless, also power grids show complex structures and sometimes counterintuitive behavior. Finally in the group of technological systems, *transportation networks* form a very important part of our modern infrastructure. Obviously, roads and railway lines form networks connecting cities and railway stations. Although these networks are restricted to exist on a plane (except for tunnels and bridges) they exhibit complex structures. Networks of airplane or ship connections, on the other hand, are not that strictly bound to any man-made connecting infrastructure.

A third realm in which network models are applied are the social and economic sciences. For most people the term might be associated with online social networking platforms like Facebook but the notion of a *social network* was introduced long before the Internet even existed. In fact, sociology may actually have the longest tradition in quantitative, empirical work on network systems. The concept of sociograms—a network representation of social relations in a group—even dates back to the 1930s (Moreno, 1934). In general, a social network describes a set of people and some relation like friendship, partnership, sexual contacts, or professional relations between them. The size of such networks may range from small groups like the 34 members of a karate club (one of the most famous examples in the literature, the network structure is shown in figure 1.3) to possibly all human beings on the earth. The collection of data for these networks by traditional methods like interviews and questionnaires was quite cumbersome and restricted to smaller communities. Today, much larger datasets exist—but might not be publicly accessible—for online networking platforms. A special case of social networks are *collaboration networks* recording the participation of people in joint projects like actors appearing in the same movie or scientists co-authoring an article. Large datasets for these networks can be extracted from online databases such as the Internet Movie Database or the Web of Science. By means of the latter, also *citation networks* can be recorded, describing the citation relations between scientific publications. These relations are strictly directed, mutual relations are (almost) impossible since only previously published articles can be cited. In economic sciences, networks of business relations are studied. These might be *banking networks* describing interbank lending and therefore interdependencies of banks. In the study of markets, *trading networks* record trade relationships between companies, states, or whole economic zones.

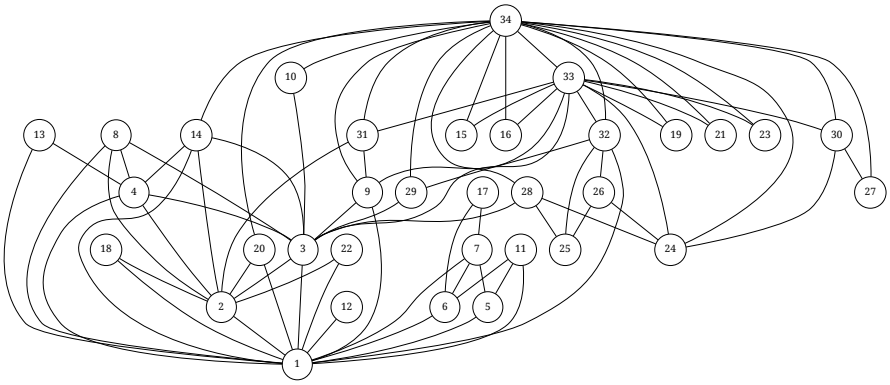

Figure 1.3. Zachary's karate club network. A network of friendship between the 34 members of a university karate club, recorded by Zachary (1977). By analyzing the structure of the network, Zachary was able to predict how the club divided in two groups, which was observed later on in a conflict between the instructor (number 1) and the administrator (number 34) of the club.

The various systems mentioned here are of completely different nature. The constituent elements and also the represented relations of one example might have nothing in common with those of another one. Most of the systems vary substantially in their sizes, in the way they formed, and how they evolve. However, if one solely focuses on the network representation and forgets about what this abstraction stands for some reappearing structural properties shared among many of those systems can be uncovered. This observation triggered the general interest in networks by themselves as objects of scientific studies. The hope to discover universal properties of networked systems is what made them especially interesting from the view of theoretical physics. In the most simple description, a network consists of a set of points connected by lines. In mathematical terms the points are referred to as *vertices*, the lines are called *edges*, and together they form a *graph*. The number of edges that are connected to a given vertex is called its *degree*. One observation in many networks is a rather broad distribution of the vertex degrees. This means that most elements have just a few connections while a small but significant number of elements have extremely many, the so-called "hubs". Specifically, such heavy tails are found if the degree distribution follows a power law. In this case, the network is often referred to as a *scale-free network*. Although already observed in the 1960s in the context of citation networks (de Solla Price, 1965), the phenomenon was popularized more recently by Barabási and Albert (1999). Another intriguing characteristic of many network systems is the *small-world effect*. The distance between two vertices in a network can be defined by the minimum number of edges that have to be traversed in order to reach one vertex from the other, the so-called geodesic distance. It has been observed that the mean geodesic distance between vertex pairs in many networks is surprisingly small, the networks form a "small world". This has been popularized as the "six degrees of separation" referring to the average number of

acquaintance connections necessary to link two randomly chosen people in the United States of America (Milgram, 1967). Although it might sound peculiar at first, the small-world property alone is also found in random networks, even in the most simple random graph models, the famous Erdős-Rényi networks (Erdős and Rényi, 1959). Only together with a high level of clustering, the tendency to find groups of densely interconnected vertices, it becomes a non-trivial and astonishing characteristic of many empirical networks. This has been addressed in a seminal work by Watts and Strogatz (1998), a second foundation of modern network theory.

Network Dynamics

Why do people study networks and complex systems? By themselves the complex structures found in real-world network systems might appear fascinating but in the end it is the understanding of how these systems function that drives the interest in network research. In most of the systems described above the pattern described by the network actually serves as substrate for a dynamical process taking place on top of it, one sometimes speaks of *dynamics on networks*. Each vertex in the network stands for a dynamical variable and edges describe which variables are influenced by the dynamical state of which other variables. In gene regulation networks, the vertices stand for the different genes and the dynamical variables are their expression levels. The expression of a certain gene dynamically regulates the expression of all other genes connected to it by an edge. Similarly, the dynamics of neural networks may be viewed. The neurons form the vertices of the network and their activities are the dynamical variables regulating each other via the synaptic connections. In food webs, the number of individuals of certain species is described by dynamical variables that change due the foraging interactions. A slightly different point of view can also be taken by considering the distribution of biomass as dynamical quantity and the network as describing its flow through the ecosystem. In supply networks, usually a single commodity has to be transported from sources to sinks. One can simply consider the amount of the commodity at a vertex as dynamical variable. This might, however, not be enough if more complicated processes are imposed by the special system. In AC power grids, for example, the voltage oscillations govern the dynamics of the whole system and Kirchhoff's laws have to be satisfied. In contrast to the single commodity case, in a transportation network different objects need to be transported to their individual targets. In this case, their individual positions in the network might be a better choice of dynamical variables. The Internet can also be considered as a transport network for data packages which are sent from one computer to another. On social networks, a large variety of dynamical processes is taking place which can be considered separately or in combination. Of high practical importance are epidemic spreading dynamics describing how infectious diseases spread from individual to individual through personal contacts. Also the spread of information, of fads, or of innovations in a society are dynamical processes on social networks. The term "viral phenomenon" has been adapted from epidemic spreading for these processes,

if the piece of information, the fad, or the innovation is passed on from one individual to the next without a central broadcaster, just like a contagious disease.

The local dynamical rules—the internal dynamics of an isolated node as well as the couplings—of these systems obviously differ a lot. Nevertheless, they share an important feature putting them in the focus of dynamical network research. Namely, their internal dynamical rules and local coupling can be considered as rather "simple". The complex dynamics of a system as a whole arise from the way they are coupled, i.e., the network structure. Concerning the practical question of how the dynamics of such a system can be controlled or at least influenced, often not much can be done about the local dynamics. Instead, the important question is how the network structure has to be altered in order to change (or preserve) the overall dynamics. In the case of epidemic spreading dynamics, for example, one might want to know which individuals should be removed from the process by vaccination in order to efficiently prevent the disease from spreading over the whole population. In the case of the Internet or other infrastructure networks usually the opposite case is desired, how can the whole system be kept running if some parts of it fail.

A second dynamical aspect of networks is the *evolution of network structure*, sometimes called dynamics "of" networks to distinguish it from the dynamics "on" networks mentioned above. An evolving network changes its structure in the course of time, i.e., vertices or edges are added to or removed from the network. Again, almost all of the exemplary systems mentioned above are actually evolving their structure in time. The technological networks are often expanded to increase their capacity or downsized to make them more cost efficient. Frequently, new computers are added to the Internet and others are removed, new websites appear and old ones are deleted from the servers of the World Wide Web, new streets and railways are built, new airline connections are established while others are closed down, old power plants are replaced by new ones. Also social networks are constantly changing, people die and others are born, every day we meet different people and new technological developments or changing habits allow us to interact with other people. In biological networks, biological evolution is one ubiquitous source of changes in the constituents and interaction patterns. Especially on molecular scales, genetic mutations can directly change protein sequences and therefore their structure and interactions with other proteins or modify the metabolic pathways of certain substances and, by this, the ability of an organism to make use of different nutrients. Additionally, there may exist other sources of structural changes in biological networks like the brain plasticity in neural networks or the invasion of alien species into an ecosystem.

In the case when both kinds of dynamics, dynamics on a network and network evolution, are to be considered, the typical time scales of the two processes become important. They may either be of the same order of magnitude meaning that the dynamical variables of the process change their state more or less at the same rate as the network structure is modified. In this case, one speaks of coevolutionary or adaptive networks (Gross and Blasius, 2008) since the network structure can adapt to the state of the dynamical variables. Or the two time scales are well separated meaning that the dynamical variables change many times and probably have enough time to reach an

attractor until the network structure changes. In this case it is reasonable to assume that the evolution "sees" the overall behavior or performance of the dynamical process instead of a single dynamical state. This is certainly the case in biological evolution where, for example, gene expression levels change many, many times during the lifetime of an organism until mutations might occur from one generation to the next. Also the dynamics of neuronal activity is much faster than the formation of synaptic connections. In most technological networks a similar situation is found. Data packages are sent through the Internet much faster than new computers are installed, trains and cars run faster than new streets or railways are built, and voltage oscillations in a power grid have obviously a higher frequency than the construction of new power plants and transmission lines. In social networks, the situation can be different. It is very likely that people change their behavior and the way they interact with other people when they become infected by a disease or when they change their political opinion. In these cases the interaction network adapts to the current dynamical state. A different situation might have to be considered on other scales of human interaction. To describe the global spreading of some disease one might use airports as network vertices and consider them as infected if a certain threshold of people in the city or region carry the infection. In this case, the airline connectivity will usually not adapt to the dynamical state and time scales are well separated again.

In this thesis, the relation between network structure and the global behavior of dynamical processes is studied. The focus lies on Laplacian dynamics, an important class of processes in which the time evolution is governed by the graph Laplacian operator. This class comprises very basic but highly relevant processes such as diffusion and random walks or synchronization of oscillators with many significant applications. Inspired by the evolution of dynamical networks under a separation of time scales, a method of evolutionary optimization of network structure for a prescribed global dynamical behavior is developed. The overall dynamical behavior is characterized by the eigenvalue spectrum of the graph Laplacian bridging between network structure and dynamics. By this, the dynamics do not have to be carried out explicitly. Related ideas were applied before in studies on optimal or pessimal network structures for the synchronizability of oscillator systems (Donetti *et al.*, 2005, 2006, 2008) and reconstruction of networks from their Laplacian spectra (Comellas and Diaz-Lopez, 2008; Ipsen and Mikhailov, 2002).

Thesis Outline

The remainder of this thesis is structured as follows. Chapter 2 introduces the mathematical concepts and tools for the description of complex networks. Observables to quantify topological characteristics are covered as well as random network models reproducing different structural properties. An overview of dynamical processes on networks is given with a focus on diffusion and random walks. Finally, the graph Laplacian as generator of an important class of dynamics is introduced together with its relation to structural and dynamical network properties. In chapter 3 the method of

evolutionary optimization of network structure for anomalous diffusion described by the spectral dimension is developed. The technique is successfully applied and the resulting network structures are analyzed. Chapter 4 investigates the influence of structural correlations on the dynamical behavior in the evolved networks. To this end, random networks are generated having the same correlation functions as the evolved networks and their Laplacian spectra are compared. In chapter 5 the evolutionary optimization is further restricted to find networks which are homogeneous in their local connectivity. The resulting evolved networks exhibit a high level of structural symmetry which is exploited in a systematic coarse-graining to find the underlying backbone structures and their relevance for the anomalous diffusion behavior. Chapter 6 summarizes the results of this thesis and discusses its findings. In the appendix, the algorithm applied for the random network generation is explained in detail and the numerical libraries which were used are listed.

2. Dynamical Networks

In this chapter, the mathematical concepts for the quantitative description of networks and their dynamics are introduced. The corresponding mathematical field is *graph theory*. Since this is a large field with a long history, only the concepts and results that are necessary for the understanding of this thesis are covered. More extensive information about the presented concepts and additional topics can be found in the literature (e.g. Barrat *et al.*, 2008; Newman, 2010).

2.1. The Structure of Complex Networks

A network, as already introduced above, consists of a set of basic entities and a collection of pairwise connections between them. In the language of graph theory the basic entities are called *vertices* or *nodes*. The connections are described as pairs of vertices and are called *edges* or *links*. Together, the vertex set $\mathcal{V}$ and the corresponding edge set $\mathcal{E}$ form a *graph* $G = \{\mathcal{V}, \mathcal{E}\}$. Two vertices that are connected by an edge are called *adjacent* or *neighboring*. Some authors distinguish between networks as objects of empirical observations and graphs as strictly mathematical concepts that can be models for these "real-world" networks. In this thesis, as in most physical literature, no such differentiation is made, the terms "graph" and "network" are used interchangeably. Several cases, or types of networks, have to be distinguished. The edges may be either directed or undirected, so that the edge set consists of either ordered or unordered pairs of vertices. In the former case, one speaks of a *directed network* or *directed graph*, shortly called *digraph*. Secondly, it may be allowed or forbidden to have *multiedges* (several edges between the same pair of vertices) so that the edge set contains an element more than once. Thirdly, it may be allowed or forbidden to have edges from a vertex to itself, so-called *self-edges* or *self-loops*. An undirected network which contains neither multiedges nor self-loops is referred to as a *simple network*.

2.1.1. Formal Description

Vertex and edge sets provide the defining representation of networks. There exist, however, a number of equivalent ways to describe a network formally. The vertices are most easily identified by their position in the vertex set, which can then be simply written as $\mathcal{V} = \{1, \ldots, N\}$ where $N = |\mathcal{V}|$ denotes the number of vertices in the network. In physics, N is called the size of the network although in graph theory the "size" of the graph is $|\mathcal{E}|$, the number of edges, whereas $|\mathcal{V}|$ is called the "order" of the graph. An edge between vertices i and j is then denoted as the pair (i, j), the edge is called *incident* to i and to j. The edge set is then the list of pairs $\mathcal{E} = \{(i_1, j_1), \ldots, (i_M, j_M)\}$ where

$M = |\mathcal{E}|$ is the total number of edges in the network. Such *edge lists* are sometimes used to represent a network on a computer. For most computational purposes, however, the *adjacency list* representation is more efficient. It associates with each vertex a list of all its adjacent vertices. In a directed network, only the outgoing or only the incoming connections are considered in the adjacency list. For analytical calculations, it is usually more convenient to work with matrices rather than with lists. The basic matrix representation of a network is the *adjacency matrix* $\boldsymbol{A}$. For a network with N vertices it is a $N \times N$ square matrix with elements

$$A_{ij} = \begin{cases} 1 & \text{if there is an edge between vertices } i \text{ and } j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

The adjacency matrix of an undirected network is symmetric, $A_{ij} = A_{ji}$. For directed networks, it has to be specified whether A_{ij} relates to an edge from i to j or from j to i. Both conventions are found in the literature and often the authors do not explicitly specify which definition they use. In this thesis, A_{ij} in a directed network will stand for edges from i to j, $i \to j$. Also networks with multiedges and self-loops can be uniquely described by an adjacency matrix. For this, A_{ij} simply has to be generalized to be the number of edges between the vertices i and j. A self-loop in an undirected network is most practically described by setting the corresponding diagonal element $A_{ii} = 2$, or $A_{ii} = 2n$ in the case of n self-loops at vertex i. This definition is consistent with the idea that every undirected network can be equivalently described as a directed network in which each undirected edge is replaced by two oppositely oriented directed edges between the same pair of vertices. A single directed self-loop at vertex i is then described by $A_{ii} = 1$. A second important matrix is the graph Laplacian which will play a major role in the section on dynamical processes. In the case of a simple network it is also uniquely defined and characterizes the network structure completely. Recall that the *degree* k_i of a vertex i is the number of edges incident to it. In terms of the adjacency matrix it can be calculated as $k_i = \sum_j A_{ij}$. The *graph Laplacian* $\boldsymbol{L}$ is another $N \times N$ matrix defined as

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}\,. \tag{2.2}$$

Here, $\boldsymbol{D}$ is the diagonal $N \times N$ matrix of vertex degrees, $D_{ij} = k_i \delta_{ij}$, where δ_{ij} denotes Kronecker's delta, $\delta_{ii} = 1$ and $\delta_{ij} = 0$ if $i \neq j$. Since the diagonal elements of the adjacency matrix are zero for a simple network, the elements of the graph Laplacian simply read

$$L_{ij} = \begin{cases} k_i & \text{if } i = j, \\ -1 & \text{if there is an edge between vertices } i \text{ and } j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

The graph Laplacian $\boldsymbol{L}$ is sometimes called "algebraic Laplacian" in order to distinguish it from slightly different matrices which are sometimes also called Laplacian. The "random walk" *normalized Laplacian* $\boldsymbol{L'}$ has elements

$$L'_{ij} = \begin{cases} 1 & \text{if } i = j, \\ -\frac{1}{k_i} & \text{if there is an edge between vertices } i \text{ and } j, \\ 0 & \text{otherwise} \end{cases} \tag{2.4}$$

whereas the "symmetric" *normalized Laplacian* $\boldsymbol{L''}$ is defined by

$$L''_{ij} = \begin{cases} 1 & \text{if } i = j, \\ -\frac{1}{\sqrt{k_i k_j}} & \text{if there is an edge between vertices } i \text{ and } j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

Throughout this thesis, only the "algebraic" Laplacian $\boldsymbol{L}$ is called graph Laplacian, Laplacian matrix, or simply Laplacian. The others will always be specified as normalized Laplacians. For completeness, also the *incidence matrix* $\boldsymbol{B}$ shall be mentioned here as another matrix describing a network's structure. It is an $N \times M$ matrix in which each row represents a vertex and each column an edge of the network. For a directed network its elements are defined as

$$B_{ij} = \begin{cases} -1 & \text{if edge } j \text{ starts at vertex } i, \\ +1 & \text{if edge } j \text{ ends at vertex } i, \\ 0 & \text{otherwise.} \end{cases} \tag{2.6}$$

An undirected network first has to be given an orientation by arbitrarily assigning a direction to each edge. Then, the incidence matrix is constructed as above. The incidence matrix provides a second way to define the graph Laplacian as

$$\boldsymbol{L} = \boldsymbol{B}\,\boldsymbol{B}^{\mathrm{T}} \tag{2.7}$$

where $\boldsymbol{B}^{\mathrm{T}}$ is the matrix transpose of $\boldsymbol{B}$. Note that for an originally directed network, this definition provides the graph Laplacian of the underlying undirected network, i.e., the network obtained by taking the same vertex set and introducing an undirected edge between each pair of vertices where a directed edge in either or both directions was present.

Some additional notions from graph theory will become important later on in this thesis. A *subgraph* of a given graph is another graph that consists of a subset of the original vertex set and all edges from the original edge set which connect vertices that are both in this subset. A simple graph with N vertices in which all possible edges are present is called a *complete graph*. It has $M = N(N-1)/2$ edges. A *path* connects two vertices of a graph through intermediate vertices by traversing edges. Formally, it is a sequence of vertices in which each pair of consecutive vertices is connected by an edge. The length of a path is the number of traversed edges. A *geodesic path* between

two vertices is a shortest path connecting those vertices. The length of a geodesic path is called the *geodesic distance*, *chemical distance*, or simply *distance* ℓ of the two vertices. A graph is said to be *connected* if every pair of vertices can be connected by a path. Otherwise the graph is *disconnected*. A *connected component*, sometimes just called *component*, of a graph is a maximal connected subgraph, i.e., a subgraph which is connected and to which no additional vertex from the original graph can be added without loosing this property. Here, when speaking of networks all these concepts will be also used with "graph" in their names replaced by "network".

2.1.2. Basic Observables

Some standard observables to quantify and compare network structures have been established during the last years. The probably most prominent feature of a network is the *degree distribution* $P(k)$, the distribution of the degrees k_i over all vertices $i \in \mathcal{V}$. $P(k)$ represents the probability that a randomly chosen vertex has degree k. To be distinguished is the *edge end (degree) distribution* $P_\mathrm{e}(k)$. $P_\mathrm{e}(k)$ is the probability that the vertex at one of the two ends of a randomly chosen edge has degree k. As the probability for a vertex to be selected in such a process is proportional to its degree, the edge end distribution is related to the degree distribution by

$$P_\mathrm{e}(k) = \frac{k}{\langle k \rangle} P(k) \tag{2.8}$$

where $\langle k \rangle$ is the average vertex degree,

$$\langle k \rangle = \frac{1}{N} \sum_{i \in \mathcal{V}} k_i = \sum_k k\, P(k)\,. \tag{2.9}$$

Since $P_\mathrm{e}(0) = 0$ always holds the edge end distribution does not capture vertices of degree zero which then have to be described separately.

The degree distributions are one-point properties of a network. They only capture properties of single vertices, namely their degree, and do not relate them to their surroundings. The next step in a systematic treatment is to consider two-point correlations between vertex degrees. The *joint degree distribution* $P(j,k)$ describes these correlations. $P(j,k)$ is the probability that a randomly chosen edge connects vertices with degrees j and k. It is symmetric under the exchange of arguments, $P(j,k) = P(k,j)$. The edge end distribution can be directly recovered by marginalization over one of the arguments,

$$P_\mathrm{e}(k) = \sum_j P(j,k)\,. \tag{2.10}$$

If the vertex degrees are independent then the joint degree distribution factorizes as $P(j,k) = P_\mathrm{e}(j)\, P_\mathrm{e}(k)$. This is used to quantify the overall degree-degree correlations of a network in a single scalar. The *assortativity coefficient* r, sometimes called *Newman*

factor, is the Pearson correlation coefficient of the degrees of adjacent vertices (Newman, 2002, 2003a),

$$r = \frac{1}{\sigma_{\mathrm{e}}^2} \sum_{j,k} jk \left[P(j,k) - P_{\mathrm{e}}(j) P_{\mathrm{e}}(k) \right] . \tag{2.11}$$

The normalization $\sigma_{\mathrm{e}}^2 = \sum_k k^2 P_{\mathrm{e}}(k) - [\sum_k k P_{\mathrm{e}}(k)]^2$ is the variance of k with respect to the edge end distribution $P_{\mathrm{e}}(k)$ and guarantees that r is in the interval $[-1, 1]$. If $r = 0$ the degrees of neighboring vertices are uncorrelated and the network is said to be non-assortative. If $r > 0$ ($r < 0$) the degrees of neighboring vertices are positively (negatively) correlated and the network is said to be (dis)assortative. All the observables introduced so far have been defined for undirected networks but their generalizations for directed networks are straightforward.

In the next step, three-point correlations are considered. A full systematic treatment of three-point correlations between vertex degrees is, however, rather involved. Instead, these are usually subsumed as clustering coefficients. The term clustering refers to the tendency to find clusters of densely connected vertices observed in many empirical networks. One way to characterize this is to analyze the level of transitivity in a network, i.e., the tendency that two neighbors of a vertex are also connected to each other. The (global) *clustering coefficient* C quantifies this as the density of triangles in the network,

$$C = \frac{3 \times (\text{number of triangles})}{(\text{number of connected triples})} . \tag{2.12}$$

In this definition, a connected triple consists of three vertices $u, v, w \in \mathcal{V}$ with at least two edges $(u, v), (v, w) \in \mathcal{E}$, and a triangle is a fully connected triple which additionally has the third edge $(u, w) \in \mathcal{E}$. The factor 3 appears since each triangle consists, according to this definition, of three connected triples. C takes values between 0 (no transitivity, no triangles in the network) and 1 (complete transitivity, all components are complete subgraphs). One can also consider the clustering as a local property at a single vertex i by relating the number of edges between neighbors of i, T_i, to its maximally possible value $k_i(k_i - 1)/2$. This defines the *local clustering coefficient* C_i,

$$C_i = \frac{2T_i}{k_i(k_i - 1)} . \tag{2.13}$$

Note that T_i is also the number of triangles of which vertex i is a part. This clustering coefficient was introduced by Watts and Strogatz (1998) where they also proposed its average over all vertices

$$\langle C_i \rangle = \frac{1}{N} \sum_{i \in \mathcal{V}} C_i \tag{2.14}$$

as measure for the global clustering of a network. The average local clustering coefficient lies between 0 and 1 as well but is generally different from the global clustering coefficient defined above, $\langle C_i \rangle \neq C$. In order to relate the clustering effect to the degree, a *degree-*

dependent clustering coefficient $C(k)$ can be constructed by averaging the local cluster clustering coefficient over all vertices with the same degree k (Vázquez *et al.*, 2002),

$$C(k) = \frac{1}{N_k} \sum_{\{i \in \mathcal{V} | k_i = k\}} C_i \,, \tag{2.15}$$

where $N_k = NP(k)$ is the number of vertices with degree k.

A different approach to study a network structure is to systematically analyze its substructures from small building-blocks up to their large-scale arrangement. Small frequently reappearing subgraphs are called *network motifs* (Milo *et al.*, 2002). A network can be scanned for these motifs recording the numbers of their occurrences. The smallest non-trivial subnetworks consist of 3 nodes, but also larger motifs can be traced systematically. The frequency of a motif is usually compared to its frequency in a randomized network in order to see if it is statistically over- or underrepresented in the system under study. These statistics provide a different classification of networks and might reveal useful information about the various functions in a complex system or how it evolved.

The small-world effect mentioned above can also be translated into a more rigorous mathematical statement. A network is said to have the *small-world property* if the geodesic distance ℓ averaged over all pairs of vertices grows logarithmically or more slowly with the size N,

$$\langle \ell \rangle \sim \log N \,. \tag{2.16}$$

In many empirical networks, this definition cannot be applied, since they only exist with one given size. In such a case, one considers the number of vertices with distance ℓ or less from a given vertex, $M(\ell)$. In a regular lattice, this number is expected to increase as a power law with ℓ, while in a small-world network it grows exponentially or faster.

2.2. Network Models

The study of empirical networks and the analysis of their topology reveals many structural features of various network systems. This knowledge, on the one hand, allows for a classification of networks by their structural properties. On the other hand, these features provide a data basis for the search of structural properties that shape the behavior of dynamical processes and ensure the functionality of network systems. There is, however, one pitfall. By just working with the empirical data one can never be sure if a given property is a special characteristic of the system or just a product of either randomness or some external constraint. It is therefore very useful to have null models available that reproduce certain structural properties but are apart from that completely random. For this reason, a large part of network research is devoted to the development of network models and to the study of their properties. A random network model always defines an ensemble of networks by a probability distribution $P(G)$ over a set of networks $\{G\}$. The properties of a network model are then the properties of the ensemble and not the properties of a single realization. For the modeling of macroscopic

systems it is often convenient to consider a "thermodynamic limit" $N \to \infty$. As usual when dealing with infinities, all kinds of divergences and counterintuitive behavior are observed in infinite graphs. To avoid such problems only a restricted class of infinite networks are considered. Such "physical graphs" are characterized by a limited growth with bounded maximum degree and at most polynomial growth in the number of vertices. An advantage of working with infinite networks is that there is no distinction between a single realization and the ensemble. Also, singularities and universal asymptotic behavior can only be observed in infinite systems (Burioni and Cassi, 1996, 2005).

The notion of *random graphs* and also their extensive study is mostly attributed to Paul Erdős and Alfréd Rényi who published a series of papers on those models in the beginning of the 1960s (Erdős and Rényi, 1959, 1960, 1961). A similar model was introduced as "random net" about a decade earlier by Solomonoff and Rapoport (1951) and already related to neural networks and epidemic spreading. Consider a simple graph with N vertices and M edges randomly chosen out of the $N(N-1)/2$ possible edges. This defines the random graph $G(N, M)$. A very similar model is the random graph $G(N, p)$ in which each possible edge is present with probability p and absent with probability $1-p$. In the limit of large N the two models are statistically equivalent. With an average vertex degree

$$\langle k \rangle = \frac{2M}{N} = (N-1)p \tag{2.17}$$

their degree distribution tends to a Poisson distribution,

$$P(k) = \mathrm{e}^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!} \,, \tag{2.18}$$

which is the reason why they are sometimes called Poisson random graphs to distinguish them from other random graph models. In recent network literature also the terms "Erdős-Rényi model" or "Erdős-Rényi network" are commonly used. Apart from the rather narrow light-tailed degree distribution, Erdős-Rényi networks have a small clustering coefficient which vanishes in the limit of large N,

$$C = \frac{\langle k \rangle}{N-1} \to 0 \quad \text{for } N \to \infty \,, \tag{2.19}$$

and an average geodesic distance $\langle \ell \rangle$ growing logarithmically with the network size,

$$\langle \ell \rangle \sim \log N \,, \tag{2.20}$$

making them "small worlds". Since all vertices in an Erdős-Rényi random graph are equivalent by construction there are no degree-degree correlations. So, Erdős-Rényi model networks have the small-world property like many real-world networks but can never reproduce heavy-tailed degree distributions, degree-degree correlations, or a given clustering.

Regular lattices have a clustering coefficient which is well defined and easily controllable by the local structure. The average number of vertices within distance ℓ of a vertex $M(\ell)$, however, grows as a power law making them "large worlds". By

interpolation between the extremes of a clustered regular lattice and a small-world random graph Watts and Strogatz (1998) set up a model which combines both properties. The *small-world model*, or Watts-Strogatz model, starts with a circle of N vertices which are connected to their $2m$ nearest neighbors. In a second step, each edge is with probability p "rewired", i.e., disconnected from one or both of its vertices and then reconnected to randomly chosen vertices, or left in place with probability $1 - p$. For a large parameter region, $1/N \ll p \ll 1$, the resulting networks share both properties, a high clustering coefficient and the small-world effect.

Another way to extend the random graph model in order to better resemble the structures found in empirical networks is to accommodate degree distributions which are different from the Poisson distribution. The *configuration model* (Molloy and Reed, 1995) is a generalized random graph model accomplishing this task. The starting point is a degree sequence $k_1, \ldots, k_N$. If a degree distribution $P(k)$ instead of a single sequence is to be sampled, the $\{k_i\}$ can be simply drawn from that distribution in advance. The degrees are assigned to the vertices of the network such that the ith vertex has k_i dangling half-edges or "stubs" attached to it. These stubs are then randomly matched and connected to form the edges of the network. By this procedure, networks with the given degree sequence are generated that are completely random in every other respect. It was shown that, in the limit of infinite size, the configuration model samples all networks with the given degree sequence with equal probability (Molloy and Reed, 1995). The algorithm can also be used to construct networks of finite size $N < \infty$. There are, however, some subtleties to be taken into account. First of all, the degree sequence has to be graphical, i.e., there has to exist at least one graph with that degree sequence. Luckily, it is known from graph theory that as long as the sum of all degrees is an even number (obviously, $\sum_i k_i = 2M$ has to be even in any graph) this is satisfied by some multigraph with self-loops. Secondly, by matching the stubs completely at random, in general, a multigraph with self-loops will be generated. If the goal is to sample simple networks, one has to restart the whole matching procedure whenever a multiedge or self-loop is generated. A simple rejection of such an edge would introduce a bias into the sampling. Since, for some sequences, this can lead to very high and uncontrolled rejection rates, more sophisticated algorithms were proposed for an unbiased sampling (Blitzstein and Diaconis, 2011; Del Genio *et al.*, 2010). Nevertheless, due to its simplicity the configuration model is widely used to generate random networks. It has also been extended in various ways to accommodate other measures in addition to the degree distribution. An extension of the configuration model to create correlated and clustered random networks will be discussed and applied in chapter 4.

A completely different class of random networks is formed by growth models. These models do not uniformly sample networks with a given set of properties but rather focus on the process of network formation. The study of how the history and formation principles of such a process shape the final outcome shall provide possible explanations for certain properties observed in empirical networks. In a growing network, vertices are added steadily and connected to the existing ones following certain rules. In this simple setting, the growth process will obviously not converge. A network generated by such a growth process is just a snapshot taken after some "growth time". Therefore, the

distribution of such networks will certainly not be an equilibrium distribution and models of network growth are non-equilibrium models. The first model aiming to explain power laws in the degree distribution of citation networks was introduced by de Solla Price (1976). At that time, it was already known that "rich-get-richer" phenomena can lead to power-law distributions in other contexts. In *Price's model*, every time a new paper (vertex) is added to the network it cites (is linked to) on average m of the existing ones. Those are chosen with a probability proportional to their in-degree k (plus one to give also non-cited papers finite probability) such that vertices with a high in-degree have a "cumulative advantage". This results in an in-degree distribution with a power-law tail of the form

$$P(k) \sim k^{-(2+1/m)} \,. \tag{2.21}$$

The effect was rediscovered some decades later by Barabási and Albert (1999) who called it "preferential attachment". The *Barabási-Albert model* is very similar to Price's model but for undirected networks. In each time step, a vertex is added to the network and connected to m existing vertices which are selected with a probability proportional to their degree k (no additive constant is needed here). Independently of m, this generates networks with a power-law tail with exponent $\gamma = 3$ in the degree distribution,

$$P(k) \sim k^{-3} \,. \tag{2.22}$$

Although these networks are not exactly in the focus of this thesis, the growing networks are an important example of network evolution. They are not "typical realizations" of scale-free networks but, like most real-world systems, shaped by the non-equilibrium process of their generation (Callaway *et al.*, 2001). In the real world, a second mechanism, the selection pressure on a system depending on its functional performance, will additionally shape the evolution of many network systems.

2.3. Dynamical Processes on Networks

The study of dynamical processes on networks is an even more diverse topic than the characterization of structural network properties. For the investigation of the relation between structure and dynamics in networks, a broadly applicable characterization of dynamical processes on networks would be highly desirable. Although there have been attempts to categorize network dynamics and find stereotypical models for those categories (e.g. Barzel and Barabási, 2013), a widely accepted general classification scheme is still missing.

In general, a dynamical process on a network with N vertices is a N-dimensional dynamical system. The state of the system at time t is described by a vector $\boldsymbol{x}(t) = (x_1(t), \ldots, x_N(t))$ where each $x_i(t)$ characterizes the state of vertex i at time t. The time evolution of $x_i(t)$ is determined by its own value and the values $x_j(t)$ of the adjacent vertices j only. The network structure describes this interaction pattern. The variable x_i itself is usually a real or complex number or an element of some discrete set, but can also be a more complicated mathematical object. The first distinction between dynamical

processes is whether time is a discrete or a continuous variable. In the case of discrete time $t \in \mathbb{N}$, the dynamics are governed by an equation of the form

$$x_i(t+1) = \hat{f}_i\left(x_i(t), \{x_j(t) \mid j \in \mathcal{N}_i\}\right) . \tag{2.23}$$

Here, the function $\hat{f}_i$ describing the time evolution of vertex i consists of an internal part depending on $x_i(t)$ and an external part depending on the $x_j(t)$ from $\mathcal{N}_i$, the set of all neighbors of i. In the most simple setting, the x_i are just binary variables (describing "on" and "off" states). Such *Boolean networks* were introduced by Kauffman (1969) to model genetic networks. Boolean networks are the most prominent example of time- and state-discrete dynamical networks exhibiting a rich and well-studied behavior (Bornholdt, 2001; Drossel, 2008). The case of continuous time $t \in \mathbb{R}$ and real-valued state variables $x_i \in \mathbb{R}$ on an undirected network is generally described by a differential equation of the form

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = f_i(x_i(t)) + \sum_{j \in \mathcal{V}} A_{ij} g_{ij}(x_j(t), x_i(t)) . \tag{2.24}$$

In this case, the function f_i specifies the intrinsic dynamics of vertex i—as if it was isolated from all other vertices—and the g_{ij} describe the influence of the neighbors j on the dynamics of i. In the most common case, the model system consists of basically identical units which are all coupled in the same way. This means that the functions f_i, and the g_{ij}, respectively, are all identical $f_i \equiv f$, and $g_{ij} \equiv g$. Many important processes, such as synchronization and spreading dynamics, fall into this class of dynamics.

The study of *synchronization phenomena* on networks has attracted much attention in the scientific community. The problem can be formulated in relatively simple terms and, at the same time, has applications to a broad range of relevant research questions (Arenas *et al.*, 2008). AC power grids, earthquakes, economic cycles, neuronal activity during an epileptic seizure, flashing of fireflies, or oscillations in predator-prey dynamics are some examples where synchronization plays an important role. The paradigmatic model for the synchronization of non-identical oscillators with non-linear interactions is the Kuramoto model, originally formulated with all-to-all couplings. On a network, consider a planer rotor with phase variable ϕ_i and angular frequency ω_i at each vertex i. With sinusoidal couplings of strength K along the edges of the networks, the time evolution is described by

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = \omega_i + K \sum_{j \in \mathcal{V}} A_{ij} \sin(\phi_j(t) - \phi_i(t)) . \tag{2.25}$$

For small differences in the phase variable, linearization of the coupling term by $\sin x \approx x$ leads to

$$K \sum_j A_{ij}(\phi_j(t) - \phi_i(t)) = K \sum_j A_{ij}\phi_j(t) - k_i\phi_i(t) = K \sum_j \left(A_{ij} - k_j\delta_{ij}\right)\phi_j . \tag{2.26}$$

If additionally all the natural angular frequencies are the same, $\omega_i \equiv \omega$, one can transform to a co-moving frame $\phi_i \to \theta_i = \phi_i - \omega$. The linearized system of differential equations reads then

$$\frac{\mathrm{d}\theta_i}{\mathrm{d}t} = K \sum_{j \in \mathcal{V}} (A_{ij} - k_j \delta_{ij})\, \theta_j = -K \sum_{j \in \mathcal{V}} L_{ij} \theta_j \,. \tag{2.27}$$

Here, the graph Laplacian $\boldsymbol{L}$ appears as time evolution operator of the linearized oscillator dynamics.

In the general case of equation (2.24) with identical intrinsic dynamics f and couplings g, a similar equation is the result of linearization around a fixed point $\{x_i^*\}$. Writing $x_i(t) = x_i^* + \epsilon_i(t)$ and performing a Taylor expansion around the fixed point up to linear order in the $\{\epsilon_i(t)\}$, the linearized equation

$$\frac{\mathrm{d}\epsilon_i}{\mathrm{d}t} = \Big[\alpha_i + \sum_j \beta_{ij} A_{ij}\Big] \epsilon_i(t) + \sum_j \gamma_{ij} A_{ij} \epsilon_j(t) \tag{2.28}$$

is obtained. The constants $\alpha_i, \beta_{ij}, \gamma_{ij}$ are the first order derivatives of f and g at the fixed point,

$$\alpha_i = \left.\frac{\partial f}{\partial x}\right|_{x=x_i^*} , \quad \beta_{ij} = \left.\frac{\partial g(u,v)}{\partial u}\right|_{u=x_i^*, v=x_j^*} , \quad \gamma_{ij} = \left.\frac{\partial g(u,v)}{\partial v}\right|_{u=x_i^*, v=x_j^*} . \tag{2.29}$$

The linearized equation (2.28) can be written in matrix form as

$$\frac{\mathrm{d}\boldsymbol{\epsilon}}{\mathrm{d}t} = \boldsymbol{T}\, \boldsymbol{\epsilon}(t) \,, \tag{2.30}$$

where $\boldsymbol{\epsilon}(t)$ is the vector whose components are $\epsilon_i(t)$. The time evolution operator is the matrix $\boldsymbol{T}$ with elements

$$T_{ij} = \Big[\alpha_i + \sum_l \beta_{il} A_{il}\Big] \delta_{ij} + \gamma_{ij} A_{ij} \,. \tag{2.31}$$

The equation can be solved by expanding $\boldsymbol{\epsilon}(t)$ in terms of the right eigenvectors $\boldsymbol{v}_r$ of $\boldsymbol{T}$,

$$\boldsymbol{\epsilon}(t) = \sum_r c_r(t)\, \boldsymbol{v}_r \,. \tag{2.32}$$

Since $\boldsymbol{T}\boldsymbol{v}_r = \lambda_r \boldsymbol{v}_r$, equation (2.30) transforms into equations for the expansion coefficients $c_r(t)$,

$$\frac{\mathrm{d}c_r}{\mathrm{d}t} = \lambda_r c_r(t) \,, \tag{2.33}$$

which can be solved easily by

$$c_r(t) = c_r(0)\, \mathrm{e}^{\lambda_r t} \,. \tag{2.34}$$

By this, the eigenvectors $\{\boldsymbol{v}_r\}$ and eigenvalues $\{\lambda_r\}$ of the time evolution operator completely determine the behavior of any linear (or linearized) dynamical system on a network. Note how both, the rules of the dynamical process, as described by the

functions f and g, and the network structure, in form of the adjacency matrix $\boldsymbol{A}$, are encoded in the matrix $\boldsymbol{T}$ and, thus, in its eigenvalues and eigenvectors. Following the picture induced by equation (2.32), the resulting dynamics can be viewed as being composed of different modes corresponding to the eigenvectors $\boldsymbol{v}_r$. For the stability of such a dynamical system—or its fixed points, in the case of linearization—the signs of the respective eigenvalues λ_r play a major role. Only if the real parts of all eigenvalues are smaller or equal to zero the system—or the fixed point under consideration—is stable. All modes corresponding to eigenvalues with negative real parts decay exponentially. Only those which are zero or purely imaginary survive in the long-time limit, resulting in a constant or oscillatory state.

Consider now the special case when the coupling has the form of a difference between the values of a single function h of x_i and x_j individually, i.e., $g(x_i, x_j) = h(x_i) - h(x_j)$. If additionally the fixed point is symmetric, $x_i^* \equiv x^*$, the matrix $\boldsymbol{T}$ can be further analyzed. The constants defined in equation (2.29) now take the simpler form

$$\alpha_i \to \alpha = \left.\frac{\partial f}{\partial x}\right|_{x=x^*}, \quad \beta_{ij} \to \beta = \left.\frac{\partial h(x)}{\partial x}\right|_{x=x^*}, \quad \gamma_{ij} \to \gamma = -\beta\,, \tag{2.35}$$

such that the elements of the time evolution operator $\boldsymbol{T}$ read

$$T_{ij} = (\alpha + \beta k_i)\delta_{ij} - \beta A_{ij} = \alpha\delta_{ij} + \beta(k_i\delta_{ij} - A_{ij})\,. \tag{2.36}$$

Written in matrix form, equation (2.30) for this case becomes

$$\frac{\mathrm{d}\boldsymbol{\epsilon}}{\mathrm{d}t} = (\alpha\boldsymbol{I} + \beta\boldsymbol{L})\,\boldsymbol{\epsilon}(t)\,, \tag{2.37}$$

where $\boldsymbol{I}$ is the identity matrix with elements $I_{ij} = \delta_{ij}$. Again, the graph Laplacian $\boldsymbol{L}$ appears as (essential part of the) time evolution operator in the linearized dynamics. As in the case of the linearized oscillator dynamics, its eigenvalues determine the dynamical behavior. In particular, the fixed point can be stable only if for all eigenvalues λ_r of the graph Laplacian

$$\alpha + \beta\lambda_r \leq 0\,. \tag{2.38}$$

Since $\lambda_1 = 0$ is always a Laplacian eigenvalue and all eigenvalues are non-negative (this will be derived in section 2.4 below), necessary conditions for the stability are $\alpha \leq 0$ and $\lambda_N \leq -\alpha/\beta$ with λ_N denoting the largest Laplacian eigenvalue.

2.3.1. Diffusion Dynamics, Random Walks, and Graph Laplacians

A particularly simple but generic class of dynamics is formed by diffusion processes. It is simple because the mathematical formulation of diffusion on a network is as basic as it could be. It is a linear process, there are no intrinsic dynamics, $f(x) = 0$, and the coupling function is just the difference between the two variables, $g(x_i, x_j) = x_j - x_i$. On the other hand, diffusion is rather generic because it is a paradigmatic model for all kinds of spreading dynamics. Traditionally, diffusion in physics describes the movement of a gas from regions with high concentration to regions with low concentration or

along some gradient, for example in pressure or temperature. But also other types of spreading dynamics are modeled as diffusion processes. The spread of infectious diseases in a population, the transfer of information or fads from individual to individual without a common broadcaster, the change of opinions in a society, and the exploration of the World Wide Web by random searching are just some examples. Additionally, diffusion forms a basis of more complicated dynamics such as reaction-diffusion processes describing spatial aspects of chemical reactions (Burioni *et al.*, 2012). In the context of diffusion, the state variable $x_i(t)$ describes, for example, the amount of a substance at vertex i. The substance will flow along the edges from vertices with a high value to vertices with a lower value. Hence, $x_i(t)$ changes with time according to

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = c \sum_{j \in \mathcal{V}} A_{ij} \left(x_j(t) - x_i(t) \right) . \tag{2.39}$$

The constant c is a diffusion coefficient. Splitting the two terms in the sum as in equation (2.26) yields

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = -c \sum_{j} L_{ij} x_j(t) , \tag{2.40}$$

or, in matrix notation,

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} + c\boldsymbol{L}\boldsymbol{x}(t) = 0 . \tag{2.41}$$

This equation has the form of a continuum diffusion equation with the Laplace operator ∇^2 replaced by $-\boldsymbol{L}$ which is the reason why it is called the graph Laplacian.

A possible microscopic picture of diffusion are random walks. On a network, consider a particle that moves randomly along the edges from vertex to vertex in discrete steps. The diffusion process is recovered from the random walk by either considering a number of $n \to \infty$ random walkers and let $x_i(t)$ describe their density at vertex i or, equivalently, let $x_i(t)$ simply be the probability of a random walker to be at vertex i at time t. On a heterogeneous network, in which not all vertex degrees are the same, two types of random walks have to be distinguished. The total rate of escape from a given vertex equals the sum of the random walker's jump rates across all incident edges. For a "completely random" process either this total escape rate is set constant or the jump rate across each individual edge is set constant, resulting in two slightly different processes. In the latter case, the total escape rate from a vertex is proportional to the vertex degree and the random walker will, on average, spend less waiting time at high-degree vertices than at low-degree vertices. This describes the "diffusing substance" case where the total capacity of flux out of or into a vertex is also proportional to its degree. The time evolution is correctly described by equation (2.39). The first case corresponds to a random walker in discrete time that has to jump to the next vertex at each time step. Here, the jump rate across each edge is normalized by the degree of the vertex the random walker is currently occupying, i.e., the rate to jump from vertex i to vertex j is k_i^{-1}. The corresponding diffusion equation thus reads

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = c \sum_{j \in \mathcal{V}} A_{ij} \left(\frac{x_j(t)}{k_j} - \frac{x_i(t)}{k_i} \right) . \tag{2.42}$$

As before, the sum on the right hand side can be split,

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = c\sum_j A_{ij}\frac{x_j(t)}{k_j} - cx_i(t)\frac{k_i}{k_i} = c\sum_j \left(\frac{A_{ij}}{k_j} - \delta_{ij}\right) x_j(t) = -c\sum_j L'_{ij}x_j(t)\,. \quad (2.43)$$

In this case, the normalized Laplacian $\boldsymbol{L}'$ is the time evolution operator of the diffusion process which is obviously the reason why it is called the "random walk" normalized Laplacian.

2.3.2. Directed Graph Laplacian

Up to this point, the mathematical descriptions of dynamical processes were formulated for undirected networks. In most cases, the equations are straightforwardly generalized to dynamics on directed networks. Some care has to be taken with placing the indices of the adjacency matrix in the correct order. As noted before, here the convention is used that the matrix element A_{ij} stands for an edge from vertex i to vertex j, denoted by $i \to j$, which is often found in the mathematical literature.

While the adjacency matrix for directed networks is completely defined by this convention, for the graph Laplacian it is not obvious which numbers have to be put on the diagonal replacing the vertex degrees. Plausible alternatives could be the in-degrees, the out-degrees, or the sum or average of both, depending on what is to be described. In order to find the correct choice here, consider a diffusion process as formulated in equation (2.39). As before, let $x_i(t)$ be the amount of a diffusing substance at vertex i. The flux of the substance along an edge $i \to j$ shall be proportional to $x_i(t)$. (If the oppositely directed edge $j \to i$ is also present this results a net flux proportional to the difference $x_i(t) - x_j(t)$ as in the undirected case.) Then, $x_i(t)$ increases at rate $cx_j(t)$ for each incoming edge $j \to i$ and decreases at rate $cx_i(t)$ for each outgoing edge $i \to j$. The resulting coupled differential equations read

$$\begin{aligned}\frac{\mathrm{d}x_i}{\mathrm{d}t} &= \sum_{j\in\mathcal{V}} A_{ji}cx_j(t) - \sum_{j\in\mathcal{V}} A_{ij}cx_i(t) = c\sum_j A_{ji}x_j(t) - ck_i^{\mathrm{out}}x_i(t) \\ &= c\sum_j \left(A_{ji} - \delta_{ji}k_j^{\mathrm{out}}\right) x_j(t) = -c\sum_j L_{ji}^{\mathrm{out}}x_j(t) \end{aligned} \quad (2.44)$$

where $k_i^{\mathrm{out}} = \sum_j A_{ij}$ denotes the out-degree of vertex i and

$$L_{ji}^{\mathrm{out}} = k_j^{\mathrm{out}}\delta_{ji} - A_{ji} \quad (2.45)$$

are the elements of the out-degree Laplacian $\boldsymbol{L}^{\mathrm{out}}$. To express this in matrix notation, it is most convenient to consider vectors as row matrices. Then, the action of a matrix operator on a vector is calculated by the multiplication of the row vector from the left to the matrix, just as the sum in the last term of equation (2.44) suggests. In matrix form, the equation for directed diffusion thus reads

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = -c\,\boldsymbol{x}(t)\boldsymbol{L}^{\mathrm{out}}\,. \quad (2.46)$$

Concluding, in the setting of diffusion-like dynamical processes with the flux along edges being proportional to the state variable of the source vertex the out-degree Laplacian is the correct generalization of the graph Laplacian as time-evolution operator on directed networks.

2.3.3. Spectral and Other Dimensions

There are different ways to characterize the overall behavior of diffusion or random walks. Let $P_{ij}(t)$ denote the probability that a random walker starting from vertex i is at vertex j after t time steps. If r_{ij} denotes the (Euclidean) distance between vertices i and j then

$$R^2(t) = \left\langle R_i^2(t) \right\rangle = \left\langle \sum_j r_{ij}^2 P_{ij}(t) \right\rangle \tag{2.47}$$

is the *mean-square displacement* averaged over all starting vertices i. It characterizes the average spread of a diffusive process. Many networks are not naturally embedded in any Euclidean space. Not only in such a case but generally the displacement of a random walker can be measured in terms of the chemical distance ℓ_{ij} between two vertices i and j,

$$L^2(t) = \left\langle L_i^2(t) \right\rangle = \left\langle \sum_j \ell_{ij}^2 P_{ij}(t) \right\rangle . \tag{2.48}$$

A second measure is the probability $P_{ii}(t)$ that a random walker returns to its origin exactly at time t. The *average return probability* is defined as

$$P_0(t) = \langle P_{ii}(t) \rangle \tag{2.49}$$

and also called random walk autocorrelation function. A slightly different measure is the probability that a random walker has returned up to time t. To characterize this property, often the *number of distinct sites* visited after time t by a random walker starting from vertex i, denoted by $S_i(t)$, or its average $S(t) = \langle S_i(t) \rangle$ is used. On a finite network, there is always a finite probability for the random walker to visit any site (of the connected component it started in). So, in the limit of $t \to \infty$ the random walker will have returned to its starting vertex at some point in time with probability one. On an infinite network, this probability can be less than one and the random walk is called *transient*. Otherwise, it is *recurrent*.

The behavior of random walks on various types of lattices as well as deterministic and random structures has been studied extensively (ben-Avraham and Havlin, 2000). On regular lattices, many properties are independent of the specific lattice structure but depend only on the spatial dimension d. In an infinite lattice for $t \to \infty$ many quantities exhibit power-law dependencies with exponents that are independent of the local structures. The asymptotic behavior of the return probability and the number of visited sites are

$$P_0(t) \sim t^{-d/2} \tag{2.50}$$

and

$$S(t) \sim t^{\min(1,d/2)} \quad (\text{for } d \neq 2), \tag{2.51}$$

respectively. The mean-square displacement is even independent of d,

$$R^2(t) \sim t\,. \tag{2.52}$$

In disordered systems, this is not necessarily the case which led to the introduction of generalized dimensions. The *fractal dimension* d_f is not related to dynamics. As generalization of the spatial dimension it describes the scaling of the mass M of an object, i.e. the number of sites, with its linear extension l,

$$M(l) \sim l^{d_\mathrm{f}}\,. \tag{2.53}$$

The *random walk dimension* d_w is introduced to generalize the scaling of the mean-square displacement in equation (2.52),

$$R^2(t) \sim t^{2/d_\mathrm{w}}\,. \tag{2.54}$$

The *spectral dimension* d_s generalizes the scaling of the return probability in equation (2.50),

$$P_0(t) \sim t^{-d_\mathrm{s}/2}\,, \tag{2.55}$$

and also describes the scaling of the number of visited sites in equation (2.51),

$$S(t) \sim t^{\min(1,d_\mathrm{s}/2)} \quad (\text{for } d_\mathrm{s} \neq 2). \tag{2.56}$$

The three fractal dimensions are not independent but related (Havlin and ben-Avraham, 2002) by

$$d_\mathrm{s} = \frac{2d_\mathrm{f}}{d_\mathrm{w}}\,, \tag{2.57}$$

which agrees with the scaling behavior in regular d-dimensional lattices where they take the values $d_\mathrm{f} = d$, $d_\mathrm{w} = 2$, and $d_\mathrm{s} = d$.

The property of recurrence can be related to the spectral dimension by simply integrating equation (2.55). The integral

$$\int_{t_0}^{\infty} P_0(t)\mathrm{d}t \sim \int_{t_0}^{\infty} t^{-d_\mathrm{s}/2}\mathrm{d}t \tag{2.58}$$

diverges for $d_\mathrm{s} < 2$ and, hence, the random walk is recurrent. For $d_\mathrm{s} > 2$ the integral remains finite and the random walk is transient. For the marginal case $d_\mathrm{s} = 2$, the integral diverges logarithmically but sub-leading corrections to the power law may influence the random walk behavior (Burioni and Cassi, 2005).

As introduced above, all the scaling relations can also be set up in terms of the chemical distance instead of the Euclidean metric. The fractal dimension d_f and the random walk dimension d_w are then replaced by the connectivity dimension or fractal dimension in chemical space d_ℓ and the random walk dimension in chemical space $d_\mathrm{w}^{(\ell)}$, respectively,

$$M(\ell) \sim \ell^{d_\ell}\,, \quad L^2(t) \sim t^{2/d_\mathrm{w}^{(\ell)}}\,. \tag{2.59}$$

Since equation (2.57) is analogously valid, the spectral dimension in chemical space is the same as in Euclidean space (ben-Avraham and Havlin, 2000),

$$d_\mathrm{s} = \frac{2d_\mathrm{f}}{d_\mathrm{w}} = \frac{2d_\ell}{d_\mathrm{w}^{(\ell)}}\,. \tag{2.60}$$

Due to this property, the spectral dimension is the most convenient out of the three generalized dimensions to characterize networks that are not naturally embedded in some Euclidean space.

2.4. The Spectrum of the Graph Laplacian

The graph Laplacian was introduced in section 2.3 as time evolution operator of an important class of dynamical processes on networks. Its eigenvalue spectrum completely determines the dynamical behavior of such processes in terms of the Laplacian eigenvectors. But the Laplacian spectrum also encodes important structural properties of a network. It is, besides the spectrum of the adjacency matrix, the second central mathematical object studied in spectral graph theory (Chung, 1997). Much work has been dedicated to the understanding of its properties and its relations to the network structure.

First, recall equation (2.7) stating that the graph Laplacian can be written as matrix product between the incidence matrix and its transpose, $\boldsymbol{L} = \boldsymbol{B}\boldsymbol{B}^\mathrm{T}$. Let λ_i be an eigenvalue of $\boldsymbol{L}$ with corresponding normalized eigenvector $\boldsymbol{v}_i$, $\boldsymbol{L}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$. Then,

$$\lambda_i = \boldsymbol{v}_i^\mathrm{T}(\lambda_i \boldsymbol{v}_i) = \boldsymbol{v}_i^\mathrm{T}(\boldsymbol{L}\boldsymbol{v}_i) = \boldsymbol{v}_i^\mathrm{T}(\boldsymbol{B}\boldsymbol{B}^\mathrm{T}\boldsymbol{v}_i) = (\boldsymbol{v}_i^\mathrm{T}\boldsymbol{B})(\boldsymbol{B}^\mathrm{T}\boldsymbol{v}_i) = (\boldsymbol{B}^\mathrm{T}\boldsymbol{v}_i)^\mathrm{T}(\boldsymbol{B}^\mathrm{T}\boldsymbol{v}_i)\,, \tag{2.61}$$

which is nothing but the inner product of the real vector $\boldsymbol{B}^\mathrm{T}\boldsymbol{v}_i$ with itself. Hence, all eigenvalues of the graph Laplacian are non-negative,

$$\lambda_i \geq 0\,. \tag{2.62}$$

Since all row sums in the graph Laplacian are zero the constant vector $\mathbf{1} = (1, 1, \ldots, 1)$ is a Laplacian eigenvector with corresponding eigenvalue 0. So, the zero is always present in the Laplacian spectrum and the eigenvalues are usually numbered in ascending order,

$$0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N\,. \tag{2.63}$$

Some additional intuitively understandable as well as more involved properties are well known and proven (Mohar, 1997). The multiplicity of zero as an eigenvalue is equal to the number of connected components in the network. The second smallest eigenvalue λ_2 is referred to as the *algebraic connectivity*. λ_2 is non-zero if and only if the network consists of a single connected component and its magnitude is a measure for "how well" the network is connected. The smaller λ_2 the easier it is to disconnect some part of the network by the removal of edges. λ_2 is bounded from above by

$$\lambda_2 \leq \frac{N}{N-1} k_\mathrm{min} \tag{2.64}$$

where $k_{\min}$ is the minimum degree in the network. The largest Laplacian eigenvalue λ_N can be estimated by the maximum degree $k_{\max}$,

$$\frac{N}{N-1}k_{\max} \leq \lambda_N \leq 2k_{\max}\,. \tag{2.65}$$

The eigenvalue spectra of the normalized Laplacians $\boldsymbol{L}'$ and $\boldsymbol{L}''$ are the same but different from the spectrum of $\boldsymbol{L}$ (Samukhin *et al.*, 2008). Nevertheless, the two different spectra share a number of properties. The eigenvalues $\{\lambda'_i\}$ of the normalized Laplacian are also positive but bounded from above by 2. The smallest eigenvalue λ'_1 is always zero, hence, $0 = \lambda'_1 \leq \lambda'_2 \leq \cdots \leq \lambda'_N \leq 2$. Also in this case, the multiplicity of the zero eigenvalue equals the number of connected components in the network and the second smallest eigenvalue λ'_2 is called the algebraic connectivity—although it is, of course, different from λ_2. These are the most simple out of a large number of mathematical properties of the spectrum of the normalized Laplacian (Chung, 1997).

The Laplacian spectrum of a finite network is a set of N numbers. For a more easily tractable description and to make the spectra of different networks straightforwardly comparable it is better represented as a function. A natural choice for such a function is the *eigenvalue density*

$$\rho(\lambda) = \frac{1}{N}\sum_{i=1}^{N}\delta(\lambda - \lambda_i)\,, \tag{2.66}$$

where $\delta(\cdot)$ denotes the Dirac delta function. For a finite network, $\delta(\cdot)$ is usually replaced by smoothing kernel, for example a Gaussian distribution with finite width, in order to obtain a real-valued function which can be plotted and analyzed. By this, the spectra of individual realizations of finite networks can be compared among each other but also with infinite networks and network ensembles. Such *spectral plots* were used to characterize different classes of empirical networks and network models (Banerjee and Jost, 2008, 2009). The problem of smoothing can be avoided by working with the *integrated eigenvalue density* $I(\lambda)$ instead,

$$I(\lambda) = \int \mathrm{d}\lambda\,\rho(\lambda) = \frac{1}{N}\sum_{i=1}^{N}\Theta(\lambda - \lambda_i)\,, \tag{2.67}$$

where $\Theta(\cdot)$ denotes the Heaviside step function, $\Theta(x) = 1$ for $x \geq 0$ and $\Theta(x) = 0$ for $x < 0$. For a finite network, $I(\lambda)$ is a step function with discrete steps of size $1/N$ at each eigenvalue.

The spectrum of the graph Laplacian provides a second way to define the spectral dimension of a network by its asymptotic behavior for small λ. This is, obviously, the reason why the quantity is called "spectral". For an infinite network, the definition of the "vibrational" spectral dimension is given by the relation

$$I(\lambda) \sim \lambda^{d_{\mathrm{s}}/2} \quad \text{for } \lambda \to 0\,. \tag{2.68}$$

This definition of the spectral dimension is referred to as "vibrational" because it relates to the density of vibrational modes in a network of particles connected by harmonic

springs (ben-Avraham and Havlin, 2000). It is independent of the choice of the Laplacian or the normalized Laplacian,

$$\frac{d_\mathrm{s}}{2} = \lim_{\lambda\to 0}\frac{\log(I(\lambda))}{\log(\lambda)} = 1 + \lim_{\lambda\to 0}\frac{\log(\rho(\lambda))}{\log(\lambda)} = 1 + \lim_{\lambda'\to 0}\frac{\log(\rho(\lambda'))}{\log(\lambda')}\,, \tag{2.69}$$

and equal to the "random walk" spectral dimension defined in equation (2.55) for the "physical graphs" (Burioni and Cassi, 2005) as introduced in section 2.2. The average return probability is then related to the spectral density by a Laplace transform,

$$P_0(t) = \int_0^\infty \mathrm{d}\lambda \mathrm{e}^{-\lambda t}\rho(\lambda)\,. \tag{2.70}$$

A derivation of this relation can be found in Samukhin *et al.* (2008) and Barrat *et al.* (2008, appendix 5).

All real systems are of course finite. But a system might be large enough to be successfully described by an infinite model system. This is the case when the bulk properties dominate over the boundary properties, i.e., in the case of dynamics, the process does not reach the boundary. The long time limit in such a finite system actually refers to the period after the transient behavior—determined by the initial conditions and local structures—has ceased but before the process experiences the finite size. The "asymptotic" power-law scalings in time are then observed at these intermediate times. The simplest way to achieve such a situation is to have a power law at all times. Such a power law in the average return probability is generated by a power law in the spectral density. Assume

$$\rho(\lambda) \propto \lambda^{d_\mathrm{s}/2-1}\,, \tag{2.71}$$

then the average return probability can be calculated easily using equation (2.70) and the definition of the gamma function,

$$\Gamma(x) = \int_0^\infty \mathrm{e}^{-y}y^{x-1}\mathrm{d}y \overset{[y=\lambda t]}{=} t^x\int_0^\infty \mathrm{e}^{-\lambda t}\lambda^{x-1}\mathrm{d}\lambda\,, \tag{2.72}$$

to be

$$P_0(t) \propto \int_0^\infty \mathrm{d}\lambda \mathrm{e}^{-\lambda t}\lambda^{d_\mathrm{s}/2-1} = \Gamma\left(d_\mathrm{s}/2\right)t^{-d_\mathrm{s}/2} \propto t^{-d_\mathrm{s}/2}\,. \tag{2.73}$$

By these relations the Laplacian spectrum in general and in particular the spectral dimension build an important bridge between the structure and dynamics, in this case diffusion processes, on networks.

As the graph Laplacian appears as time-evolution operator in other dynamical processes as well, its spectrum and eigenvectors can also be useful for the characterization of their dynamical behavior. In oscillator dynamics, the overall synchronizability of a network was found to be determined by the ratio of the smallest (non-zero) eigenvalue to the largest eigenvalue λ_2/λ_N, the so-called *Laplacian eigenratio*. Due to the boundedness of λ_N, see equation (2.65), it was also argued that already λ_2 alone is the major factor for the synchronizability of an oscillator network (Arenas *et al.*, 2008). Historically, the

first definition of a spectral dimension was the "vibrational" spectral dimension defined in equation (2.69). The model system is a network of particles connected by harmonic springs. Such systems can be viewed as generalization of a bead-and-spring polymer model which is why they are called generalized Gaussian structures (Gurtovenko and Blumen, 2005). The density of vibrational modes with frequency ω is directly related to the Laplacian spectral density by $\rho(\lambda)\mathrm{d}\lambda = \rho(\omega)\mathrm{d}\omega$ with $\lambda = \omega^2$. Thus, if $\rho(\lambda) \propto \lambda^{d_\mathrm{s}/2-1}$ then $\rho(\omega) \propto \omega^{d_\mathrm{s}-1}$ (ben-Avraham and Havlin, 2000).

In the following chapters of this thesis, the argumentation will be focused on diffusion as basic dynamical process. One should, however, keep in mind the much more general importance of the Laplacian spectrum in network dynamics.

3. Network Evolution

This chapter describes how the strategy of evolutionary optimization can be applied to construct networks with a non-trivial anomalous diffusion behavior. The key idea is to start from any valid network configuration and, by successive steps of local changes in the structure, approach networks with a prescribed value of the spectral dimension. For this, a "spectral distance" function has to be defined measuring how far a given network configuration is away from the target spectral dimension d_s defined by the power law

$$I(\lambda) \propto \lambda^{d_\mathrm{s}/2} \tag{3.1}$$

in the integrated spectral density. Then, this distance function is minimized by the evolutionary optimization (Karalus and Porto, 2012).

3.1. Anomalous Diffusion

The overall behavior of diffusion or random walk dynamics is characterized by the asymptotic power laws as in equations (2.54) and (2.55). In regular lattices the power-law exponents are determined by the spatial dimension d of the lattice only. This behavior is called normal diffusion. The random walk dimension of normal diffusion is $d_\mathrm{w} = 2$ and the spectral dimension is $d_\mathrm{s} = d$ such that equations (2.50) and (2.52) are recovered. Otherwise, one speaks of anomalous diffusion. If $d_\mathrm{w} < 2$ the mean square displacement increases faster than normal and the process is called superdiffusive. For $d_\mathrm{w} > 2$, the average spreading is slower than normal which is referred to as subdiffusion. In terms of the spectral dimension, all non-integer values of d_s describe anomalous diffusion behavior. In this work, the focus lies on the construction of networks which exhibit anomalous diffusion in the sense of a non-trivial value of the spectral dimension.

The occurrence of anomalous diffusion can have various causes. A possible source of anomalous behavior is the dynamical process itself. In the case of Lévy flights the jump lengths of a random walk in continuous space are drawn from a heavy-tailed distribution resulting in a superdiffusive spreading. In continuous-time random walks, on the other hand, random waiting times between between two jumps are drawn. A broad distribution of waiting times leads to subdiffusive behavior. A second possible reason for anomalous diffusion is the underlying structure on which the diffusion takes place. In disordered or irregular structures like fractals subdiffusion is a very well-studied phenomenon (Havlin and ben-Avraham, 2002). The different generalized dimensions of many fractal structures have been determined by analytical or numerical methods. Networks provide the most general description of discrete substrates for diffusion processes comprising regular lattices, deterministic and random fractals, and

completely disordered structures which not necessarily need to be embedded in any space.

Although the anomalous diffusion properties of many discrete structures have been studied extensively, there is no constructive method available to form networks with a prescribed diffusion behavior. Since each variation in the network structure possibly affects the whole spectrum of the graph Laplacian there is no way to predict the Laplacian eigenvalues of a network, and thus its spectral dimension, from its constituent subnetworks. So, it is in general not possible to build up networks with a prescribed value of the spectral dimension from smaller building blocks with known Laplacian spectra. Other methods have to be applied to find networks with the desired anomalous diffusion behavior.

3.2. Evolutionary / Monte Carlo Optimization

A rather generally applicable strategy for a complex optimization problem—especially if no constructive method is available—is to explore the space of valid configurations by successive steps of (local) modifications towards overall improvements. This method can, of course, be applied to the optimization of networks with respect to their spectral dimension.

The strategy of optimization by successive improvements is a basic principle of biological evolution. In a simplified picture, evolution optimizes the viability of living organisms by modifications in the genetic information from one generation to the next and selection for the fitter individuals or groups. In the language of evolution, the optimization consists of two steps, mutation and selection. A mutation is a (local) change in the current state of the system. A fitness function measures how well a state of the system performs with respect to the optimization goal. It is a mapping from the state space into the real numbers. The second step is selection which decides, based on the fitness values of the different mutants, which one spreads and possibly takes over the population. In this picture, evolutionary optimization means finding states of maximal fitness by successive steps of mutation and selection.

The method of sequential sampling of a previously unknown configuration space is also the principle of Monte Carlo (MC) simulations in statistical physics (Newman and Barkema, 1999). At each MC time step an update of the current state of the system by a (local) modification is proposed. An acceptance function decides whether the proposed update is accepted as new state or discarded. The acceptance is based on the energy of the two states and possibly additional parameters like temperature. Usually the subspace sampled by MC simulations represents the Boltzmann distribution or some other ensemble grounded on basic physical laws. The seminal algorithm by Metropolis *et al.* (1953) sampling the Boltzmann distribution at a fixed temperature forms the basis of essentially all modern MC methods. In the Metropolis algorithm the acceptance probability equals the maximum value satisfying the detailed balance condition. Grounding on this sampling strategy very efficient optimization algorithms were developed. An example is simulated annealing (Kirkpatrick *et al.*, 1983) in which

the idea is to lower the temperature in the course of a standard Metropolis MC simulation in order to drive the system towards states of lower energy while giving it enough time at higher temperatures to escape from local minima. Other algorithms like taboo search (Cvijović and Klinowski, 1995) or energy landscape paving (Hansmann and Wille, 2002) introduce additional weights to lead the simulation away from previously sampled states into "unexplored" regions.

Evolutionary optimization of network structure has been successfully applied in various contexts such as:

- The emergence of critical connectivity in networks of Boolean threshold dynamics (Bornholdt and Rohlf, 2000).
- The evolution of reliability in such networks (Braunewell and Bornholdt, 2008).
- The effect of degree distributions in the evolution of Boolean threshold networks towards predefined output functions (Greenbury *et al.*, 2010; Oikonomou and Cluzel, 2006).
- The emergence of heterogeneous structures in local majority dynamics (Shao and Zhou, 2009).
- The optimization of Boolean networks with respect to their stability as well their ability to switch between different attractors (Szejka and Drossel, 2010).
- The emergence of modularity and network motifs in changing environments (Kashtan and Alon, 2005).
- The emergence of complex network structures in evolutionary food webs (Allhoff *et al.*, 2015).
- The synchronizability of networks with explicit calculation of the synchronization dynamics (Gorochowski *et al.*, 2010).
- The optimization of the Laplacian eigenratio λ_N/λ_2 as measure for the synchronizability (Donetti *et al.*, 2005, 2006, 2008; Rad *et al.*, 2008).
- The reconstruction of networks from their Laplacian spectra (Comellas and Diaz-Lopez, 2008; Ipsen and Mikhailov, 2002).

The most simple optimization strategy is to accept only modifications improving the current state of the system. In the language of evolution this means that only beneficial mutations are selected. The resulting evolutionary process is an adaptive walk in the configuration space. In the context of biological evolution adaptive walks occur in the regime of strong selection and weak mutations (Orr, 2002). In this regime, mutations happen rarely such that at any time only single mutations are present in a population and selection is so strong that, with a certain probability, the fitter mutant takes over the whole population before the next mutation occurs. In biological evolution, the choice of the transition probability distinguishes different types of adaptive walks.

The algorithm used here corresponds to the random adaptive walk in which any fitter mutant *always* becomes prevalent, independently of the fitness difference. From the viewpoint of MC simulation the optimization strategy simply corresponds to a simulation at zero temperature. In this case only updates lowering the energy are accepted. As an optimization algorithm, the viability of such a strategy will strongly depend on the difficulty of the optimization problem. On a "rugged landscape", a fitness or energy function with many local minima and maxima, this simple algorithm will very likely not find the global optimum but get trapped in a local fitness maximum or energy minimum. However, if the fitness or energy function has a simple form, e.g., a funnel-like shape, it can be a very efficient method since it is easy to implement and computationally less costly than more advanced algorithms. Since the shape of the function to be optimized is unknown in advance there is no way to decide a priori which is the best optimization method. So, it is usually a good strategy to start with the simplest algorithm and introduce more advanced techniques if needed.

Here, the simple adaptive walk strategy is combined with the most simple mutation rule sampling the space of connected networks with fixed numbers of vertices N and edges M. Namely, at each evolutionary time step a randomly chosen edge is removed from the network and reintroduced between two previously unconnected random vertices. This results in a minimal change in the adjacency and, hence, in the Laplacian matrix. As a consequence, one expects a minimal change in the Laplacian eigenvalues as well. There is, however, no guarantee for how small the resulting change in the spectrum actually is. If the mutated network does not consist of a single connected component, the edge is put back in its place and the whole procedure is repeated. The mutated network is selected as new state if its spectral distance is lower than the spectral distance of the previous state, i.e., if its integrated spectral density better resembles the power law in equation (3.1).

3.3. Quantification of Spectral Distance

In order to apply any optimization method a function is needed measuring how different a given (integrated) spectral density is from the target power law in equation (3.1). Such a mapping from the space of valid network configurations into the real numbers is called spectral distance measure and corresponds to the energy in a MC simulation or the negative fitness in an evolutionary context.

For the general problem of quantifying the similarity between two Laplacian spectra several distance functions were proposed. Jurman *et al.* (2011) analyze six different spectral distances, labeled D1 to D6. For two graphs G and H with Laplacian spectra $\{\lambda_1^{(G)} \leq \ldots \leq \lambda_N^{(G)}\}$ and $\{\lambda_1^{(H)} \leq \ldots \leq \lambda_N^{(H)}\}$, respectively, and corresponding spectral densities $\rho_G(\lambda)$ and $\rho_H(\lambda)$ they are defined as follows.

D1: The symmetrically normalized sum of squared differences between the ordered eigenvalues,

$$\mathcal{D}^{(1)}(G,H) = \begin{cases} \sqrt{\dfrac{\sum_i \left(\lambda_i^{(G)} - \lambda_i^{(H)}\right)^2}{\sum_i \left(\lambda_i^{(G)}\right)^2}} & \text{for } \sum_i \left(\lambda_i^{(G)}\right)^2 \leq \sum_i \left(\lambda_i^{(H)}\right)^2 , \\ \sqrt{\dfrac{\sum_i \left(\lambda_i^{(G)} - \lambda_i^{(H)}\right)^2}{\sum_i \left(\lambda_i^{(H)}\right)^2}} & \text{for } \sum_i \left(\lambda_i^{(H)}\right)^2 < \sum_i \left(\lambda_i^{(G)}\right)^2 . \end{cases} \tag{3.2}$$

D2: Let $\omega_i = \sqrt{\lambda_i}$ be the eigenfrequencies of the network of harmonic springs and $\tilde{\rho}(\omega)$ the corresponding eigenvalue densities. For the calculation, $\tilde{\rho}(\omega)$ is approximated by a sum of Lorentz distributions, $\tilde{\rho}(\omega) = K \sum_i \gamma/[(\omega - \omega_i)^2 + \gamma^2]$, with width γ and normalization K. Then,

$$\mathcal{D}^{(2)}(G,H) = \sqrt{\int \left[\tilde{\rho}_G(\omega) - \tilde{\rho}_H(\omega)\right]^2 \, \mathrm{d}\omega} \, . \tag{3.3}$$

D3: The square root of the summed squared differences between the ordered eigenvalues,

$$\mathcal{D}^{(3)}(G,H) = \sqrt{\sum_i \left(\lambda_i^{(G)} - \lambda_i^{(H)}\right)^2} . \tag{3.4}$$

D4: The same as D3 without taking the square root,

$$\mathcal{D}^{(4)}(G,H) = \sum_i \left(\lambda_i^{(G)} - \lambda_i^{(H)}\right)^2 . \tag{3.5}$$

D5: Integration over the difference between the spectral densities raised to the pth power. Additionally, a weight function $\mu(\lambda)$ is introduced,

$$\mathcal{D}^{(5)}(G,H) = \int \mu(\lambda) \left[\rho_G(\lambda) - \rho_H(\lambda)\right]^p \mathrm{d}\lambda \, . \tag{3.6}$$

For the computations, a binning into K equally spaced bins and a weight function $\mu(\lambda) = (1-\lambda)^4$ is suggested.

D6: Uses the Jensen-Shannon measure

$$\mathrm{JS}[p_1, p_2] = \frac{1}{2} \mathrm{KL}\left[p_1, \frac{p_1 + p_2}{2}\right] + \frac{1}{2} \mathrm{KL}\left[p_2, \frac{p_1 + p_2}{2}\right] \tag{3.7}$$

which is a symmetrization of the Kullback-Leibler divergence

$$\mathrm{KL}[p_1, p_2] = \int p_1(x) \log \frac{p_1(x)}{p_2(x)} \, \mathrm{d}x \tag{3.8}$$

as measure for the difference between two (probability) distributions. The spectral distance is then defined as the square root of the Jensen-Shannon measure,

$$\mathcal{D}^{(6)}(G,H) = \sqrt{\mathrm{JS}(\rho_G, \rho_H)}\,. \tag{3.9}$$

For the computations, a Gaussian kernel is used, i.e., the densities are approximated as sum of Gaussian distributions, $\rho(\lambda) = 1/\sqrt{2\pi\sigma}\sum_i \exp(-(\lambda - \lambda_i)^2/2\sigma^2)$, with a finite width σ.

Some of these measures where originally suggested as distances between spectra of the normalized Laplacian. The integration ranges in D2, D5, and D6 can be restricted to $[0, 2]$ in this case and $[0, \infty)$ for the algebraic Laplacian. Only the binning suggested for the approximate calculation of D5 can exclusively be carried out on a finite interval and is, hence, restricted to the normalized Laplacian. However, other approximations to calculate $\mathcal{D}^{(5)}$ like a Gaussian or Lorentzian kernel as suggested for D2 and D6 can be used here as well.

The six spectral distances mentioned above obviously share some common ideas. D1, D3, and D4 are based on the summation of differences between the eigenvalues in ascending order. They differ only in the rescaling of the total range. D2 and D5 are based on an integration over the difference in spectral densities while D6 builds on the Kullback-Leibler divergence as an established "distance function" between two probability distributions. All six measures have their particular (dis)advantageous properties. For example, consider two spectra which are identical up to a removal of the largest and introduction of an additional small eigenvalue. D1, D3, and D4 will estimate a large distance between these spectra since always eigenvalues of the same rank in the spectrum, and not of similar size, are compared. D2 and D5 will produce a much smaller distance in this case since only two peaks in the densities will contribute to the integrals. For explicit calculations of D2, D5, and D6 between realizations of finite networks approximations like the use of a kernel function are needed. By means of D1, D3, and D4 it is not possible to compare a specific realization of a finite network with an ensemble or some other theoretical spectral density. Jurman *et al.* (2011) compared the spectral distances D1 to D6 in several benchmark tests. They do, however, not achieve to construct a clear picture of which one is, in general, better or worse suited.

For the purpose of the evolutionary optimization in this thesis a different spectral distance will be used. Like D2 and D5 it is based on an integration over the difference between the spectral densities. Instead of the spectral density function $\rho(\lambda)$ itself the integrated spectral density $I(\lambda)$ is used here. This has the advantage that no approximation—for example by means of a Gaussian or Lorentzian kernel as suggested in D2, D6—is needed for the comparison of specific finite networks. Additionally, it is straightforward to compare the Laplacian spectrum of a specific network with any theoretical, and not necessarily realizable, spectral function. In this work, the evolutionary targets are spectral densities that follow a power law as in equation (3.1). It is therefore convenient to work on logarithmic scales and introduce a *logarithmically*

integrated spectral density. To this end, first the Laplacian eigenvalues are rescaled by their maximum value,

$$\tilde{\lambda}_i = \frac{\lambda_i}{\max_j\{\lambda_j\}}\,. \tag{3.10}$$

This results in a rescaled spectrum between 0 and 1, $0 = \tilde{\lambda}_1 \leq \tilde{\lambda}_2 \leq \ldots \leq \tilde{\lambda}_N = 1$, so that the integration can be restricted to a finite interval. The logarithmically integrated spectral density is then defined as a step function, similar to equation (2.67), on double-logarithmic scale,

$$\tilde{I}(\log\tilde{\lambda}) = \log\left[\frac{1}{N}\sum_{i=1}^{N}\Theta(\log\tilde{\lambda} - \log\tilde{\lambda}_i)\right]. \tag{3.11}$$

The spectral distance $\mathcal{D}$ here is considered to be a function of two logarithmically integrated spectral densities $\tilde{I}$ and $\tilde{I}'$,

$$\mathcal{D}(\tilde{I},\tilde{I}') = \int_{\log\tilde{\lambda}^*_{\min}}^{0}\left|\tilde{I}(\log\tilde{\lambda}) - \tilde{I}'(\log\tilde{\lambda})\right|^2 \mathrm{d}(\log\tilde{\lambda})\,. \tag{3.12}$$

The lower integration boundary $\log\tilde{\lambda}^*_{\min}$ has to be chosen carefully. Since $\tilde{\lambda}_1 = 0$, $\log\tilde{\lambda}_1$ is not defined. For any connected network λ_2 is the smallest non-zero eigenvalue and $\tilde{I}(\log\tilde{\lambda}) = \log(N^{-1})$ for $\log\tilde{\lambda} < \log\tilde{\lambda}_2$. In the case of $\tilde{I}$ and $\tilde{I}'$ being the logarithmically integrated spectral densities of two connected networks of the same size, both functions will have the same value in that range. Thus, the lower integration boundary $\log\tilde{\lambda}^*_{\min}$ can be chosen as the minimum of the two second eigenvalues, $\tilde{\lambda}^*_{\min} = \min\{\tilde{\lambda}_2, \tilde{\lambda}'_2\}$, where the two functions become different. For comparison of a given connected network with a general target function $\tilde{I}^{\text{target}}$ by $\mathcal{D}(\tilde{I},\tilde{I}^{\text{target}})$, the lower integration boundary is chosen such that $\tilde{I}^{\text{target}}(\log\tilde{\lambda}) < \log(N^{-1})$ for $\log\tilde{\lambda} < \log\tilde{\lambda}^*_{\min}$ and $\tilde{I}^{\text{target}}(\log\tilde{\lambda}) \geq \log(N^{-1})$ for $\log\tilde{\lambda} \geq \log\tilde{\lambda}^*_{\min}$. Note that $\tilde{I}^{\text{target}}$ should be monotonically increasing but is not necessarily continuous. If $\tilde{I}^{\text{target}}$ is continuous, then $\tilde{I}^{\text{target}}(\tilde{\lambda}^*_{\min}) = \log(N^{-1})$. For the power-law integrated spectral density (3.1) the target logarithmically integrated spectral density reads

$$\tilde{I}^{\text{target}}(\log\tilde{\lambda}) = \frac{d_{\text{s}}}{2}\log\tilde{\lambda} \tag{3.13}$$

and the resulting lower integration boundary is

$$\log\tilde{\lambda}^*_{\min} = -\frac{2}{d_{\text{s}}}\log N\,. \tag{3.14}$$

Compared to an integration on linear scales, the defined spectral distance introduces an effective weight function $\tilde{\lambda}^{-1}$ enhancing the importance of the smaller eigenvalues. This, on the one hand, accounts for the fact that in the target density $I(\lambda) \propto \lambda^{d_{\text{s}}/2}$ the eigenvalues are not equally distributed but actually more dense in the lower part of the spectrum. On the other hand, in the limit $N \to \infty$ the spectral dimension is defined by the limit $\lambda \to 0$ so that the correct scaling in the regime of small eigenvalues can be considered as more important.

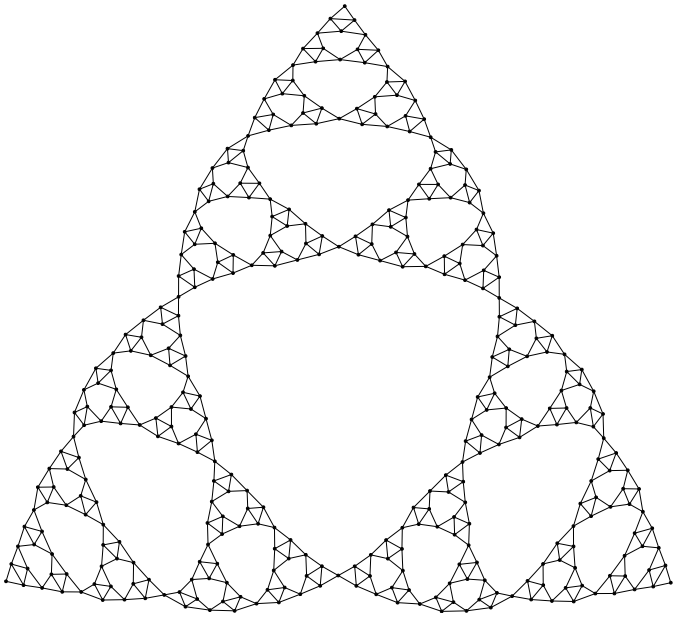

Figure 3.1. Network topology of the finite Sierpinski triangle of generation 6.

3.4. Application to Anomalous Diffusion in Networks

The stage is now set for the evolutionary optimization of networks towards a power law in the spectral density with a prescribed non-trivial spectral dimension. The evolution target is the logarithmically integrated spectral density defined by equation (3.13) with two exemplary values $d_{\mathrm{s}}^{(1)} = 1.4$ and $d_{\mathrm{s}}^{(2)} = 1.1$. Being non-integer values, both are non-trivial in the sense that it is assured that no regular lattices exist with these values of the spectral dimension. The first target $d_{\mathrm{s}}^{(1)} = 1.4$ is chosen close to the spectral dimension of a well-studied deterministic fractal, the Sierpinski triangle, which takes the value $d_{\mathrm{s}} = 2\log(3)/\log(5) \approx 1.365$ (ben-Avraham and Havlin, 2000). The finite Sierpinski triangle network of generation 6, shown in figure 3.1, has $N = 366$ vertices and $M = 729$ edges. So, choosing a network size in the same order of magnitude, it can be expected that the construction of a network with a spectral dimension around $d_{\mathrm{s}}^{(1)} = 1.4$ is not an impossible task. The second evolutionary target $d_{\mathrm{s}}^{(2)} = 1.1$ is more demanding in the sense that this number is already close to 1 and the given numbers of edges and vertices will definitely not fit to form a 1-dimensional lattice (chain). Thus, configurations with $d_{\mathrm{s}} = 1.1$ can be expected to be more difficult to achieve than those with $d_{\mathrm{s}} = 1.4$. The spectral distance from these evolution targets is calculated by $\mathcal{D}(\tilde{I}, \tilde{I}^{\mathrm{target}})$ as defined in equation (3.12). It is minimized by an adaptive walk in the space of connected networks with fixed numbers of vertices N and edges M. In each evolutionary time step one edge of the network is randomly moved. A move is accepted if it lowers the spectral distance.

In order to evaluate to what extent the outcome depends on the starting conditions two kinds of initial configurations are chosen, i) 2-dimensional square lattices with periodic

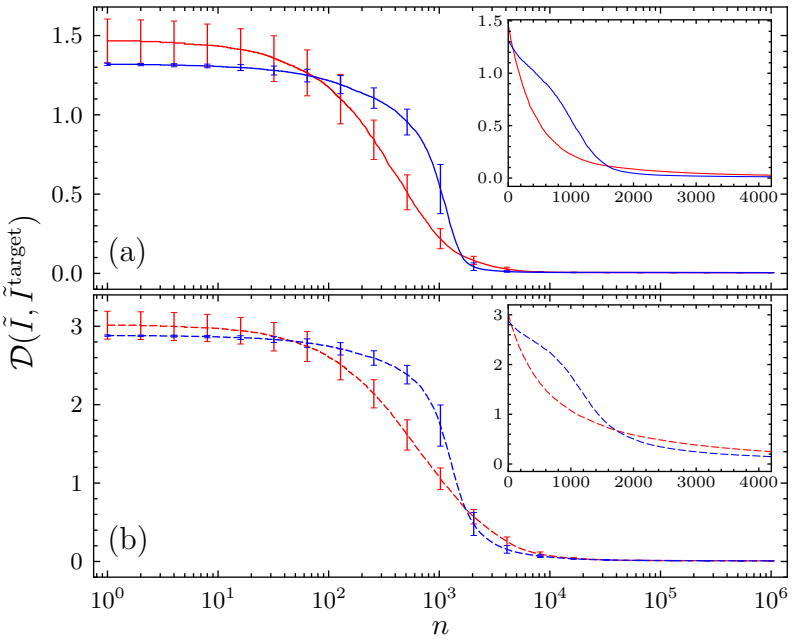

Figure 3.2. Change of spectral distance $\mathcal{D}(\tilde{I}, \tilde{I}^{\text{target}})$ with evolutionary time steps n for target spectral dimensions (a) $d_{\text{s}}^{(1)} = 1.4$ (solid lines) and (b) $d_{\text{s}}^{(2)} = 1.1$ (dashed lines). The lines display the mean spectral distance of 100 realizations and the error bars mark one standard deviation of the respective distribution. The initial configurations are indicated by the colors, red for random graphs $G(N, M)$ and blue for 2d square lattices. In the insets, the mean distances are shown on a linear scale for small number of evolutionary time steps.

boundary conditions and ii) connected random graphs $G(N, M)$. In both cases the same numbers of vertices and edges are chosen,

$$N = 19 \times 19 = 361\,,$$
$$M = 2N = 722\,,$$

yielding an average degree of $\langle k \rangle = 4$. Each evolution is carried out for 10^6 evolution steps and repeated with different random numbers for 100 independent realizations. In the following, the outcome of this optimization procedure is discussed.

The first question is whether the evolutionary algorithm was appropriately chosen and succeeds to find network configurations with a minimal spectral distance to the target eigenvalue density. To this end, figure 3.2 shows the evolution of the spectral distance in the course of the evolutionary time. For both target spectral dimensions, (a) $d_{\text{s}}^{(1)} = 1.4$ in the upper panel and (b) $d_{\text{s}}^{(2)} = 1.1$ in the lower panel, a significant drop in the spectral distance during the first 2000 evolution steps followed by a slow further decay can be observed. After around 10^4 evolution steps the variation between the individual realizations and also the difference between the two initial conditions vanish.

Target spectral dimension	Inital configuration	Spectral distances $\mathcal{D}_{\text{ini}}$	$\mathcal{D}_{\text{fin}}$	Factor $\mathcal{D}_{\text{ini}}/\mathcal{D}_{\text{fin}}$
$d_{\text{s}}^{(1)} = 1.4$	lattice	1.3(0)	0.0046(3)	280
	random	1.5(2)	0.0047(5)	320
$d_{\text{s}}^{(2)} = 1.1$	lattice	2.9(0)	0.0077(6)	380
	random	3.0(2)	0.0077(6)	390

Table 3.1. Spectral distances $\mathcal{D}_{\text{ini}}$ before and $\mathcal{D}_{\text{fin}}$ after the evolutionary optimization. The values in parentheses indicate the standard deviations of the respective distributions.

This is a strong indication that the spectral distance "landscape" has a rather simple form and most of the evolution runs end up in the same region of the configuration space. The exact figures of the initial and final spectral distances $\mathcal{D}_{\text{ini}}$ and $\mathcal{D}_{\text{fin}}$, respectively, are given in table 3.1. The relative improvement factor $\mathcal{D}_{\text{ini}}/\mathcal{D}_{\text{fin}}$ indicates that the evolution succeeds to lower the spectral distance by approximately a factor of 300, more then two orders of magnitude.

The absolute figures of the spectral distance are not easy to interpret. There is no scale involved telling which values of $\mathcal{D}$ can be considered as small or large in absolute numbers. Hence, the question to what extent the evolutionary optimization succeeds to construct networks with the prescribed spectral dimension cannot be answered just by the absolute values of the spectral distance. A power law in the spectral density, i.e., a linear relation in the logarithmically integrated spectral density $\tilde{I}(\log\tilde{\lambda})$, however, should be easily recognizable by visual inspection. Figure 3.3 shows typical logarithmically integrated spectral densities of the initial and evolved networks after 10^6 evolution steps. While among the initial networks the logarithmically integrated spectral density of the square lattice might exhibit a linear region, the spectra of the random graphs clearly do not follow any power law.[1] The spectra of the evolved networks, however, follow the respective target functions, $\tilde{I}^{\text{target}}(\log\tilde{\lambda}) = d_{\text{s}}/2\log\tilde{\lambda}$, very closely. Only for the larger eigenvalues a systematic deviation above the target function, especially for $d_{\text{s}}^{(2)} = 1.1$, can be observed. Of course, also the discretization due to the immanent finite step size in the integrated densities leads to a deviation from the linear target function.

How do these spectra translate into the dynamical behavior of diffusion? The average return probability of a random walker $P_0(t)$ is related to the eigenvalue density by equation (2.70). For a discrete spectrum $\{\lambda_1, \ldots, \lambda_N\}$, this yields a simple relation to calculate $P_0(t)$ from Laplacian eigenvalues,

$$P_0(t) = \int_0^\infty \mathrm{e}^{-\lambda t}\rho(\lambda)\,\mathrm{d}\lambda = \frac{1}{N}\sum_{i=1}^{N}\int_0^\infty \mathrm{e}^{-\lambda t}\delta(\lambda-\lambda_i)\,\mathrm{d}\lambda = \frac{1}{N}\sum_{i=1}^{N}\mathrm{e}^{-\lambda_i t}\,. \tag{3.15}$$

The average return probability for the initial as well as for the evolved networks is shown in figure 3.4. The lines are averaged over 100 realizations with the respective standard

[1] Although not explicitly derived, the spectral density of Erdős-Rényi random networks should follow a semi-circle law. An exact expression is available only implicitly via the solution of a nonlinear integral equation (Bray and Rodgers, 1988).

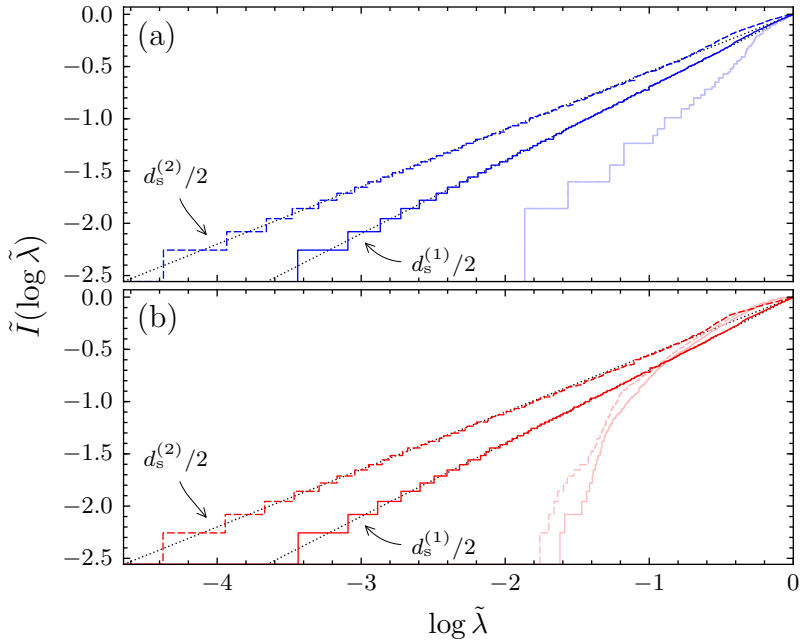

Figure 3.3. Logarithmically integrated eigenvalue densities $\tilde{I}(\log\tilde{\lambda})$ of individual realizations before and after the evolutionary optimization. The initial configurations are (a) 2d square lattices (light blue line) and (b) connected random graphs $G(N, M)$ (light red lines). The spectra of the corresponding evolved networks are shown in the same panels in dark colors. The target spectral dimensions are $d_s^{(1)} = 1.4$ (solid lines) and $d_s^{(2)} = 1.1$ (dashed lines). The black dotted lines show the two target logarithmically integrated densities $\tilde{I}^{\text{target}}(\log\tilde{\lambda}) = d_s/2 \log\tilde{\lambda}$ with annotated slopes.

deviations indicated by the error bars. For the evolved networks, $P_0(t)$ indeed follows the expected power law $P_0(t) \propto t^{-d_s/2}$ with $d_s^{(1)} = 1.4$ and $d_s^{(2)} = 1.1$, respectively, up to around 10^2 to 10^3 time steps. Also the square lattice exhibits this behavior with the expected spectral dimension of normal diffusion in 2 dimensions, $d_s^{(n)} = 2$. As indicated by the standard deviation, the dynamical variation between the individual realizations is rather small. Hence, the return probability in the evolved networks follows the prescribed power law very closely—not only on the level of averages but also for the individual realizations.

Having seen that the evolutionary optimization succeeds to minimize the defined spectral distance and that the resulting networks indeed exhibit the prescribed eigenvalue scaling and dynamical behavior, the subsequent question is what the structural properties of the evolved networks are. The size and the average degree of the networks are kept fixed, so the first property to look at shall be the distribution of vertex degrees. Figure 3.5 displays the degree distributions $P(k)$ of the two initial and the corresponding evolved networks for both target spectral dimensions. Despite the large dissimilarity between the initial configurations, a delta distribution for the square lattice and a Poisson

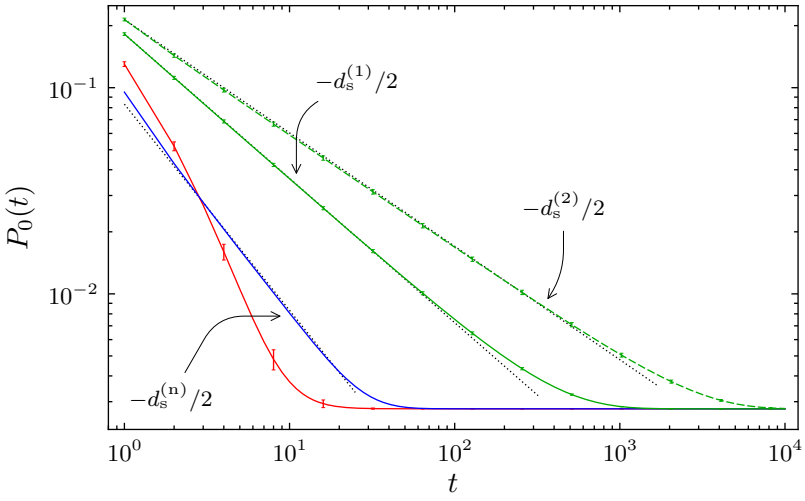

Figure 3.4. Average return probability $P_0(t)$ for initial and evolved networks. The lines display the mean $P_0(t)$ averaged over 100 realizations each with the error bars indicating one standard deviation. Shown are curves for random graphs $G(N, M)$ (red), 2d square lattices (blue), and evolved networks for the two evolution targets, $d_s^{(1)} = 1.4$ (solid green) and $d_s^{(2)} = 1.1$ (dashed green). The black dotted lines are linear guides to the eye and have the slopes annotated, $d_s^{(n)} = 2$ for normal diffusion in 2 dimensions.

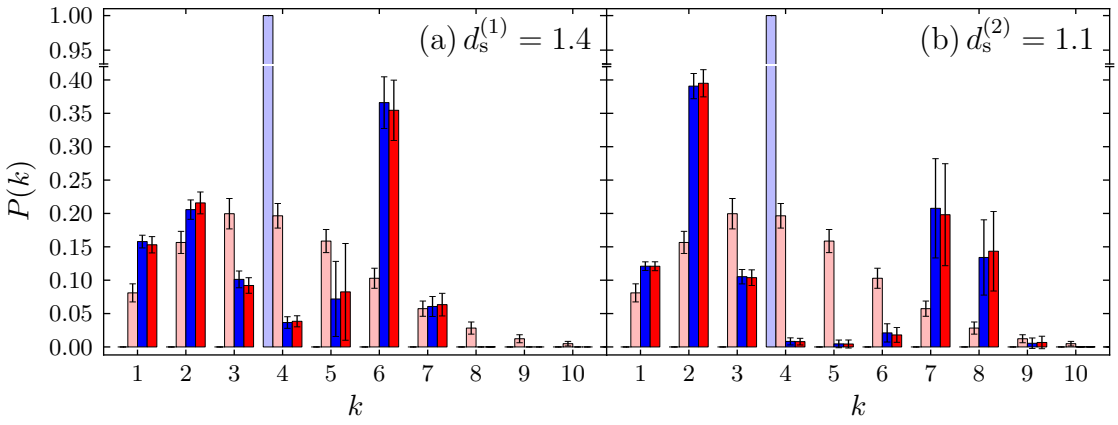

Figure 3.5. Degree distributions $P(k)$ of—from left to right—the initial networks, square lattices (light blue) and random graphs (light red), as well as the corresponding evolved networks (dark blue and dark red, respectively) for target spectral dimensions (a) $d_s^{(1)} = 1.4$ and (b) $d_s^{(2)} = 1.1$. The histogram bars show the average $P(k)$ over 100 realizations of the evolution and the error bars mark one standard deviation.

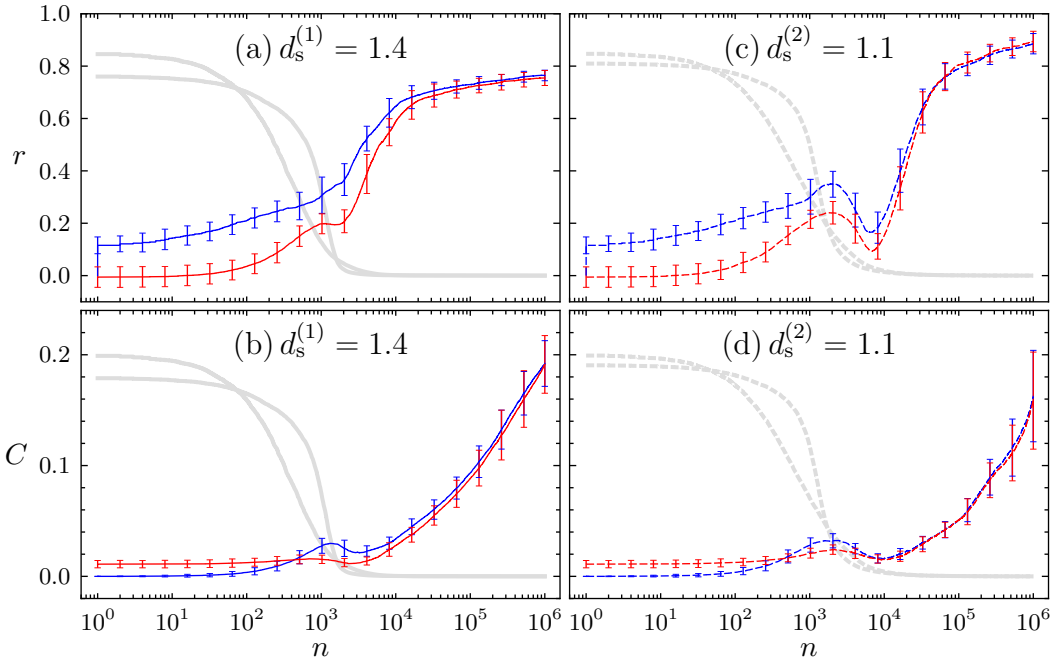

Figure 3.6. Evolution of the degree assortativity r (upper panels (a) and (c)) and clustering coefficient C (lower panels (b) and (d)) with the number of steps n for target spectral dimensions $d_s^{(1)} = 1.4$ (left panels (a) and (b)) and $d_s^{(2)} = 1.1$ (right panels (c) and (d)). The lines display the average values over 100 realizations of the evolution and the error bars mark one standard deviation. For comparison, the evolution of the spectral spectral distances from figure 3.2 is shown qualitatively as gray curves in the background.

distribution for the random graphs $G(N, M)$, the degree distributions of the networks which have been evolved towards the same target spectral dimension closely resemble each other. For both target spectral dimensions bimodal degree distributions emerge. The fraction of vertices with degrees around the average degree $\langle k \rangle = 4$ is very small while the number of vertices with higher and lower degrees is enhanced by the evolution. For the target spectral dimension $d_s^{(1)} = 1.4$ the distribution is peaked highest at $k = 6$ and secondly around $k = 2$, see figure 3.5(a). For $d_s^{(2)} = 1.1$, highest fraction of vertices is found with $k = 2$ and a second peak is located at $k = 7, 8$, as seen in figure 3.5(b). In both cases, the networks become more heterogeneous in their degree in the course of the evolution.

Further insight into the topological properties of the evolved networks can be achieved by examining correlations in the network structures. In figure 3.6 the evolution of the degree assortativity r and the clustering coefficient C for all four combinations of the two initial conditions and both evolution targets are shown. The lines show the change of the average values of r and C over 100 realizations and the error bars display one

standard deviation indicating the spread in the respective quantity. As before, the curves for the different initial conditions become indistinguishable in the course of the evolution supporting the conjecture that the outcomes are independent of the initial configurations. The degree assortativity r as shown in figure 3.6(a) and (c) increases in the course of the evolution to rather high values, $r = 0.76(3)$ and $r = 0.89(4)$ for target spectral dimensions $d_{\mathrm{s}}^{(1)} = 1.4$ and $d_{\mathrm{s}}^{(2)} = 1.1$, respectively. This positive correlation between the vertex degrees strongly indicates a segregation of vertices with high and low degrees, a second level of heterogeneity in the network structure. Interestingly, the increase in r is not simultaneous with the drop in the spectral dimension but occurs only afterwards in the course of the evolution. For the second target spectral dimension $d_{\mathrm{s}}^{(2)} = 1.1$ an intermediate drop followed by an even stronger rise in r is observed. Hence, the observed heterogeneity might actually be associated with the fine tuning of the spectrum instead of its overall shape. This interpretation is supported by the evolution of the clustering coefficient C shown in figure 3.6(b) and (d) for the target spectral dimensions $d_{\mathrm{s}}^{(1)} = 1.4$ and $d_{\mathrm{s}}^{(2)} = 1.1$, respectively. C also increases in the course of the evolution. Since the low degree vertices with $k = 1, 2$ in a connected network cannot form any triangles an increasing clustering coefficient must be generated by more densely connected regions of the vertices with higher degree. The increase in C, however, happens even later in evolutionary time than the change of the spectral distance and the degree assortativity. Additionally, the clustering coefficient seems not to converge but instead continues to increase and diversify in the course of the evolution. This, again, suggests that the segregation might actually be a secondary effect associated with the fine tuning of the spectrum.

Taken together, the observed structural properties suggest that rather heterogeneous structures evolve out of the homogeneous initial configurations. The bimodal degree distributions together with large degree-degree correlations and increasing clustering are indicators for a core-periphery structure. The high-degree vertices in the core form a rather den-sly connected community while in the periphery the low-degree nodes are more loosely connected. To verify whether such a core-periphery structure can indeed be seen in the evolved network, exemplary network configurations taken from the evolution are shown in figure 3.7. Figure 3.7(a) and (b) display the initial networks, a 2d square lattice which due to the periodic boundary conditions has the global topology of a torus and a connected random graph $G(N, M)$. While the local structure of the lattice should be still recognizable although not all vertices and edges can be distinguished clearly, random graphs are usually almost impossible to visualize in a plane. Nevertheless, one can guess from the visual appearance that vertices of different degrees are well mixed and overall form a rather homogeneous network. Exemplary evolved networks with target spectral dimensions $d_{\mathrm{s}}^{(1)} = 1.4$ and $d_{\mathrm{s}}^{(2)} = 1.1$ are shown in figure 3.7(c) and (d), respectively. Although the structures of the den-sly connected regions are again not visually accessible in this planar representation the pictures support the core-periphery conjecture. In both cases, a fraction of vertices is found in one or several den-sly connected cores while the periphery is formed by an extended tree-like structure.

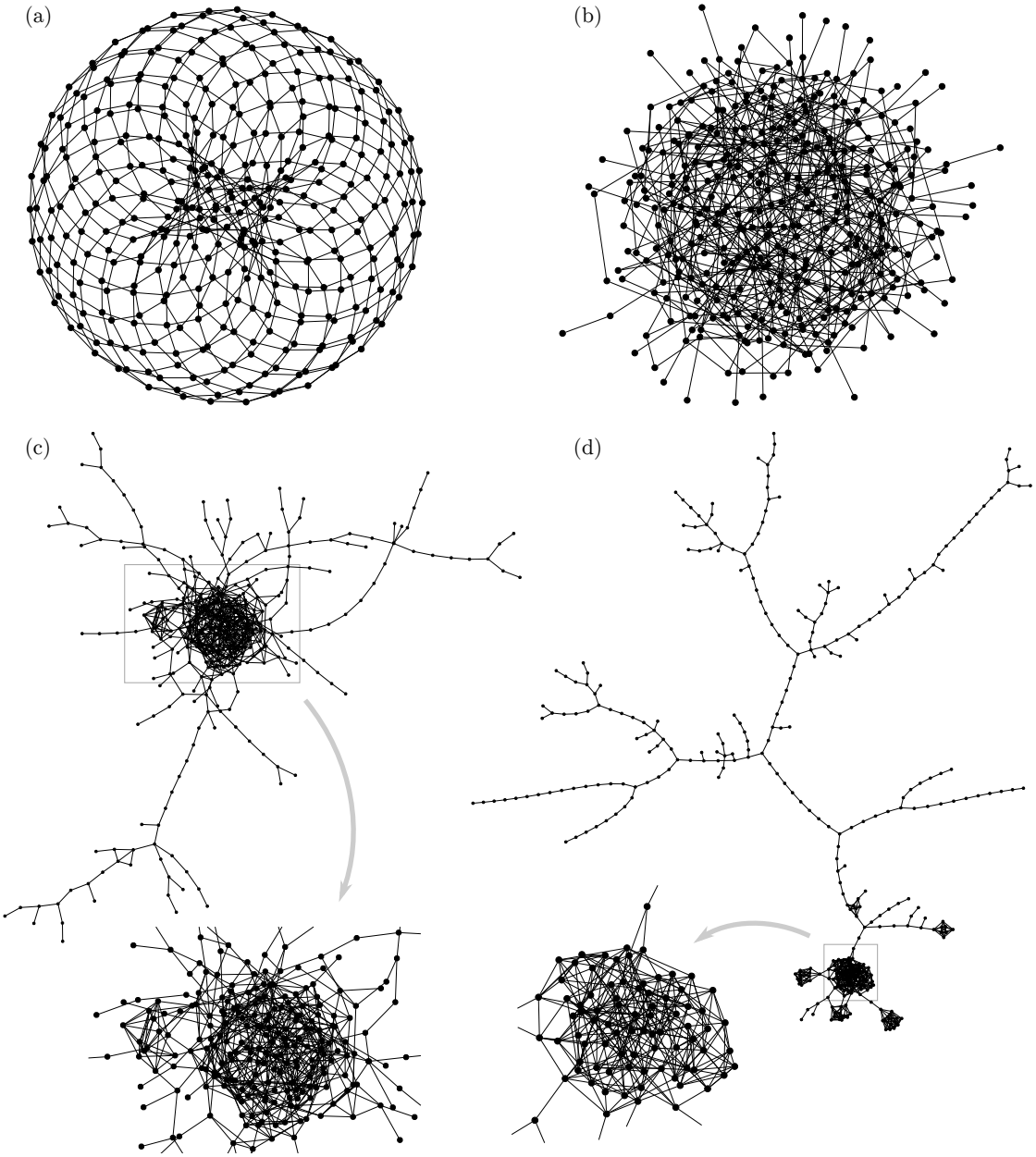

Figure 3.7. Visualization of typical network configurations taken from the evolutionary optimization. Top: Initial configurations (a) 2d square lattice with periodic boundary conditions and (b) random graph $G(N, M)$. Bottom: Evolved networks with target spectral dimensions (c) $d_{\mathrm{s}}^{(1)} = 1.4$ and (d) $d_{\mathrm{s}}^{(2)} = 1.1$. For better visibility, two very dense regions of the evolved networks are additionally shown enlarged.

For the lower target spectral dimension $d_{\mathrm{s}}^{(2)} = 1.1$ more vertices are in the periphery which seems to be further extended and less interconnected than for $d_{\mathrm{s}}^{(1)} = 1.4$.

Concluding this chapter, it was shown that evolutionary optimization can be successfully applied to construct networks of a prescribed non-trivial spectral dimension with satisfactory precision. To this end, a spectral distance measure was defined which allows for comparison of a given discrete spectrum with other spectra as well as with a general function. The simple mutation-selection scheme of moving a single edge in each evolutionary step and accepting only those mutations that lower the spectral distance proved sufficient to explore the relevant part of the configuration space. The evolution improves the spectral distance by more than two orders of magnitude and the evolved networks closely reproduce the prescribed eigenvalue scaling and, consequently, the dynamical behavior. Structurally, the networks become heterogeneous in the course of the evolution. They exhibit a bimodal degree distribution with the high-degree vertices forming den-sly connected cores and the low-degree vertices spreading out in a more loosely connected tree-like periphery.

4. Reconstruction of Evolved Networks

In the previous chapter, networks were successfully evolved towards an approximative power-law scaling of the Laplacian spectrum. The degree distributions of these networks as well as correlations, measured by the assortativity and clustering coefficients, were observed to change significantly in the course of the evolution. To which extent do these structural properties determine the spectral and, consequently, the dynamical behavior of the networks? Is there a way to construct random networks with power-law Laplacian spectra from scratch based on the correlation functions? These questions are addressed in the following. The correlations are extracted from the evolved networks and used to generate random networks following the prescribed distributions.

4.1. Generation of Two-Point Correlated Random Networks with Clustering

Several algorithms have been proposed to construct random networks with a given degree distribution and correlations between vertex degrees. Mostly, these algorithms are extensions of the configuration model (Molloy and Reed, 1995) explained in section 2.2. The basic idea of the configuration model is to first assign a number of half-edges according to a degree sequence to each vertex and then randomly pair these in order to form the edges of the networks. If the algorithm succeeds, the result is a random network with exactly the chosen degree sequence. Ángeles Serrano and Boguñá (2005) extended the configuration model algorithm to incorporate the degree-dependent clustering $C(k)$, defined in equation (2.15), as additional specification of the random networks to be constructed. In a two-step process, first those half-edges are selected to be paired that form additional triangles in the classes of vertex degrees in which the prescribed level of clustering is not yet reached. Secondly, remaining free half-edges are matched randomly. A different approach was proposed by Weber and Porto (2007) for the generation of networks with a prescribed degree-degree correlation matrix $P(j,k)$ which also defines the degree distribution $P(k)$. In this algorithm, the half-edges to be paired are selected according to the remaining number of edges to be built between vertices of the corresponding degree class.

In order to incorporate both, degree-degree correlations and clustering, into random network generation, Pusch *et al.* (2008) extended the algorithm by Ángeles Serrano and Boguñá in order to additionally include prescribed two-point correlations $P(j,k)$. Just as the original configuration model actually generates random networks with a given discrete degree sequence instead of a degree distribution, this algorithm takes discrete frequency distributions as input. The vertex degrees are given by $\hat{P}(k)$, the number

of vertices with degree k, the two-point correlations by $\hat{P}(j,k)$, the number of edges between vertices with degrees j and k, and the clustering by $\hat{T}(k)$, the number of triangle corner vertices with degree k. The specification of $\hat{T}(k)$ or the corresponding normalized distribution $T(k)$—the probability that a randomly chosen triangle vertex has degree k—is completely equivalent to the degree-dependent clustering $C(k)$. The distributions only differ in their normalization. The algorithm itself consists of three steps. First, half-edges are selected by the degree of the vertices to which they are connected under the conditions that in the corresponding degree classes the prescribed number of edges is not yet reached and that their paring forms triangles in the degree classes that have not reached the prescribed number of triangle corners. Secondly, random half-edges are chosen to form edges in the degree classes which are not satisfied yet. Thirdly, still remaining half-edges are paired randomly as in the configuration model. The algorithm will, if possible, exactly reproduce the prescribed degree sequence, very closely resemble the given two-point correlations, and approximately generate the prescribed numbers of triangle corners. A more detailed description of the algorithm is presented in appendix A.

This random network generation algorithm is used here to reconstruct the evolved networks of chapter 3 from their two-point correlations and degree-dependent clustering. The goal is to find out to which extent the spectral properties of these networks are determined by the correlations.

4.2. Individual Reconstruction of Evolved Networks

First, different realizations of the evolved networks are reconstructed individually. The network configurations are taken from the evolution of random networks towards the target spectral dimension $d_{\mathrm{s}}^{(1)} = 1.4$. The degree distribution, the degree-degree correlations, and the degree-dependent clustering of the final configurations, i.e., after 10^6 evolution steps, are calculated as discrete frequencies $\hat{P}(k)$, $\hat{P}(j,k)$, and $\hat{T}(k)$. This is done individually for the 100 realizations of the evolution. For each of the triples $\hat{P}(k)$, $\hat{P}(j,k)$, and $\hat{T}(k)$, 100 samples of the random networks are generated and analyzed.

In order to see how well the reconstructed networks reproduce the evolution target, i.e., the prescribed power-law scaling in the Laplacian spectrum, the first quantity to look at is the spectral distance $\mathcal{D}$ to the evolution target. Figure 4.1 displays the distribution $f(\mathcal{D})$ of the spectral distances for the reconstructed networks. For comparison, the distributions $\hat{P}(k)$, $\hat{P}(j,k)$, and $\hat{T}(k)$ were also calculated for an uncorrelated random network of the same size, the initial configuration of the evolutionary optimization. These distributions were used for the generation of 1000 realizations of reconstructed random networks by the same algorithm. The spectral distance of the reconstructed evolved networks is always higher than the average value of the evolved networks but significantly lower than the values of the random network and its reconstructed networks. Hence, the degree distribution and the two correlation measures together indeed encode the spectral behavior to a significant extent. Interestingly, the distribution shows a structure of several, mostly well-separated peaks. As indicated by the different colors in figure 4.1, the multi-peak structure is generated by different numbers of connected components n_{c}

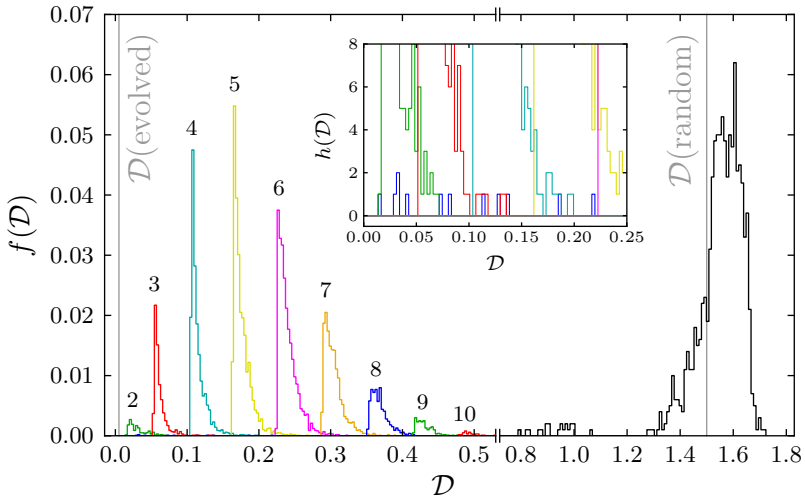

Figure 4.1. Histogram of spectral distances in the individually reconstructed networks. Shown is the fraction $f(\mathcal{D})$ of networks with spectral distance $\mathcal{D}$ to the evolution target reconstructed from the evolved networks (colored lines) and a random network of the same size (black line). In the former, the different colors stand for different numbers of connected components n_c, indicated by the numbers above the curves. Networks with $n_c = 1$ and $n_c = 11$ appear with very low frequencies. The (average) spectral distances of the networks which were reconstructed are indicated by the vertical gray lines. The inset shows an enlarged segment such that the networks with $n_c = 1$ become visible (blue lines), now in terms of absolute frequencies $h(\mathcal{D}) = f(\mathcal{D}) \times 10^4$.

in the reconstructed networks. In contrast to the network evolution, the construction of the correlated random networks provides no possibility to control the number of components the generated networks will have. Only a very small fraction of the resulting reconstructed networks, namely 12 out of 10^4, are found to be globally connected like the evolved networks. An explanation for the separation of the distribution into classes of equal number of connected components is that the degeneracy of the smallest eigenvalue $\lambda_1 = 0$ is given by the number of components and the smallest eigenvalues have the largest influence on the spectral distance. Interestingly, the globally connected networks with $n_c = 1$ deviate from the trend and do not exhibit spectral distances lower than those with $n_c = 2$. Due to the small number of realizations this observation can, however, not be considered as statistically significant.

The influence of the small eigenvalues on the spectral distance is also clearly visible in the integrated spectral densities. Figure 4.2(a) displays the averaged logarithmically integrated spectral densities of the evolved and their reconstructed networks. Evidently, the higher presence of small eigenvalues is the main cause for the larger deviation from target function. In figure 4.2(b) the spectral densities are individually averaged for the different classes of equal number of connected components. It shows how the deviation from the target function in the region of small eigenvalues indeed increases with the

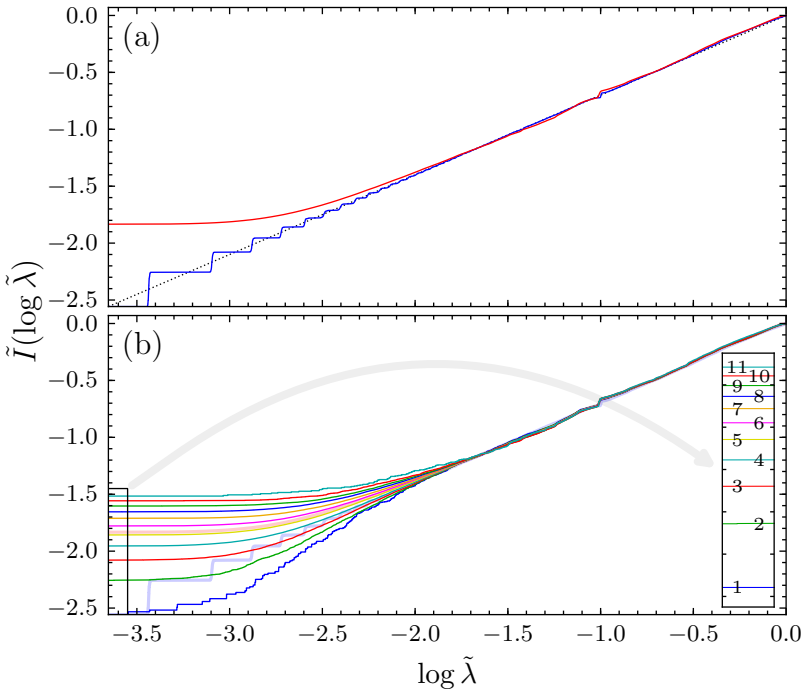

Figure 4.2. Averaged logarithmically integrated Laplacian spectral density of reconstructed networks. In the upper panel (a), the averaged spectral densities of the evolved (blue line) and the reconstructed (red line) networks are shown. The dotted black line displays the evolution target. The latter is broken down to the number connected components in the lower panel (b). The curves are labeled by their numbers of components in the inset, the color code is the same as in figure 4.1. For comparison, the transparent lines in the background show the curves of panel (a) again.

number of connected components in the network. The increasing degeneracy of the zero eigenvalue makes the integrated densities start out from higher values so that an increasing initial exceedance of the target function is inevitable. The integrated densities of the globally connected networks with $n_c = 1$ actually fall below the target function explaining the higher spectral distance than for $n_c = 2$ although the global trend of a lower initial value is continued. But again, due to the small number of realizations this observation should not be considered as statistically significant.

Exemplary configurations of the reconstructed networks are shown in figure 4.3. The displayed networks are chosen with (a) an average, (b) a small, and (c) a large number of connected components. In all three examples, the networks consist one large and several very small components. The largest components of these networks also exhibit a core-periphery structure, although less pronounced as in the evolved network in figure 3.7(c).

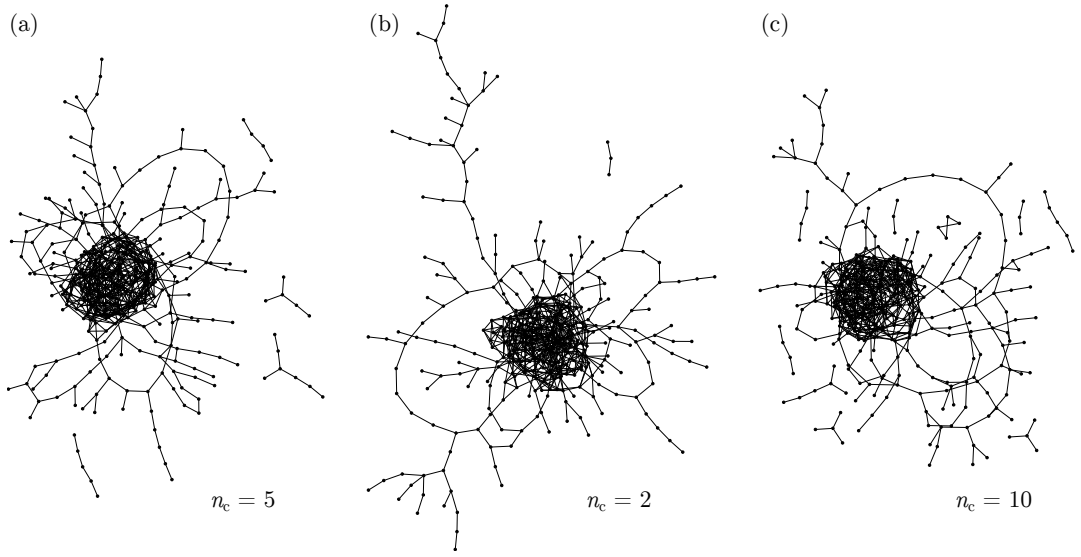

Figure 4.3. Typical network configurations taken from the individual reconstruction. The shown networks are representatives of configurations with (a) an average number of connected components $n_c = 5$, (b) a low number $n_c = 2$, and (c) a rather high number $n_c = 10$.

4.3. Reconstruction from Independently Averaged Correlations

In the previous section, the distributions $P(k)$, $P(j,k)$, and $T(k)$ were extracted as discrete frequencies $\hat{P}(k)$, $\hat{P}(j,k)$, and $\hat{T}(k)$ and directly fed into the reconstruction algorithm. The three distributions are, of course, highly correlated. So, it is worth to note that in this case the three discrete frequency distributions "fit" to each other, i.e., it is guaranteed that at least one network configuration with the given discrete distributions exists. A different approach is to first calculate average distributions from all the realizations of the evolved networks and then draw samples of discrete frequencies from these distributions. For this, the three distributions are treated separately here and the samples $\hat{P}(k)$, $\hat{P}(j,k)$, and $\hat{T}(k)$ are drawn independently from the averaged distributions. As before, the evolved networks are taken from the evolution of random networks towards the target spectral dimension $d_s^{(1)} = 1.4$. The degree distribution $P(k)$, the two-point correlations $P(j,k)$, and the distribution of triangle corner degrees $T(k)$ are calculated as normalized averages over the 100 realizations of the evolved networks. Also, the distribution of the total number of triangles $h(S)$ is calculated. From these distributions, 1000 samples are generated in the following way.

1. $N = 361$ vertex degrees k are drawn from $P(k)$ to form $\hat{P}(k)$.
2. $M = 722$ edges with end degrees (j,k) are drawn from $P(j,k)$ to form $\hat{P}(j,k)$.
3. A number of triangles S is drawn from $h(S)$.

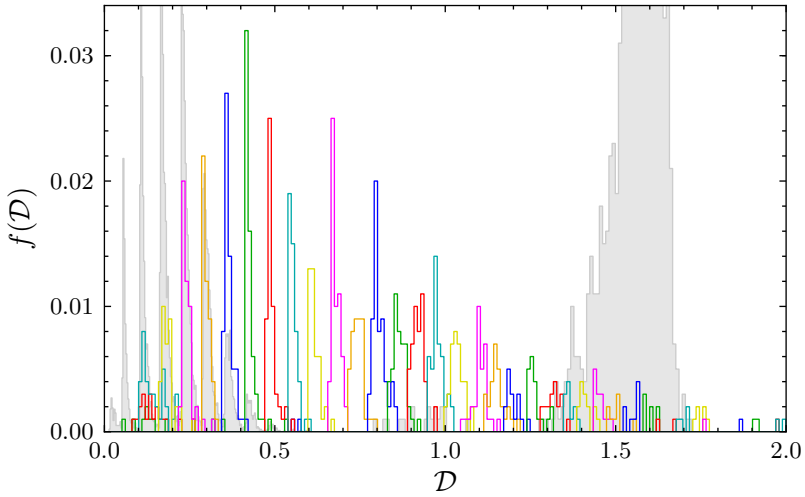

Figure 4.4. Histogram of spectral distances for the reconstruction from independently averaged correlations. The colored lines display the fraction $f(\mathcal{D})$ of reconstructed networks with spectral distance $\mathcal{D}$ broken down into classes of equal number of connected components n_c. The numbers range from $n_c = 1$ to $n_c = 39$, colors are repeated after every seventh curve. For comparison, the gray curves in the back shows the same for the individually reconstructed networks and a random network of the same size as in figure 4.1.

4. $3S$ triangle corner degrees k are drawn from $T(k)$ to form $\hat{T}(k)$.

These triples of $\hat{P}(k)$, $\hat{P}(j,k)$, and $\hat{T}(k)$ are used to construct one random network each. Of course, it is very likely that the correlations between the three distributions are not satisfied so that, in other words, the triple is not graphical. The algorithm will in this case generate random networks with only approximately the desired distributions.

Looking at the distribution of spectral distances in the reconstructed networks as shown in figure 4.4, a much broader spread than for the individual reconstructions can be observed. Nevertheless, the reconstructed networks on average still have a smaller spectral distance to the evolution target than in the reconstruction of a random network. Again, the distribution is multi-peaked and the separations are generated by different numbers of connected components in the reconstructed networks.

The distributions of the resulting numbers of connected components n_c for the individual reconstructions and the reconstruction from averaged correlation functions are compared in figure 4.5. The first observation is that the distribution of n_c is much broader in the latter case. As seen before, the number of components largely influences the spectral distance. Is this the only effect giving the networks reconstructed from the averaged correlation functions a higher spectral distance? To see this, figure 4.5 also shows the average spectral distance for each class of networks with equal number of connected components. Except for the networks with a small number of components, where only few realizations exist and the fluctuations are large, the spectral distances

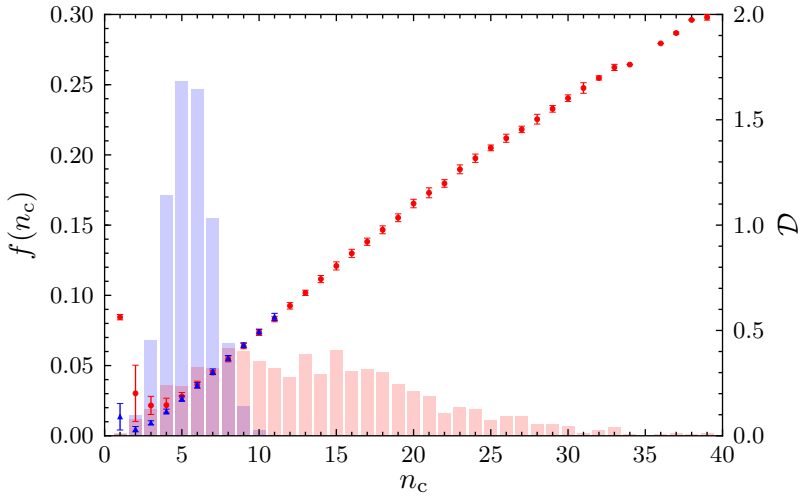

Figure 4.5. Distributions of number of connected components and average spectral distances by number of components. The histogram bars display the distribution $f(n_c)$ of the number of components n_c in the individually (blue bars) and from independently averaged distributions (red bars) reconstructed networks. The symbols indicate the average spectral distance of reconstructed networks with the respective number of components for the individually (blue triangles) and from independently averaged distributions (red circles) reconstructed networks. The error bars mark one standard deviation.

indeed appear very similar for the same numbers of connected components in both cases. The conclusion is that indeed the number of connected components is the major influence for the on average higher spectral distances here and not the choice of individual reconstruction or independently averaging the correlation functions.

Figure 4.6 shows three exemplary configurations of the reconstructed networks, again with (a) an average, (b) a low, and (c) a rather high number of connected components. As before, the networks seem to consist of one large and several very small components, out of which many are even single isolated vertices. In the largest component, again a core-periphery structure can be observed. But it seems less pronounced than before.

4.4. Reconstruction of Evolution Time Series

Thirdly, the relevance of the correlations in the evolutionary process shall be examined. To see if the degree distribution, two-point correlations, and degree-dependent clustering equally well characterize the network configurations at different stages of the evolution process, the reconstruction is applied to all intermediate steps of one exemplary evolution from a random graph towards the target spectral dimension of $d_s^{(1)} = 1.4$. As in section 4.2, the distributions are extracted as discrete frequencies $\hat{P}(k)$, $\hat{P}(j,k)$, and

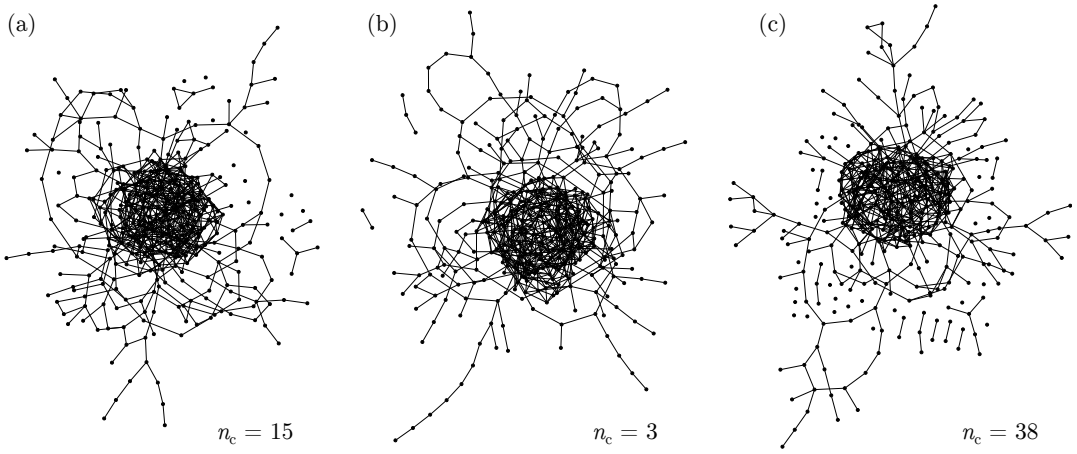

Figure 4.6. Visualization of typical network configurations reconstructed from independently averaged distributions. The shown networks are representatives of configurations with (a) an average number of connected components $n_c = 15$, (b) a low number $n_c = 3$, and (c) a rather high number $n_c = 38$.

$\hat{T}(k)$, now for each evolutionary time step. For each point in this time series, 100 samples of the random correlated networks are generated and analyzed.

The results of the time series reconstruction are summarized in figure 4.7 showing (a) the spectral distance $\mathcal{D}$ and (b) the coefficients r, C, and $\langle C_i \rangle$ of the reconstruction at each time step n of the evolution. The spectral distance of the reconstructed networks takes values in the same range as the evolving network until around 10^3 evolution steps. Afterwards, the reconstructed networks have significantly higher spectral distances to the evolution target than the evolving networks themselves. The assortativity coefficient r is very precisely reproduced in the reconstructed networks during the whole evolutionary process. Also the clustering coefficients C and $\langle C_i \rangle$ of the reconstructed networks follow the respective values of the evolving network. Their values are, however, always slightly higher. The fluctuations between the different realizations are negligible in the degree assortativity and rather small in both clustering coefficients. The spectral distance exhibits larger fluctuations at all time steps which additionally increase in the period between 10^2 and 10^3 evolutionary time steps where the largest drop in the spectral distance is observed. As the algorithm is designed to reproduce the given correlations in the generated random networks, it is no surprise that the correlation coefficients are reasonably well reproduced. Also the overrepresentation of triangles and the slightly larger fluctuations in the clustering can be attributed to the fact that the corresponding distribution is less precisely reproduced by the algorithm. Notably, no difference in the recreation of the two clustering coefficients is observed, although the distribution $\hat{T}(k)$ fed into the generation algorithm is describing the local clustering C_i and not the global clustering coefficient C.

Concluding this chapter, it was shown that, to a certain extent, the power-law scaling in the Laplacian spectrum is encoded in the degree distribution and degree correlations

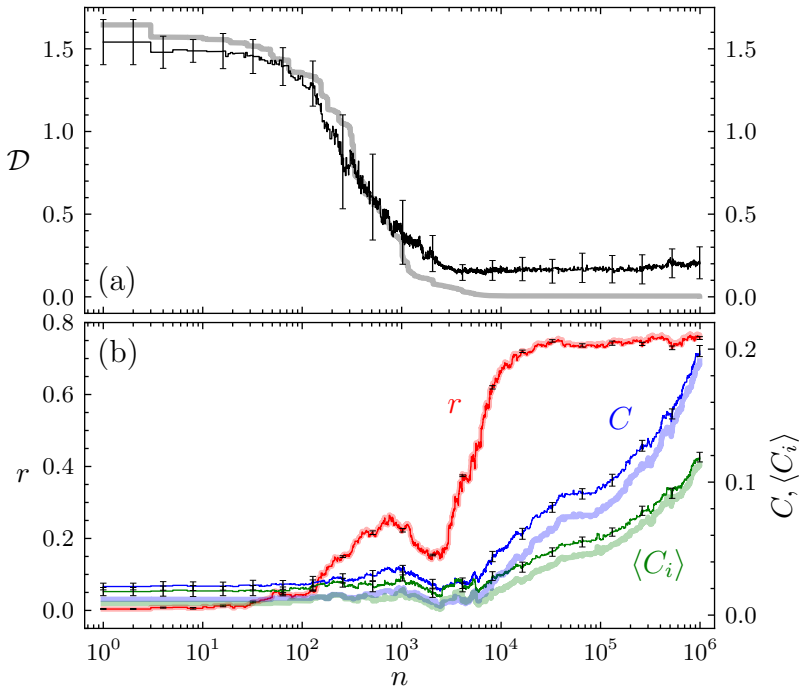

Figure 4.7. Reconstruction averages of evolutionary time series. Shown are averages of (a) the spectral distance $\mathcal{D}$ and (b) the assortativity coefficient r (left scale), the clustering coefficient C, and the mean local clustering coefficient $\langle C_i \rangle$ (both right scale) of reconstructed networks at each evolutionary time step n. The averages are calculated over 100 samples of the reconstruction and the error bars indicate one standard deviation. The light colored curves in the background display the time series of the respective quantities in the evolving network.

of the evolved networks. To this end, evolved networks were reconstructed from their degree distribution, two-point degree correlations, and degree-dependent clustering as random networks with the prescribed functions. The spectral densities of the evolved networks are surprisingly well reproduced up to one major issue. The algorithm mostly generates networks with several connected components. This is an inherent problem of all constructive methods of random network generation. To my knowledge, there is no way known to incorporate the number of connected components as additional constraint in the network generation. A large number of components produces a high degeneracy of the zero eigenvalue and, thus, results in a significant deviation from the target spectral density in the regime of low eigenvalues. No qualitative difference was observed between the individual reconstruction of evolved networks and the reconstruction based on independently averaged correlation functions. However, the independently drawn degree distributions and correlation functions are very likely to be non-graphical. This results in a worse performance of the random network algorithm such that the generated networks posses a high number of very small components. As mentioned before, this

produces an even larger deviation from the power-law scaling in the Laplacian spectrum. The reconstructed networks exhibit a core-periphery structure which seems, however, less pronounced than in the evolved networks themselves. The reconstruction of a whole evolution time series shows that the reconstructed networks are as close to the target density as the evolving networks up to a certain point in evolutionary time. Afterwards, a significant difference in the spectral distances is observed although the correlations are still well reproduced. This is an indication that at later stages of the evolution the large scale structure of the networks becomes more important.

5. Regular Evolved Networks and Symmetry-Based Coarse-Graining

This chapter extends the evolutionary optimization procedure of chapter 3 to k-regular networks, i.e., networks in which each vertex has the same degree k. The evolved networks exhibit an abundance of symmetric motifs which are arranged in loops and linear segments of different lengths. The underlying backbone structures are extracted by a systematic coarse-graining based on the network's symmetry. This allows for a full decomposition of the Laplacian spectrum of the network into the contributions from the symmetric motifs on the one hand and the coarse-grained backbone network on the other hand (Karalus and Krug, 2015).

5.1. Regular Evolved Networks

The anomalously diffusive networks found by the evolutionary optimization presented in chapter 3 show very heterogeneous structures. They exhibit a bimodal degree distribution with large fractions of vertices above and below but only few vertices around the mean degree. Additionally, various signs of a segregation of the high- and low-degree vertices into rather densely connected cores on the one hand and sparse, mostly tree-like, peripheries on the other hand have been observed. This is supported by visual inspection of the evolved networks. An interesting follow-up question is whether it is also possible to construct networks with a similar dynamical behavior but which are more homogeneous in their structure. If possible, what are the structural properties of those networks? To elucidate this issue, the evolutionary optimization can be performed on networks that are more homogeneous by construction. This is done here by the introduction of degree regularity as an additional constraint in the evolution. Starting the evolution from k-regular random graphs or lattices and performing only structural mutations that conserve the degree of each vertex results in an evolutionary process satisfying this constraint.

The goal of this evolution is to search the space of connected k-regular networks with given size N and M for networks with an approximate power law in the Laplacian spectrum, $I(\lambda) \propto \lambda^{d_\mathrm{s}/2}$. The evolution target is, as before, the logarithmically integrated spectral density (3.13) with given spectral dimension $d_\mathrm{s}^{(1)} = 1.4$ or $d_\mathrm{s}^{(2)} = 1.1$. The spectral distance (3.12) is minimized by an adaptive walk. For two different choices of k the evolutions are started from square lattices ($k = 4$) and honeycomb lattices ($k = 3$) with periodic boundary conditions as well as corresponding 4- and 3-regular random networks. All the 4-regular networks have $N = 361$ vertices and $M = 722$ edges, while for the 3-regular networks $N = 360$ and $M = 540$. The conservation of the vertex degrees

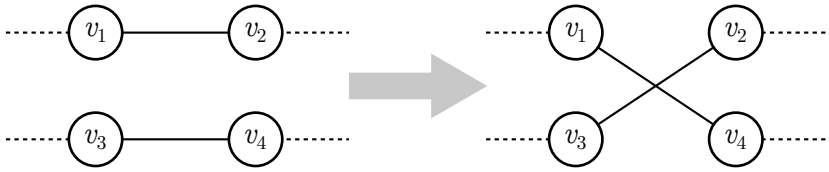

Figure 5.1. Schematic of the edge-crossing update. Two edges v_1–v_2 and v_3–v_4 are crossed to v_1–v_4 and v_3–v_2 as proposed new state. All other edges incident to the vertices $v_1, v_2, v_3, v_4 \in \mathcal{V}$—as indicated by the dashed lines—remain unchanged.

in the course of the evolution is realized by an "edge-crossing" update as sketched in figure 5.1. In each evolution step, two randomly chosen edges v_1–v_2 and v_3–v_4 are crossed to v_1–v_4 and v_3–v_2. The only important restrictions are that all vertices $v_1, v_2, v_3, v_4 \in \mathcal{V}$ have to be distinct and that the new edges v_1–v_4 and v_3–v_2 do not exist before the update. As in chapter 3, updates are rejected directly if they result in a separation of the network in several connected components. In fact, also the k-regular random networks are most easily constructed by randomization of the respective lattices, that means, by an evolution using the edge-crossing update without any selection, i.e., accepting each proposed mutation. This corresponds to an infinite-temperature MC simulation, so all possible states are sampled with equal probability. The evolutions are, again, run for 10^6 evolutionary time steps. Exemplary configurations of the evolved networks are displayed in figure 5.2(a) for a 4-regular and (b) for a 3-regular network. In contrast to most random and the evolved networks in figure 3.7, their structural properties seem to be visually accessible in this planar representation. The networks appear rather homogeneous without any regions of clearly distinguished structures. They exhibit a peculiar modular structure of small symmetric motifs arranged into—probably fractal—loops of different lengths. Since similar fractal-like or comb structures are known to generate anomalous diffusion (Havlin and ben-Avraham, 2002) the obvious question is whether the observed backbone structures can be related to the Laplacian spectrum and, hence, the diffusion behavior. In general it is not possible to construct the Laplacian spectrum of a network from its subnetworks or underlying structures. There is, however, an exception if the network is symmetric. In this case, the symmetric structures can be coarse-grained and the Laplacian spectrum decays into the spectra of the resulting backbone and the symmetric motifs.

In the following, this symmetry-based coarse-graining is applied to the regular evolved networks. For a visual impression, the resulting backbone structures, called network quotients, of the exemplary networks in figure 5.2(a) and (b) are shown in figure 5.2(c) and (d), respectively. The Laplacian spectrum and eigenvectors of such a network as a whole can be decomposed in the spectra of the quotient and the symmetric motifs. In order to be more specific, some background from algebraic graph theory on network symmetries has to be introduced first.

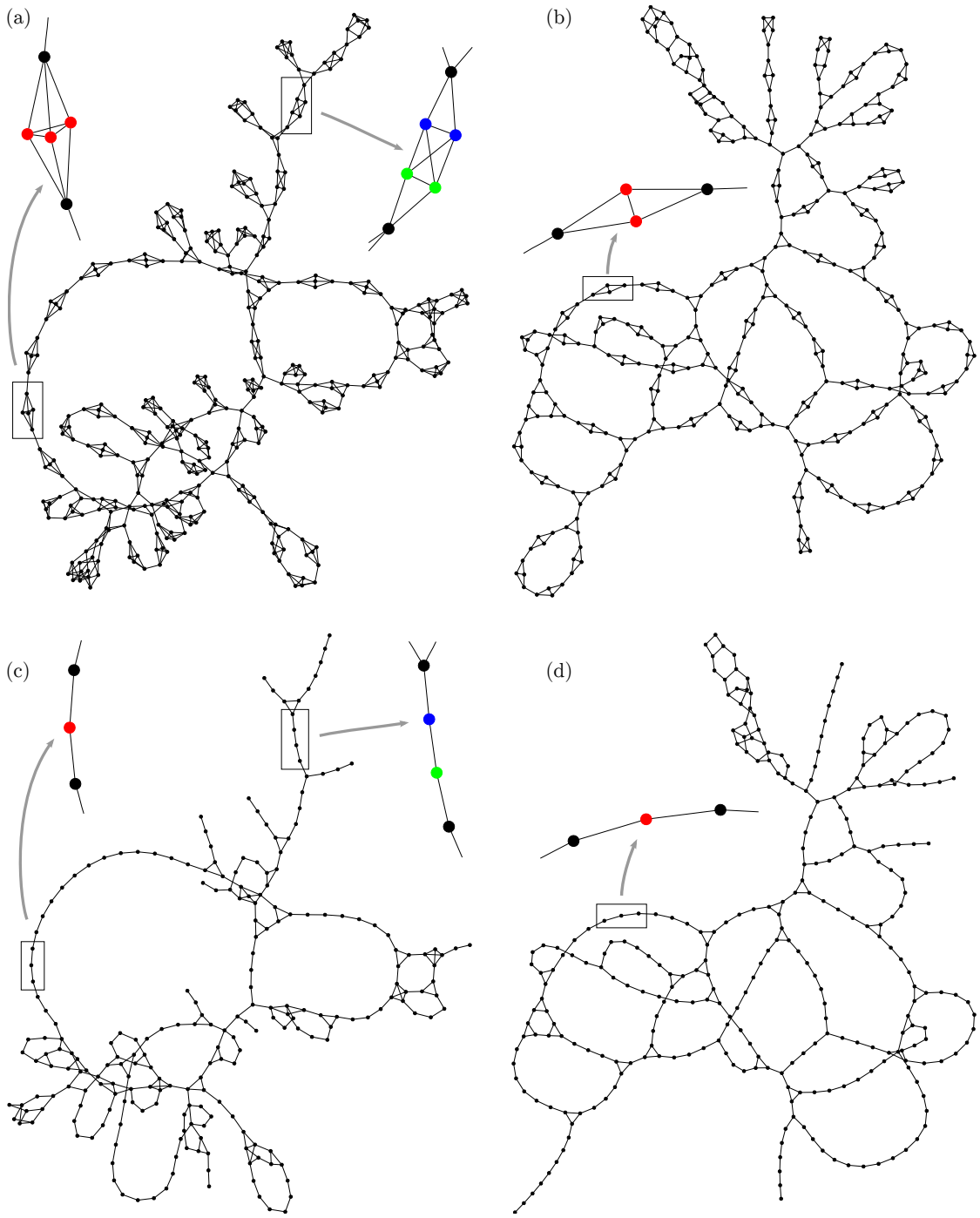

Figure 5.2. Typical configurations of k-regular networks evolved towards a target spectral dimension of $d_{\mathrm{s}}^{(1)} = 1.4$ starting from (a) a square lattice and (b) a honeycomb lattice. The corresponding s-quotients are shown in (c) and (d). In the enlarged segments, the colors indicate the orbits to which the respective vertices belong.

5.2. Symmetries in Networks

A network being symmetric means that some of its vertices can be permuted without changing the network structure. This means that the vertices are interchangeable and play the same role in the network's function. They build up a redundancy in the network. In this section, a summary of the basic concepts and results on network symmetries is given. More details and proofs can be found in the literature (Godsil and Royle, 2004; MacArthur and Sánchez-García, 2009; MacArthur *et al.*, 2008).

A network *automorphism* is a permutation of vertices in the network that does not alter the adjacency of the network. The set of all automorphisms of a network G forms a group, the automorphism group $\mathrm{Aut}(G)$, compactly describing all symmetries of the network. A network is called *symmetric* if it has a non-trivial automorphism group. The set of vertices which are "moved" by an automorphism $p \in \mathrm{Aut}(G)$ is called the support of p, $\mathrm{supp}(p) = \{i \in \mathcal{V} \mid p(i) \neq i\}$. Two sets of automorphisms P and Q are called support disjoint if all possible pairs of $p \in P$ and $q \in Q$ have disjoint supports. In this case, P and Q act independently on the network, $pq = qp$ for all $p \in P$ and $q \in Q$.

This notion of independence is used to define the symmetric motifs of a network. Let $\mathrm{Aut}(G)$ be generated by the set $S = S_1 \cup \cdots \cup S_k$ with all pairs of S_i being support-disjoint and minimal in the sense that each S_i cannot be further decomposed into support-disjoint subsets. Then the subgroups H_i generated by the S_i form a direct product decomposition of the automorphism group,

$$\mathrm{Aut}(G) = H_1 \times H_2 \times \cdots \times H_k \,. \tag{5.1}$$

This decomposition is called the *geometric decomposition* of the automorphism group and it can be shown that it is unique (MacArthur *et al.*, 2008). The H_i are called *geometric factors*. A *symmetric motif* $\mathcal{M}_H$ is the induced subgraph on the support of a geometric factor H, i.e., the subgraph consisting of the vertices in the support of H together with all edges of G that have both ends in that set. Intuitively speaking, a symmetric motif $\mathcal{M}_H$ is a minimal subnetwork whose vertices are moved by H. Note that the definition of a symmetric motif is much more restrictive than the definition of motifs by Milo *et al.* (2002). This is, however, necessary for the relation to the network spectra.

An automorphism permutes the vertices of a network without changing the network structure. It can be viewed as mapping the vertices "onto each other". This provides a partition of the vertex set $\mathcal{V}$ into disjoint classes called *orbits*. The orbit a vertex $i \in \mathcal{V}$ belongs to is the set of vertices $\Delta(i) \subset \mathcal{V}$ to which i is mapped under the action of $\mathrm{Aut}(G)$,

$$\Delta(i) = \{g(i) \in \mathcal{V} \mid g \in \mathrm{Aut}(G)\} \,. \tag{5.2}$$

In the enlarged segments of figure 5.2(a) and (b) the vertex colors indicate examples of how different vertices are grouped into orbits. Note that in order to map the vertices of an orbit onto each other without altering the adjacency it may be necessary to move a larger fraction of vertices at the same time. The vertices in the same orbit are interchangeable so they are structurally completely equivalent. Therefore, the orbits describe the redundancy in a network.

This structural equivalence is exploited in the coarse-graining to a network quotient. If a graph G contains r distinct orbits, $\mathbf{\Delta} = \{\Delta_1, \Delta_2, \ldots, \Delta_r\}$, then its *quotient graph* $Q = G/\mathbf{\Delta}$ has r vertices. The vertices of Q are the orbits of its parent graph G. Due to the structural equivalence of the vertices in the same orbit the number of neighbors in Δ_y of vertex $i \in \Delta_x$, called q_{xy}, does not depend on the choice of i but only on x and y. This property makes the orbit partition *equitable* which is important for the relation between the spectra of Q and G. The quotient graph Q has the vertex set $\mathcal{V}(Q) = \mathbf{\Delta}$ and an adjacency matrix with elements $\{q_{xy}\}_{x,y=1,\ldots,r}$. The q_{xy} are not necessarily symmetric and the diagonal elements q_{xx} may be non-zero. Hence, the quotient graph, even of a simple parent graph, is directed and generally contains multiedges and self-loops.

The networks in figure 5.2(c) and (d) actually do not show the quotients of the evolved networks but their underlying simple networks. All directions, multiedges, and self-loops have been left out. The resulting network is called simplified quotient or *s-quotient* Q_s. It already captures essential properties such as the size and the underlying adjacency of the quotient.

A key result from algebraic graph theory establishes a relation between the adjacency spectra and eigenvectors of a network G and its quotient G/π from any equitable partition π. Here, the interest lies, of course, in the orbit partition $\mathbf{\Delta}$ and the corresponding quotient $Q = G/\mathbf{\Delta}$. All eigenvalues of the quotient are also in the spectrum of the parent. Furthermore, each eigenpair $(\mu, \boldsymbol{v} = (v_1, \ldots, v_r))$ of Q has a corresponding eigenpair $(\mu, \overline{\boldsymbol{v}} = (\bar{v}_1, \ldots, \bar{v}_N))$ of G with $\bar{v}_i = v_x$ for all $i \in \Delta_x$, i.e., the eigenvectors are constant v_x on each orbit Δ_x. These eigenvectors are called *lifted* from the eigenvectors of the quotient. The symmetric motifs of G provide the remaining eigenpairs of G. An eigenvector $\boldsymbol{v}^* = (v_1^*, \ldots, v_m^*)$ of the symmetric motif $\mathcal{M}$—isolated from the rest of the network—is called *redundant* if $\sum_{i\in\Delta} v_i^* = 0$ for each orbit $\Delta \in \mathcal{M}$. The corresponding eigenvalues are also in the spectrum of G. Moreover, for each redundant eigenpair $(\mu, \boldsymbol{v}^*)$ of a symmetric motif $\mathcal{M}$ there exists an eigenpair $(\mu, \hat{\boldsymbol{v}} = (\hat{v}_1, \ldots, \hat{v}_N))$ of G with $\hat{v}_i = v_i^*$ for all $i \in \mathcal{M}$ and $\hat{v}_i = 0$ for all $i \notin \mathcal{M}$, i.e., the eigenvectors are localized on the symmetric motifs. Summarizing these statements, the eigenvalues and the corresponding eigenvectors of the adjacency matrix of a symmetric network G completely divide into two classes:

(1) The eigenvalues of the quotient of G are also eigenvalues of G itself. The corresponding *lifted eigenvectors* are constant on each orbit by assigning the value of each orbit in the eigenvector of the quotient to all vertices belonging that orbit.

(2) The redundant eigenvalues of each symmetric motif of G are also eigenvalues of G itself. The corresponding *redundant eigenvectors* of the isolated motifs are also eigenvectors of G localized on the motifs.

For the adjacency spectrum and eigenvectors, these results are elaborated and proven by MacArthur and Sánchez-García (2009). For this study, similar statements for the Laplacian spectrum and eigenvectors are needed.

5.2.1. Coarse-Graining Relations for the Laplacian Spectrum

In this section, the proof for the relations between the adjacency spectra and eigenvectors of a symmetric network on the one hand and its quotient and symmetric motifs on the other hand by MacArthur and Sánchez-García (2009, appendix A) is extended to the case of the graph Laplacian.

Consider a network G with N vertices together with a partition $\pi = \{C_1, \ldots, C_r\}$ of its vertices. Here, the indices i, j will number the vertices of G and, hence, run from 1 to N. The indices x, y run from 1 to r and denote the cells of the partition. The partition π is *equitable* if the number of neighbors in C_y of any vertex in C_x is a constant $\tilde{A}_{xy}$. The quotient $Q = G/\pi$ is a multidigraph with r vertices corresponding to the r cells of the partition π. The adjacency matrix $\tilde{\boldsymbol{A}}$ of the quotient has the elements $\tilde{A}_{xy}$. The characteristic matrix $\boldsymbol{P}$ of the partition π is an $N \times r$ matrix with entries $P_{ix} = 1$ if vertex $i \in C_x$ and $P_{ix} = 0$ otherwise. In a column notation $\boldsymbol{P} = (\boldsymbol{w}_1 | \ldots | \boldsymbol{w}_r)$ the characteristic matrix is formed by the vectors $\boldsymbol{w}_x \in \{0,1\}^N$ with entries 1 in the vertices belonging to C_x and 0 otherwise. MacArthur and Sánchez-García showed that

$$\boldsymbol{AP} = \boldsymbol{P}\tilde{\boldsymbol{A}} \tag{5.3}$$

which means that the space W spanned by the column vectors of $\boldsymbol{P}$ is $\boldsymbol{A}$-invariant, i.e., $\boldsymbol{Au} \in W$ for all $\boldsymbol{u} \in W$. At this point, two statements from linear algebra are needed (e.g. Godsil and Royle, 2004). (i) Every non-zero $\boldsymbol{A}$-invariant subspace has an orthonormal basis of eigenvectors of $\boldsymbol{A}$. (ii) The orthogonal complement of an $\boldsymbol{A}$-invariant subspace is also $\boldsymbol{A}$-invariant. Hence, $\mathbb{R}^N = W \oplus W^\perp$ and there are orthogonal bases of W and $W^\perp$ consisting of eigenvectors of $\boldsymbol{A}$. The proof finishes with the observation that

1. $\dim(W) = r$,
2. $\boldsymbol{u} \in W \Leftrightarrow \boldsymbol{u}$ is constant on each C_x,
3. $\boldsymbol{u} \in W^\perp \Leftrightarrow$ the sum of coordinates of $\boldsymbol{u}$ is zero on each C_x.

Consequently, a basis of eigenvectors was found such that the first r eigenvectors are constant on each orbit C_x and the last $N - r$ are redundant, i.e., their coordinates sum to zero on each orbit.

In order to extend this proof to the spectrum of the graph Laplacian $\boldsymbol{L}$, it remains to shown that W is also $\boldsymbol{L}$-invariant. This is true if and only if there exists a matrix $\tilde{\boldsymbol{L}}$ such that

$$\boldsymbol{LP} = \boldsymbol{P}\tilde{\boldsymbol{L}}\,. \tag{5.4}$$

The matrix $\tilde{\boldsymbol{L}}$ is then the graph Laplacian of the quotient network Q. Let $\tilde{\boldsymbol{L}} = \tilde{\boldsymbol{D}} - \tilde{\boldsymbol{A}}$ where $\tilde{D}_{xy} = \tilde{k}_x \delta_{xy}$ with $\tilde{k}_x = \sum_y \tilde{A}_{xy}$. Then, with equation (5.3)

$$\boldsymbol{LP} = \boldsymbol{DP} - \boldsymbol{AP} = \boldsymbol{DP} - \boldsymbol{P}\tilde{\boldsymbol{A}}\,. \tag{5.5}$$

It remains to show that $\boldsymbol{DP} = \boldsymbol{P}\tilde{\boldsymbol{D}}$. In components, the left hand side reads

$$(\boldsymbol{DP})_{ix} = \sum_j D_{ij} P_{jx} = \sum_j k_i \delta_{ij} P_{jx} = k_i P_{ix} \tag{5.6}$$

which is the degree of vertex i if $i \in C_x$ and zero otherwise. The right hand side is

$$(\boldsymbol{P}\tilde{\boldsymbol{D}})_{ix} = \sum_y P_{iy}\tilde{D}_{yx} = \sum_y P_{iy}\tilde{k}_y\delta_{yx} = P_{ix}\tilde{k}_x = P_{ix}\sum_y \tilde{A}_{xy}\,. \tag{5.7}$$

Remember that $\tilde{A}_{xy}$ was defined to be the number of neighbors in C_y of a vertex from C_x. Hence, summation over all y yields the total number of neighbors a vertex in C_x has. Equation (5.7), thus, evaluates to the degree of vertex i if $i \in C_x$ and zero otherwise. This is equal to the right hand side in equation (5.6) which completes the proof that W is also $\boldsymbol{L}$-invariant.

For the case of the orbit partition $\pi = \boldsymbol{\Delta}$ discussed in section 5.2, $\tilde{A}_{xy} = q_{xy}$. Note that the graph Laplacian $\tilde{\boldsymbol{L}}$ of the quotient as defined here is the same as the out-degree Laplacian $\boldsymbol{L}^{\text{out}}$ defined in section 2.3.2. Hence, the way how $\tilde{A}_{xy}$ is defined here corresponds to the previously used convention that A_{ij} stands for a directed edge from i to j.

5.3. Application to Regular Evolved Networks

After 10^6 evolutionary steps the regular evolved networks, as before, exhibit no traces of the initial conditions, random network or lattice, any more. Also in this case, the conclusion is that the optimization "landscape" seems to be sufficiently smooth so that even the simple evolutionary algorithm succeeds to find network topologies matching the target power-law scaling in the spectrum reasonably well. The networks depicted in figure 5.2 for the case of (a) 4-regular and (b) 3-regular initial states represent typical outcomes of the evolution. In these evolved networks, frequently reappearing small symmetric motifs are encountered on small length scales. On larger scales, these motifs are apparently arranged to form loops and linear chains of different lengths. In the two cases of $k = 4$ and $k = 3$, the motifs are, of course, different. The way how they are arranged, however, seems to be very similar in both cases. This becomes even more clearly visible in the s-quotients of the evolved networks shown in figure 5.2(c) and (d). Here, the backbones of the evolved networks clearly exhibit structures of linear segments arranged in loops and chains of different lengths. The three enlarged motifs from figure 5.2(a) and (b) are shown separately in figure 5.3. The vertices are labeled by the redundant Laplacian eigenvectors of the motifs with the respective eigenvalues annotated. As can be seen, the redundant Laplacian eigenvalues of the abundant motifs in the 4-regular evolved networks are all equal to 5, and equal to 4 in the 3-regular networks. Thus, a high degeneracy of those eigenvalues in the spectra of the networks as a whole is expected.

The way how the coarse-grained backbone structures are constructed is sketched in figure 5.4. First, the quotient of the evolved network by its orbits is formed according to the procedure explained in section 5.2. Then the s-quotient is simply the underlying simple network of the quotient. Table 5.1 summarizes the change of the network sizes and spectral distances in the coarse-graining procedure. Remember that the numbers of the spectral distance cannot be compared directly for different network sizes N. In order

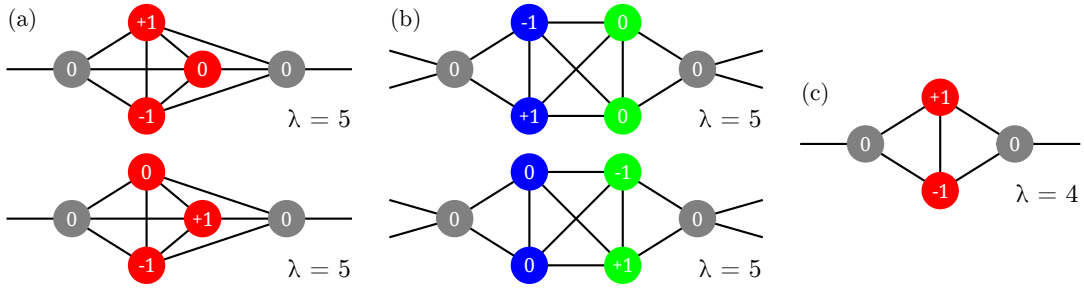

Figure 5.3. Redundant Laplacian eigenvectors of the most prominent motifs in the (a)/(b) 4-regular and (c) 3-regular evolved networks. The orbits are colored as in figure 5.2. The corresponding redundant eigenvalues are annotated.

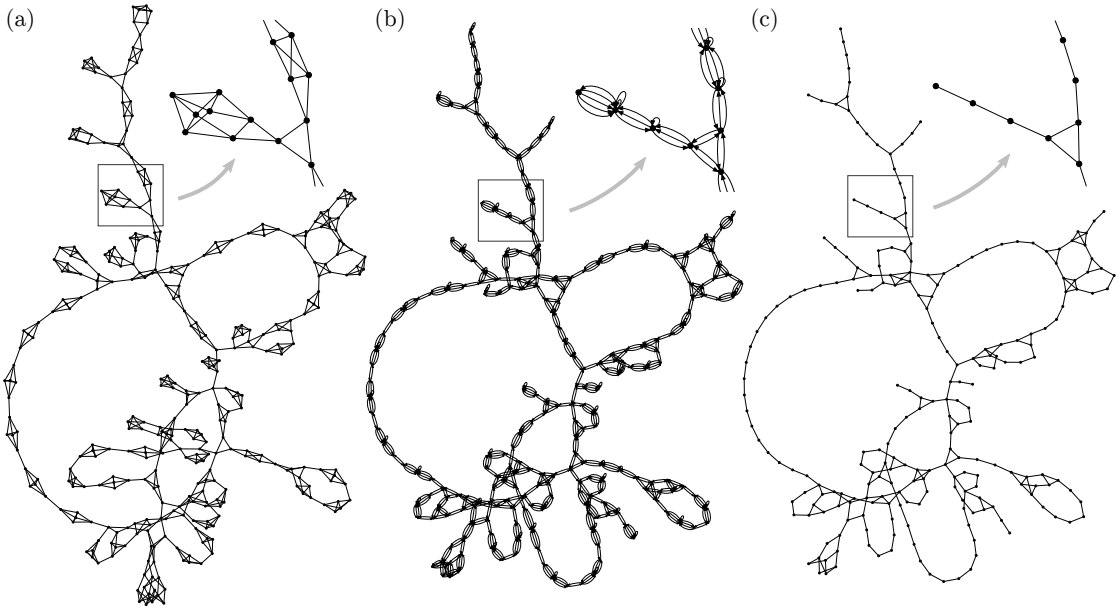

Figure 5.4. Illustration of the coarse-graining procedure from (a) the evolved network as shown in figure 5.2 via (b) the quotient of the evolved network constructed by the procedure explained in section 5.2 to (c) the underlying simple network of the quotient (s-quotient). For a better visibility of the detailed structure the boxed segment is additionally shown enlarged in the three networks.

	4-regular evolution			3-regular evolution		
Network	N	M	$\mathcal{D}$	N	M	$\mathcal{D}$
regular evolved	361	722	0.0507(3)	360	540	0.02400(1)
quotient	229(6)	914(25)	0.09(1)	281(6)	842(19)	0.035(5)
s-quotient	229(6)	302(17)	0.03(2)	281(6)	330(10)	0.030(2)
randomized s-quotient	229(6)	302(17)	0.39(9)	281(6)	330(10)	0.27(6)
evolved s-quotient	229(6)	302(17)	0.0045(1)	281(6)	330(10)	0.0045(1)
unrestricted evolved	361	722	0.0046(3)	–	–	–

Table 5.1. The coarse-graining in numbers. Average numbers of vertices N and edges M and average spectral distance $\mathcal{D}$ for the coarse-graining of the 4-regular evolved (left) and 3-regular evolved (right) networks together with the numbers for the respective quotients, s-quotients, and randomized as well as evolved networks of the same sizes as the s-quotients ("randomized/evolved s-quotient"). For comparison, the values for the unrestricted evolution (see table 3.1) are repeated in the last row. The numbers in parentheses denote one standard deviation.

to get an idea of the absolute numbers of the spectral distance of the quotients and s-quotients, they are compared with the spectral distance of networks having the same size as the s-quotients but which are fully randomized ("randomized s-quotient") and then evolved without the restriction of regularity ("evolved s-quotient"). To this end, random networks with the same numbers of vertices and edges are constructed by randomization (keeping the network as a whole connected) and then the evolutionary optimization as described in chapter 3 is applied. Looking at the precise figures, it turns out that the networks evolved under the regularity constraint have a final spectral distance of about one order of magnitude higher than those without the additional constraint. The quotients as well as the s-quotients have spectral distances around the same order while the unrestricted evolution of those succeeds to lower the spectral distance again by about an order of magnitude.

To see what the numbers of the spectral distance mean, figure 5.5 shows the Laplacian spectra as logarithmically integrated densities of the two exemplary evolved regular networks from figure 5.2(a) and (b) together with the spectra of their quotients, s-quotients, and a network of the same size as the s-quotient evolved without the restriction of degree-regularity ("evolved s-quotient"). Note that although the Laplacian of the quotient is not symmetric all of its eigenvalues turn out to be real. As predicted by the theory of symmetric networks presented in section 5.2, the Laplacian spectrum of the quotient is a subset of the spectrum of its parent. Especially in the region of small eigenvalues the integrated densities of the evolved networks and their quotients are exactly parallel. Thus, the small eigenvalues, reflecting the large-scale structure of the network, are the same in both cases. To the right hand side of the plots the integrated densities of the evolved networks in both cases exhibit a clearly visible large step indicating a high degeneracy of certain eigenvalues. As indicated by the vertical gray lines, the degenerate eigenvalues are $\lambda = 5$ for the 4-regular evolved network and $\lambda = 4$ for

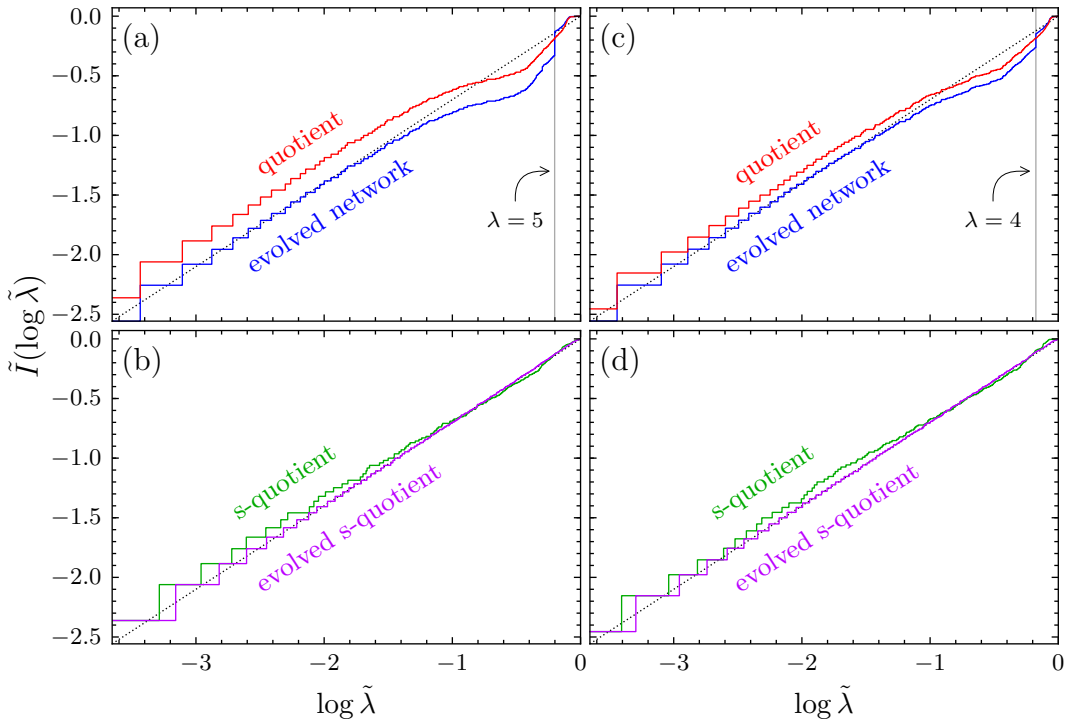

Figure 5.5. Logarithmically integrated Laplacian eigenvalue densities of, on the left hand side (a) a 4-regular evolved network together with its quotient and (b) the resulting s-quotient as well as an evolved network of the same size as the s-quotient ("evolved s-quotient"). On the right hand side: The same for (c) a 3-regular evolved network and its quotient, (d) the corresponding s-quotient and an evolved network of the same size. The two vertical gray lines indicate the position of the eigenvalues 5 and 4 in the logarithmically normalized representation.

the 3-regular case. These are exactly the redundant eigenvalues of the symmetric motifs shown in figure 5.3 and their abundance explains the high degeneracy. In the coarse-graining from the evolved networks to the quotients, the degeneracy is largely reduced resulting in an overall shift of the integrated densities. This observation is in agreement with the disappearance of the symmetric motifs. For the spectra of the s-quotients no rigorous mathematical relation to the spectra of the parent networks exits. Nevertheless, the s-quotients seem to fit the target power-law spectrum even better than the quotients. The values of the spectral distance measure the s-quotients attain are about one order of magnitude lower than those of the random networks of the same size (see table 5.1). This strongly supports the conclusion that backbone structures represented by the s-quotients are truly a highly relevant structural feature for the power-law scaling in the Laplacian spectrum. Further optimization of the s-quotients by the unrestricted evolution lowers the spectral distance significantly. From the visual appearance of the spectra, however, these further corrections seem to be secondary fine-tuning of the eigenvalues.

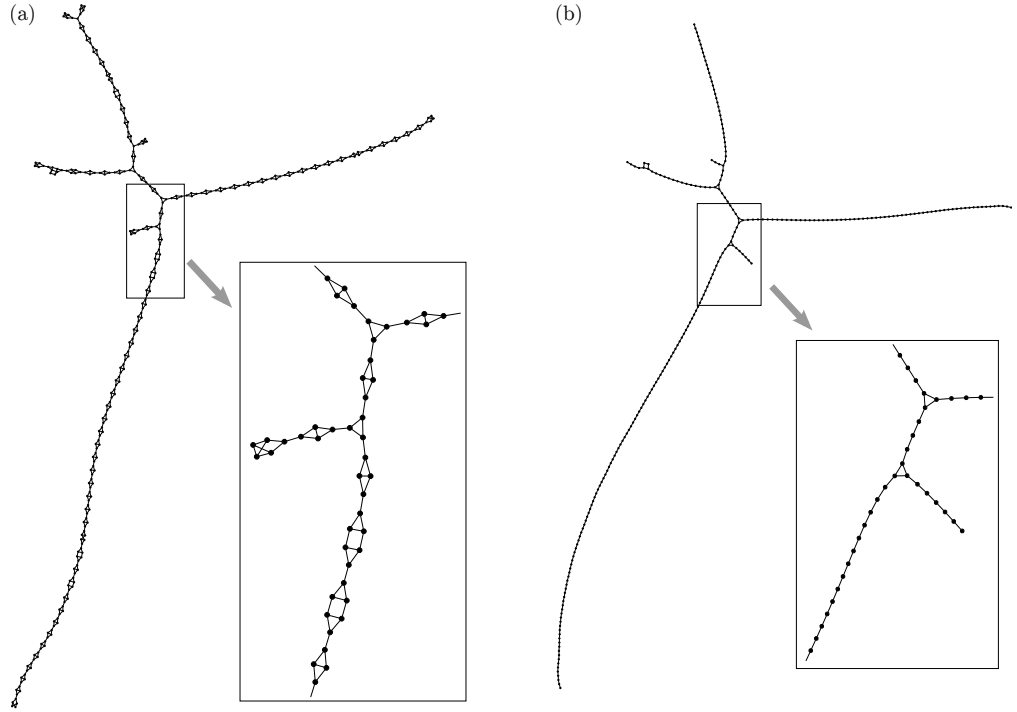

Figure 5.6. (a) Typical 3-regular evolved network and (b) corresponding s-quotient for a target spectral dimension of $d_s^{(2)} = 1.1$. For a better visibility of the details, the boxed segments are additionally shown enlarged.

The evolution towards the second target $d_s^{(2)} = 1.1$ is, again, expected to be more difficult. A typical configuration of an evolved 3-regular network towards $d_s^{(2)} = 1.1$ is depicted in figure 5.6(a). Again, the network appears to be set up by a few symmetric motifs as building blocks. In this case, these building blocks are arranged in even longer chains than before with almost no loops. The corresponding coarse-grained backbone structure in form of the s-quotient is shown in figure 5.6(b). The long linear segments with very few branchings indeed give the impression of an "almost 1-dimensional" structure.

Turning back to the question *why* the optimization process evolves the networks into the observed structures, a possible explanation is that there are too many edges present to form homogeneous structures of lower (spectral) dimension. The prescribed power-law Laplacian spectra can be approximately achieved by the backbone structures as seen in the s-quotients but the proportions of vertices and edges do not match with the numbers in the evolving networks. A possibility to nevertheless generate those structures on large scales is to "hide" the excess edges inside the symmetric motifs and build up the linear segments out of those building blocks. This agrees with the observation that in the 4-regular evolved networks the coarse-graining reduces the network size more than in the 3-regular case where from the beginning less edges are present. A first approach to

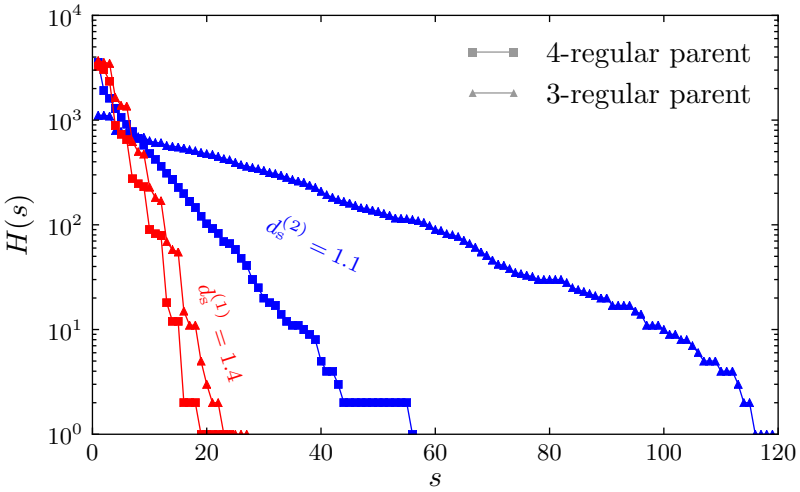

Figure 5.7. Histogram of linear segment lengths s in the s-quotients of 4-regular (square markers) and 3-regular (triangle markers) evolved parent networks for target spectral dimensions $d_s^{(1)} = 1.4$ (red) and $d_s^{(2)} = 1.1$ (blue). Shown are cumulative histograms $H(s) = \int_s^\infty h(t)\,\mathrm{d}t$ of the absolute frequencies $h(s)$ for 100 realizations each.

a quantitative description of the backbone structure is to extract the lengths of linear segments in the s-quotients of the evolved networks. Figure 5.7 shows the absolute frequencies of the linear segment lengths s in cumulative histograms. Rather broad distributions are observed which are much broader for the lower value of the spectral dimension. This supports the idea that the quantity might be a relevant one to quantify the structures of the evolved networks. The networks, however, appear too small for a real systematic scaling analysis of this distribution.

Summarizing this chapter, the evolutionary optimization algorithm introduced before was adapted to evolve k-regular networks towards given non-trivial values of the spectral dimension. By construction, these networks have to be more homogeneous than the previously found evolved networks which exhibited various signs of a core-periphery structure. The evolved networks indeed appear much more homogeneous and their overall structure seems to be surprisingly accessible just by the visual appearance. The networks consist mainly of frequently reappearing small symmetric motifs which are arranged in loops of varying length. A coarse-graining procedure known from algebraic graph theory based on the symmetry properties of the networks can be applied in this case. It was shown how the Laplacian spectrum separates into contributions from the symmetric motifs on the one hand and the underlying backbone structures, the network quotients, on the other hand. The quotients are formed by loops and linear segments of different lengths and appear to build the basis for the power-law scaling of the Laplacian spectrum. For the lower target spectral dimension, the local motif structure seems very similar. Almost no loops but even longer linear chains forming “almost 1-dimensional” quotient networks are observed resulting in a broad distribution in the linear segment

lengths which seems to be a main structural characteristic for the anomalous diffusion behavior in this case.

6. Summary and Discussion

The work presented in this thesis investigates the relation between structure and dynamics in complex networks. It addresses the important question how the network structure shapes the global behavior of dynamical processes. This behavior is rather generally determined by the spectral properties of the network, i.e., the eigenvalues and eigenvectors of the matrix describing the (linearized) dynamical interactions. The focus here lies on Laplacian dynamics, the class of processes in which the graph Laplacian operator describes the time evolution of the dynamics. This class comprises very important basic processes such as diffusion and random walks or the synchronization of oscillators. In diffusion or random walk processes the average probability of a random walker or diffusing particle to return to its origin is given by the Laplace transform of the eigenvalue density of the graph Laplacian. Thus, a power-law scaling with given exponent in the Laplacian spectrum results in power-law decay in the average return probability with the same exponent. In this case, the exponent is the so-called spectral dimension of the network. In general, little is known exactly about the relation between the structural characteristics of a network and the behavior of dynamics. In particular, no constructive method exists to generate networks with a prescribed Laplacian spectrum. Therefore, other strategies like evolutionary search in the configuration space have to be applied in order to find such networks. This approach was pursued here to construct networks with a prescribed power-law scaling in the Laplacian spectrum.

In the first part, a method was developed to evolve networks towards a target integrated Laplacian spectral density. For this, a spectral distance function was introduced measuring the difference between two spectra as integral over the squared difference between the logarithmically integrated eigenvalue densities. The advantage of this spectral distance measure is that it can be directly applied to quantify the distance of a specific spectrum from another spectrum as well as from a generic function. Inspired by biological evolution or Monte Carlo simulations, the simplest optimization strategy is an adaptive walk in which mutations occur at random and only beneficial mutations are accepted. It always terminates at a local optimum, so the efficiency of this strategy strongly depends on the structure of the optimization problem, the "optimization landscape". Starting from regular lattices and random network realizations, networks were evolved towards a power-law scaling in their Laplacian spectra with two exemplary non-trivial values of the spectral dimension characterizing anomalous diffusion behavior. The two values of the target spectral dimension were chosen such that the first one is close to the spectral dimension of a known deterministic fractal, the Sierpinski triangle, which has the same average degree as the evolving networks. For the second target spectral dimension this is not the case so it is expected to be more difficult to achieve. The adaptive walk simulations were successful in the sense that the evolution finds networks

with Laplacian spectra reasonably close to the power-law target function. No signs of significant deviations between the individual realizations were observed, an indication for a rather simple "spectral distance landscape". Structurally, the evolved networks exhibit bimodal degree distributions, high degree assortativity, and clustering. Taken together, these properties are signs of structural heterogeneity, a segregation of the vertices into densely connected cores and rather sparse peripheries. Anomalous diffusion is well known to be generated by deterministic or random fractals, which are both mainly characterized by self-similar structures. One might have expected related properties in the evolved networks here. Instead, the heterogeneous core-periphery structures seem to be the distinctive characteristic of the evolved networks. The question whether the overall anomalous diffusion behavior results simply as an average of the diffusion in the two regions or if the interface, the way how the two regions are connected, plays an important role cannot be answered at this point. It could be an interesting question in a study building upon this work.

Secondly, the influence of the degree distribution and correlations between the vertex degrees on the power-law Laplacian spectra was studied. To this end, random networks were generated with the same degree distributions, degree-degree correlation functions, and degree-dependent clustering as the evolved networks. The resulting reconstructed networks, which are apart from the three input functions completely random, were compared with the evolved networks. The Laplacian spectra of the reconstructed networks fit the target power-law spectra surprisingly well. However, one main problem is that almost all reconstructed networks consist of multiple connected components. The resulting degeneracy of the zero eigenvalue has a strong influence on the spectrum. It causes a large deviation from the power-law scaling in the most relevant regime of small eigenvalues. In the ensemble of reconstructed networks, the effect is clearly visible by a segregation in the distribution of spectral distances into well-separated groups with the same number of connected components. This is a major issue since no method is available to control the number of components in the reconstructed networks. From a dynamical point of view, the comparison of the evolved and reconstructed networks becomes problematic since global connectivity is a crucial aspect in processes like diffusion or synchronization. As only 12 out of 10^4 realization of the reconstructed networks were found connected, a restriction of the analysis to those is not an option. Nevertheless, the reconstructed networks can be analyzed and it seems that they consist mainly of one large and several small components with the largest component exhibiting a core-periphery structure as well. Restricting the analysis to the largest components of the reconstructed networks would be a possibility to overcome the problem. This, however, introduces additional deviations from the prescribed degree distributions and correlation functions and gives no control of the resulting network sizes. In order to further pursue the investigation of the influence of the three functions on spectral properties, a continuation can be the reconstruction of the evolved networks based on less information using only one or two out of the three distributions of vertex degrees, two-point correlations, and degree-dependent clustering. This should give a more precise picture of their individual importance in the evolution.

In the third part, the extent to which the power-law Laplacian spectra can be realized by more homogeneous network structures was studied. Homogeneity in the vertex degrees was introduced as additional external constraint. Starting from lattices and corresponding realizations of regular random graphs, the networks were restricted to maintain the degree regularity in the course of the evolution. Also in this case the adaptive walk optimization succeeded to evolve the regular networks towards the target spectra. The eigenvalues of the evolved networks fit the power-law scaling especially in the small eigenvalue regime. Additionally, a high degeneracy of certain eigenvalues was observed. The degenerate eigenvalues were found to be associated with an abundance of symmetric motifs in the networks. The symmetry properties can be exploited to construct quotient networks extracting the underlying backbone structures while removing the (local) symmetries. This coarse-graining indeed purges the eigenvalue degeneracies but preserves the overall power-law scaling. Thus, the extracted backbones are identified as the major factors for the scaling in the Laplacian spectrum. They exhibit sequences of linear segments connected into loops and dangling ends of different lengths. For the lower spectral dimension, largely tree-like structures were found which can be considered as "almost one-dimensional". An interpretation of this observation is that in the evolving networks too many edges are present to form the lower-dimensional structures. The excess edges are effectively hidden in the symmetric motifs in order to form a network with the relevant scaling properties on larger length scales. This is consistent with the general understanding that the large scale structures are associated with the small eigenvalues determining the long-time dynamical behavior.

Concluding, different strategies were found that lead to the prescribed power-law scaling in the Laplacian spectrum. For the general case of evolution with global connectivity as sole constraint, inhomogeneous network structures were observed. This structural heterogeneity also seems to be imprinted in the degree correlation functions. In the case of evolution under the additional constraint of degree regularity, on the other hand, inhomogeneity in the spectra by high degeneracy of certain eigenvalues were found. The question whether this is purely accidental or if there exists a duality between inhomogeneous structure with homogeneous spectrum and homogeneous structure with inhomogeneous spectrum cannot be ultimately settled here. If, however, such a general symmetry relation could be found it would certainly be a very interesting question of further research.

Throughout this work, the evolution towards a prescribed Laplacian spectrum was motivated by the resulting anomalous diffusion behavior. The significance for network dynamics, however, goes much beyond diffusion and random walks. As presented above, also other important processes such as synchronization of oscillators belong to the class of Laplacian dynamics. Although in this case the Laplacian eigenratio is usually considered as the relevant quantity to characterize the synchronizability, the dynamics towards a synchronized state are also determined by the whole spectrum. Even more general, the method of network evolution developed in this thesis can, in principle, be applied to any dynamical process on networks as long as the target behavior is prescribed in terms of the eigenvalue spectrum of the (linearized) time evolution operator.

A. Detailed Description of Algorithms

A.1. Configuration Model with Clustering

The algorithm to construct random networks with given degree distribution, two-point correlations, and degree-dependent clustering used in chapter 4 builds upon the algorithm by Ángeles Serrano and Boguñá (2005). Hence, this algrithm is explained first in the following.

The algorithm is an extension of the configuration model including the degree-dependent clustering coefficient $C(k)$ and, indirectly, degree-degree correlations as predefined quantities. As described by Ángeles Serrano and Boguñá, the input consists of the two distributions $P(k)$ and $C(k)$. The generation of the random network is carried out as follows.

1. **Assignment of degrees and number of triangles.**
   a) An *a priori* degree sequence is chosen according to $P(k)$. Each vertex is assigned number of stubs from this given degree sequence.
   b) Each class of vertices with degree k is assigned an *a priori* number of triangle corners according to $C(k)$.
   c) All vertices begin with 0 associated edges and all degree classes with 0 associated triangles.

2. **Closure of triangles.** For this, the set of eligible components (EC) is defined as the set of all stubs and edges which are to form more triangles. Those stubs and edges which cannot form more triangles or would result in additional triangles of already satisfied classes are removed from EC. The subset of eligible components in a degree class k is denoted by EC(k).

   The following steps to form new triangles are repeated until EC is empty or all degree classes are satisfied.

   a) A degree class k_1 with unsatisfied number of triangles is chosen. The number is drawn, in general, from a distribution $\Pi(k)$. The choice of this distribution allows for the introduction of degree-degree correlations. If no assortativity is wanted, $\Pi(k)$ is chosen proportional to the number of triangles to be formed in class k.

      The first vertex v_1 of the triangle to be formed is selected by randomly choosing an element from EC(k_1) and a vertex connected to it. A second component of v_1 is drawn.

b) Two more vertices are selected to close the triangle.

i. If the two selected components of v_1 are both edges, the triangle is closed by joining free stubs of two vertices v_2 and v_3 at the other ends of the two edges.

ii. If the two components are one edge and one stub, the second vertex v_2 is chosen as the one at the other end of the edge. A second component of v_2 is drawn. If it is an edge, vertex v_3 at the other end of this edge closes the triangle. Otherwise, a third vertex with now at least two free stubs is selected as in a) to find v_3.

iii. If the two components are both stubs, the second vertex v_2 is chosen as in a) and a second component of v_2 is drawn. If one of the components of v_2 is an edge the vertex at the other end of is chosen as v_3. Otherwise, v_3 with now at least two free stubs is selected as in a).

c) The triangle (v_1, v_2, v_3) is formed by pairing the stubs to edges and removing all newly build edges and triangles from the respective lists. The set of eligible components is also updated.

3. **Closure of remaining free stubs.** The remaining stubs are matched as in the original configuration model.

A.2. Two-Point Correlated Random Networks with Clustering

The algorithm by Ángeles Serrano and Boguñá was further extended by Pusch *et al.* (2008) for the generation of clustered random networks with a given degree distribution and, additionally, a given two-point correlation function $P(j, k)$. Instead of working with the distributions, the algorithm, as described by Pusch *et al.*, requires discrete realizations as input. Namely,

- the degree distribution as discrete frequencies $\hat{P}(k)$, the number of vertices with degree k,
- the two-point correlation function as discrete frequency distribution $\hat{P}(j, k)$, the number of edges connecting vertices with degrees j and k,
- the degree-dependent clustering as discrete frequency distribution $\hat{T}(k)$ of triangle corners with degree k.

The network generation is done in the following steps.

1. Assign the degrees of $\hat{P}(k)$ to the vertices.
2. Create copies of $\hat{P}(j, k)$, $\hat{T}(k)$ and a list of triangle corner degrees with $\hat{T}(k)$ entries k to updated dynamically after the formation of any edges.
3. Form triangles by repeating the following steps until the triangle list is empty or none of the needed edges can be found.

a) Draw k_1 from the triangle list and randomly choose a vertex v_1 with degree k_1 which has free stubs.

b) Out of the eligible components of v_1 draw a random element. If this is an edge the vertex at the other end is v_2. Otherwise, draw the second vertex v_2 with degree k_2 as in a) which fulfills $\hat{P}(k_2, k_1) > 0$.

c) Repeat b) for v_2 to find vertex v_3 with degree k_3. Additionally, v_1 and v_3 have to be already connected or $\hat{P}(k_3, k_1) > 0$.

d) Form the triangle (v_1, v_2, v_3) by introducing all edges which are not present so far. Update all the dynamical quantities.

4. Create an edge list with all remaining $\hat{P}(k_1, k_2)$ entries (k_1, k_2).

5. Draw random elements (k_1, k_2) from the edge list and create edges between randomly chosen vertices with degrees k_1 and k_2, respectively.

   Repeat this step until the edge list is empty or no vertices can be found matching the needed degrees.

6. Randomly match all remaining stubs like in the original configuration model.

The algorithm generates random networks which approximately satisfy the degree distribution and correlations prescribed by $\hat{P}(k)$, $\hat{P}(j, k)$, and $\hat{T}(k)$. The deviations from $\hat{P}(k)$ and $\hat{P}(j, k)$ are usually very small. For small networks and a low level of clustering, it was observed that slightly more triangles are formed than expected. This is due to the possibility to form additional triangles in steps 5 and 6.

B. Numerical Libraries

The scientific work presented in this thesis is mainly based on numerical simulations. The computer programs for these simulations and for the data analysis were implemented from scratch making use of several numerical packages for standard as well as more specialized computational tasks. In the following, the numerical libraries that were used for the specific tasks are listed and, if available, references are given.

- The simulation programs for the network evolution algorithm are written in the `C++` programming language making extensive use of the Standard Template Library.
- For the representation of networks and their convenient handling concepts and algorithms from the `Boost Graph Library` (Siek *et al.*, 2002) are used.
- Random numbers needed for the stochastic algorithms are generated by the Mersenne Twister pseudorandom number generator (Matsumoto and Nishimura, 1998) implemented in the `Boost Random Library`.
- The calculation of eigenvalues and eigenvectors is done by the respective functions from the `GNU Scientific Library` (Galassi, 2009) and `LAPACK` (Anderson *et al.*, 1999).
- For the generation of random networks with given degree-degree correlations and clustering applied in chapter 4 the original implementation in `C++` by Andreas Pusch and Sebastian Weber (Pusch *et al.*, 2008) is used.
- The calculation of automorphism groups and orbit decompositions for the construction of quotient graphs in chapter 5 is done by the `nauty` package (McKay and Piperno, 2014).
- The analysis of the simulation data is mainly implemented in the `python` programming language. The `NumPy` (van der Walt *et al.*, 2011) and `graph-tool` (Peixoto, 2014) libraries are used extensively.
- For the figures presented, the plots were made with the `matplotlib` python library (Hunter, 2007) and network visualizations with the `graphviz` program (Gansner and North, 2000).

Bibliography

R. Albert and A.-L. Barabási (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics* **74**(1), 47.

K. T. Allhoff, D. Ritterskamp, B. C. Rall, B. Drossel, and C. Guill (2015). Evolutionary food web model based on body masses gives realistic networks with permanent species turnover. *Scientific Reports* **5**, 10955.

L. A. N. Amaral and J. M. Ottino (2004). Complex networks. *The European Physical Journal B - Condensed Matter and Complex Systems* **38**(2), 147.

E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen (1999). *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia.

P. W. Anderson (1972). More Is Different. *Science* **177**(4047), 393.

M. Ángeles Serrano and M. Boguñá (2005). Tuning clustering in random networks with arbitrary degree distributions. *Physical Review E* **72**(3), 036133.

A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou (2008). Synchronization in complex networks. *Physics Reports* **469**(3), 93.

P. Bak, C. Tang, and K. Wiesenfeld (1987). Self-organized criticality: An explanation of the $1/f$ noise. *Physical Review Letters* **59**(4), 381.

A. Banerjee and J. Jost (2008). Spectral plot properties: Towards a qualitative classification of networks. *Networks and Heterogeneous Media* **3**(2), 395.

A. Banerjee and J. Jost (2009). Spectral Characterization of Network Structures and Dynamics. In N. Ganguly, A. Deutsch, and A. Mukherjee (editors), *Dynamics On and Of Complex Networks*, Modeling and Simulation in Science, Engineering and Technology, pp. 117–132. Birkhäuser Boston.

A.-L. Barabási and R. Albert (1999). Emergence of Scaling in Random Networks. *Science* **286**(5439), 509.

A. Barrat, M. Barthélemy, and A. Vespignani (2008). *Dynamical Processes on Complex Networks*. Cambridge University Press, Cambridge.

B. Barzel and A.-L. Barabási (2013). Universality in network dynamics. *Nature Physics* **9**(10), 673.

D. ben-Avraham and S. Havlin (2000). *Diffusion and reactions in fractals and disordered systems*. Cambridge University Press, Cambridge.

J. Blitzstein and P. Diaconis (2011). A Sequential Importance Sampling Algorithm for Generating Random Graphs with Prescribed Degrees. *Internet Mathematics* **6**(4), 489.

S. Bornholdt (2001). Modeling Genetic Networks and Their Evolution: A Complex Dynamical Systems Perspective. *Biological Chemistry* **382**(9), 1289.

S. Bornholdt and T. Rohlf (2000). Topological Evolution of Dynamical Networks: Global Criticality from Local Dynamics. *Physical Review Letters* **84**(26), 6114.

S. Braunewell and S. Bornholdt (2008). Reliability of genetic networks is evolvable. *Physical Review E* **77**(6), 060902(R).

A. J. Bray and G. J. Rodgers (1988). Diffusion in a sparsely connected space: A model for glassy relaxation. *Physical Review B* **38**(16), 11461.

R. Burioni and D. Cassi (1996). Universal Properties of Spectral Dimension. *Physical Review Letters* **76**(7), 1091.

R. Burioni and D. Cassi (2005). Random walks on graphs: ideas, techniques and results. *Journal of Physics A: Mathematical and General* **38**(8), R45.

R. Burioni, S. Chibbaro, D. Vergni, and A. Vulpiani (2012). Reaction spreading on graphs. *Physical Review E* **86**(5), 055101.

D. S. Callaway, J. E. Hopcroft, J. M. Kleinberg, M. E. J. Newman, and S. H. Strogatz (2001). Are randomly grown graphs really random? *Physical Review E* **64**(4), 041902.

D. Chu, R. Strand, and R. Fjelland (2003). Theories of complexity. *Complexity* **8**(3), 19.

F. R. K. Chung (1997). *Spectral graph theory*. Number 92 in Regional conference series in mathematics. American Mathematical Society, Providence.

F. Comellas and J. Diaz-Lopez (2008). Spectral reconstruction of complex networks. *Physica A: Statistical Mechanics and its Applications* **387**(25), 6436.

D. Cvijović and J. Klinowski (1995). Taboo Search: An Approach to the Multiple Minima Problem. *Science* **267**(5198), 664.

D. J. de Solla Price (1965). Networks of Scientific Papers. *Science* **149**(3683), 510.

D. J. de Solla Price (1976). A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science* **27**(5), 292.

C. I. Del Genio, H. Kim, Z. Toroczkai, and K. E. Bassler (2010). Efficient and Exact Sampling of Simple Graphs with Given Arbitrary Degree Sequence. *PLoS ONE* **5**(4), e10012.

L. Donetti, P. I. Hurtado, and M. A. Muñoz (2005). Entangled Networks, Synchronization, and Optimal Network Topology. *Physical Review Letters* **95**(18), 188701.

L. Donetti, F. Neri, and M. A. Muñoz (2006). Optimal network topologies: expanders, cages, Ramanujan graphs, entangled networks and all that. *Journal of Statistical Mechanics: Theory and Experiment* **2006**(08), P08007.

L. Donetti, P. I. Hurtado, and M. A. Muñoz (2008). Network synchronization: optimal and pessimal scale-free topologies. *Journal of Physics A: Mathematical and Theoretical* **41**(22), 224008.

S. N. Dorogovtsev and J. F. F. Mendes (2002). Evolution of networks. *Advances in Physics* **51**(4), 1079.

B. Drossel (2008). Random Boolean Networks. In H. G. Schuster (editor), *Reviews of Nonlinear Dynamics and Complexity*, pp. 69–110. Wiley-VCH.

P. Erdős and A. Rényi (1959). On random graphs. *Publicationes Mathematicae (Debrecen)* **6**, 290.

P. Erdős and A. Rényi (1960). On the evolution of random graphs. *Magyar Tud. Akad. Mat. Kutató Int. Közl.* **5**, 17.

P. Erdős and A. Rényi (1961). On the strength of connectedness of a random graph. *Acta Mathematica Academiae Scientiarum Hungarica* **12**(1-2), 261.

M. Galassi (editor) (2009). *GNU scientific library: reference manual*. A GNU manual. Network Theory, s.l.

E. R. Gansner and S. C. North (2000). An open graph visualization system and its applications to software engineering. *Software: Practice and Experience* **30**(11), 1203.

C. D. Godsil and G. Royle (2004). *Algebraic graph theory*. Number 207 in Graduate texts in mathematics. Springer, New York.

N. Goldenfeld and L. P. Kadanoff (1999). Simple Lessons from Complexity. *Science* **284**(5411), 87.

T. E. Gorochowski, M. di Bernardo, and C. S. Grierson (2010). Evolving enhanced topologies for the synchronization of dynamical complex networks. *Physical Review E* **81**(5), 056212.

S. F. Greenbury, I. G. Johnston, M. A. Smith, J. P. Doye, and A. A. Louis (2010). The effect of scale-free topology on the robustness and evolvability of genetic regulatory networks. *Journal of Theoretical Biology* **267**(1), 48.

T. Gross and B. Blasius (2008). Adaptive coevolutionary networks: a review. *Journal of The Royal Society Interface* **5**(20), 259.

A. Gurtovenko and A. Blumen (2005). Generalized Gaussian Structures: Models for Polymer Systems with Complex Topologies. In *Polymer Analysis, Polymer Theory*, number 182 in Advances in Polymer Science, pp. 171–282. Springer, Berlin Heidelberg.

U. H. E. Hansmann and L. T. Wille (2002). Global Optimization by Energy Landscape Paving. *Physical Review Letters* **88**(6), 068105.

S. Havlin and D. ben-Avraham (2002). Diffusion in disordered media. *Advances in Physics* **51**(1), 187.

J. D. Hunter (2007). Matplotlib: A 2d graphics environment. *Computing In Science & Engineering* **9**(3), 90.

M. Ipsen and A. S. Mikhailov (2002). Evolutionary reconstruction of networks. *Physical Review E* **66**(4), 046109.

G. Jurman, R. Visintainer, and C. Furlanello (2011). An introduction to spectral distances in networks. In B. Apolloni, S. Bassis, A. Esposito, and C. F. Morabito (editors), *Neural Nets Wirn10: Proceedings of the 20th Italian Workshop on Neural Nets*, number 226 in Frontiers in Artificial Intelligence and Applications, pp. 227–234. IOS Press.

S. Karalus and J. Krug (2015). Symmetry-based coarse-graining of evolved dynamical networks. *EPL (Europhysics Letters)* **111**(3), 38003.

S. Karalus and M. Porto (2012). Network evolution towards optimal dynamical performance. *EPL (Europhysics Letters)* **99**(3), 38002.

N. Kashtan and U. Alon (2005). Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences of the United States of America* **102**(39), 13773 .

S. A. Kauffman (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology* **22**(3), 437.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi (1983). Optimization by Simulated Annealing. *Science* **220**(4598), 671.

B. D. MacArthur and R. J. Sánchez-García (2009). Spectral characteristics of network redundancy. *Physical Review E* **80**(2), 026117.

B. D. MacArthur, R. J. Sánchez-García, and J. W. Anderson (2008). Symmetry in complex networks. *Discrete Applied Mathematics* **156**(18), 3525.

M. Matsumoto and T. Nishimura (1998). Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation* **8**(1), 3.

B. D. McKay and A. Piperno (2014). Practical graph isomorphism, II. *Journal of Symbolic Computation* **60**, 94.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953). Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics* **21**(6), 1087.

S. Milgram (1967). The Small-World Problem. *Psychology Today* **1**(1), 61.

R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon (2002). Network Motifs: Simple Building Blocks of Complex Networks. *Science* **298**(5594), 824.

B. Mohar (1997). Some applications of Laplace eigenvalues of graphs. In G. Hahn and G. Sabidussi (editors), *Graph Symmetry: Algebraic Methods and Applications*, number 497 in NATO ASI Series, pp. 225–275. Springer Netherlands.

M. Molloy and B. Reed (1995). A critical point for random graphs with a given degree sequence. *Random Structures & Algorithms* **6**(2-3), 161.

J. L. Moreno (1934). *Who shall survive?: A new approach to the problem of human interrelations*. Number 58 in Nervous and Mental Disease Monograph Series. Nervous and Mental Disease Publ. Co., Washington.

M. E. J. Newman (2002). Assortative Mixing in Networks. *Physical Review Letters* **89**(20), 208701.

M. E. J. Newman (2003a). Mixing patterns in networks. *Physical Review E* **67**(2), 026126.

M. E. J. Newman (2003b). The Structure and Function of Complex Networks. *SIAM Review* **45**(2), 167.

M. E. J. Newman (2010). *Networks: An Introduction*. Oxford University Press, New York.

M. E. J. Newman and G. T. Barkema (1999). *Monte Carlo methods in statistical physics*. Oxford University Press, Oxford.

P. Oikonomou and P. Cluzel (2006). Effects of topology on network evolution. *Nature Physics* **2**(8), 532.

H. A. Orr (2002). The Population Genetics of Adaptation: The Adaptation of DNA Sequences. *Evolution* **56**(7), 1317.

T. P. Peixoto (2014). The graph-tool python library. *figshare* URL `http://figshare.com/articles/graph_tool/1164194`.

A. Pusch, S. Weber, and M. Porto (2008). Generating random networks with given degree-degree correlations and degree-dependent clustering. *Physical Review E* **77**(1), 017101.

A. A. Rad, M. Jalili, and M. Hasler (2008). Efficient rewirings for enhancing synchronizability of dynamical networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **18**(3), 037104.

A. N. Samukhin, S. N. Dorogovtsev, and J. F. F. Mendes (2008). Laplacian spectra of, and random walks on, complex networks: Are scale-free architectures really important? *Physical Review E* **77**(3), 036115.

Z. Shao and H. Zhou (2009). Dynamics-driven evolution to structural heterogeneity in complex networks. *Physica A: Statistical Mechanics and its Applications* **388**(4), 523.

J. Siek, L.-Q. Lee, and A. Lumsdaine (2002). *The boost graph library: user guide and reference manual*. C++ in-depth series. Addison-Wesley, Boston.

R. Solomonoff and A. Rapoport (1951). Connectivity of random nets. *The bulletin of mathematical biophysics* **13**(2), 107.

A. Szejka and B. Drossel (2010). Evolution of Boolean networks under selection for a robust response to external inputs yields an extensive neutral space. *Physical Review E* **81**(2), 021908.

S. van der Walt, S. C. Colbert, and G. Varoquaux (2011). The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering* **13**(2), 22.

A. Vázquez, R. Pastor-Satorras, and A. Vespignani (2002). Large-scale topological and dynamical properties of the Internet. *Physical Review E* **65**(6), 066130.

D. J. Watts and S. H. Strogatz (1998). Collective dynamics of 'small-world' networks. *Nature* **393**(6684), 440.

S. Weber and M. Porto (2007). Generation of arbitrarily two-point-correlated random networks. *Physical Review E* **76**(4), 046111.

W. W. Zachary (1977). An Information Flow Model for Conflict and Fission in Small Groups. *Journal of Anthropological Research* **33**(4), 452.

Erklärung der Selbstständigkeit

Ich versichere, dass ich die von mir vorgelegte Dissertation selbständig angefertigt, die benutzten Quellen und Hilfsmittel vollständig angegeben und die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken im Wortlaut oder dem Sinn nach entnommen sind, in jedem Einzelfall als Entlehnung kenntlich gemacht habe; dass diese Dissertation noch keiner anderen Fakultät oder Universität zur Prüfung vorgelegen hat; dass sie – abgesehen von unten angegebenen Teilpublikationen – noch nicht veröffentlicht worden ist, sowie, dass ich eine solche Veröffentlichung vor Abschluss des Promotionsverfahrens nicht vornehmen werde. Die Bestimmungen der Promotionsordnung sind mir bekannt. Die von mir vorgelegte Dissertation ist von Prof. Dr. Joachim Krug betreut worden.

Köln, 10. November 2015,

Steffen Karalus

Teilpublikationen

- S. Karalus and M. Porto. Network evolution towards optimal dynamical performance. *EPL (Europhysics Letters)* **99**(3), 38002 (2012).
- S. Karalus and J. Krug. Symmetry-based coarse-graining of evolved dynamical networks. *EPL (Europhysics Letters)* **111**(3), 38003 (2015).