Network Structure Optimization with Applications to Minimizing Variance and Crosstalk

Thesis by

Dionysios Barmpoutis

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

2012

(Defended December 2, 2011)

To my mother

Acknowledgments

As I near the end of my time at Caltech, I can't help but reflect back on all the people who have helped me along the way. I have had the pleasure of working with many amazing individuals, both students and faculty. First and foremost, I would like to thank my advisor, Richard M. Murray. His energy and passion have been a source of inspiration throughout my graduate career at Caltech. With his guidance and mentorship, I have grown as a student, as a researcher, and as a person. All his insightful comments were greatly appreciated, and his ability to guide me while still allowing me to have control of the direction of my research projects created great opportunities along the way. I would also like to thank Yaser Abu-Mostafa, Joel Burdick, John Doyle, and Pietro Perona, for their advice and for generously taking the time to be part of my thesis committee.

For our many meaningful discussions, I would like to thank my officemate, Chris Kempes. I would also like to thank Sotiris Masmanidis, one of the first people I worked with at Caltech, for his leadership and advice. There are many other individuals who have made my time at Caltech fulfilling both academically and personally, and for that, I would like to extend a big thanks to Elisa Franco, Theodoros Dikaliotis, Vanessa Jönsson, Ophelia Venturelli, Andy Lamperski, Håkan Terelius, Chess Stetson, Sawyer Fuller, Dan Wilhelm, Alice Robinson, Shaunak Sen, Molei Tao, Paul Skerritt, Costas Anastassiou, Joe Meyerowitz, Xiaodi Hou, Pete Trautman, Vasileios Christopoulos, Shuo Han, Andrea Censi, Aristo Asimakopoulos, Jongmin Kim, Stephen Prajna, Mumu Xu, Jun Liu, Demetri Spanos, Chris Santis, Eyal En Gad, Katie Galloway, and Morr Mehyar. The work contained in this thesis would

not have been possible without the academic guidance I received from Erik Winfree, Shuki Bruck, Paul Rothemund, and Christof Koch. It has truly been an honor to have known and worked with all of them during my graduate studies.

I was also very fortunate to have entered Caltech with such great people in the CNS class in the Fall of 2007: Julien Dubois, Virgil Griffith, Alice Lin, Akram Sadek, Marie Suver, and Peter Welinder. It was a great pleasure spending time together working through our problem sets during our first year of classes. In addition to the support I received from fellow students and faculty, I must thank Anissa Scott and Gloria Bain for making my life so much easier with booking flights, preparing for conferences, and assisting with other administrative issues.

Of course, I am forever grateful to my parents and brother for their unconditional support, even while being thousands of miles away. Last but not least, I would like to thank Alison Rose for her unconditional love and constant support through the ups and downs of graduate life. Without her, I would not be where I am today.

Abstract

This thesis provides a unified methodology for analyzing structural properties of graphs, along with their applications. In the last several years, the field of complex networks has been extensively studied, and it is now well understood that the way a large network is built is closely intertwined with its function. Structural properties have an impact on the function of the network, and the form of many systems has been evolved in order to optimize for given functions. Despite the great progress, particularly in how structural attributes affect the various network functions, there is a significant gap in the quantitative study of how much these properties can change in a network without a significant impact on the functionality of the system, or what the bounds of these structural attributes are. Here, we find and analytically prove tight bounds of global graph properties, as well as the form of the graphs that achieve these bounds. The attributes studied include the network efficiency, radius, diameter, average distance, betweenness centrality, resistance distance, and average clustering. All of these qualities have a direct impact on the function of the network, and finding the graph that optimizes one or more of them is of interest when designing a large system. In addition, we measure how sensitive these properties are with respect to random rewirings or addition of new edges, since designing a network with a given set of constraints may include a lot of trade-offs. This thesis also studies properties that are of interest in both natural and engineered networks, such as maximum immunity to crosstalk interactions and random noise. We are primarily focused on networks where information is transmitted through a means that is accessible by all the individual units of the network and the interactions among the different entities that

comprise it do not necessarily have a dedicated mechanism that facilitates information transmission, or isolates them from other parts of the network. Two examples of this class are biological and chemical reaction networks. Such networks suffer from unwanted crosstalk interactions when two or more units spuriously interact with each other. In addition, they are subject to random fluctuations in their output, both due to noisy inputs and because of the random variance of their parameters. These two types of randomness affect the behavior of the system in ways that are intrinsically different. We examine the network topologies that accentuate or alleviate the effect of random variance in the network for both directed and undirected graphs, and find that increasing the crosstalk among different parts reduces the output variance but also contributes to a slower response.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The area of complex networks has seen an explosion in the last several years. Many properties of large scale networks have been studied extensively, along with their applications to engineering and biology [1, 8, 34, 37, 40, 45]. However, there have been relatively few studies so far on the bounds of the structural properties of networks.

The purpose of discovering these bounds is twofold. First, these bounds will give a clear measure of the importance of each property, especially relative to the others, as well as their trade-offs. If for example a natural network has a clustering coefficient that is very close to the theoretical maximum, it means that the clustering coefficient provides some advantage, or that it is correlated with some other property that is important for the function of the network. Otherwise, natural evolution would force it to drift to an average value. In addition, since several of the properties of the network require different or even contradicting topologies, usually there are limits on how much of each property a network may have. Knowing where these trade-offs lie can give us a clear picture of how important each property is relative to the others.

The second reason is for optimizing the way individual elements of a network are connected and how they communicate, especially on a large scale. Knowing the mechanisms with which specific properties affect the function of a networked system,

it will be easier to optimize its function or design it in such a way that all constraints are simultaneously satisfied with the given trade-offs. In biological and chemical reaction systems, there is yet another compelling reason: Since the structure of a natural system has been shaped by evolution, we have the chance to see the interplay between structure and evolvability, and how they affect each other [19, 35, 50], or even find evidence of different environmental conditions that may have shaped it [36].

On a different note, bioengineers have long started building simple biological networks, both as part of the cell [24] and in vitro [25, 38], but despite the many efforts, they remain relatively small so far [57, 58], especially compared with the natural networks, which are orders of magnitude larger and more complex [71]. Two likely reasons why engineering biological circuits is so difficult are the unwanted physical interactions among unrelated molecules, and the fact that noise is often times much stronger than the signal itself.

Biological networks make use of a variety of strategies in order to reduce noise, and to increase the robustness to random changes, but these mechanisms may be used for many different purposes, and are not yet well understood [39, 71]. The ones that sense the environment and process the information received are usually implemented by changing the number of molecules that are specific to each function. More generally, in networks where the different parts are free to physically move in a solution, and where interactions between different molecules require them to come in contact, there are many unmodeled interactions, since there are no dedicated information-transmitting mechanisms. Everything depends on physical presence of molecules, which is by definition random and displays significant internal noise [55]. Relations between different entities may also depend on their physical shape, and the different molecules that constitute the network may have compatible shapes, making interactions easier. As a result, many molecules temporarily form complexes that seem to serve no purpose, and in addition, prevent the respective molecules from transmitting information, or helping the cell achieve a useful function.

Another problem is that these kinds of systems are very prone to noise from many different sources [54, 56]. Although noise has been shown to be a feature and not a drawback in some cases [23], it is usually detrimental to the function of the network, since it may reduce its reliability and accuracy in reading external stimuli, or reacting to environmental inputs. Both of these problems are accentuated when there are few molecules of each type and there are many types of each molecule, as actually happens in most biological systems, and the number of molecules of each type cannot be a continuous variable, making accuracy and reliability even harder to achieve [54]. In certain contexts, it has also been shown that noise imposes limits to the accuracy of a biological network [42].

In this thesis, we tackle both aforementioned problems from a network perspective. We first find the extremal properties that are of interest in the function of many networks, and then we study the structure that optimizes them. Then, we design networks that minimize crosstalk and maximize noise immunity. We also study in detail how noise propagates in such networks. Finally, we distinguish between two types of variance sources that contribute to a nondeterministic output, the first being the noise in the inputs, and the second being variance in the network parameters. These two types of variance affect the network outputs in fundamentally different ways.

1.2 Thesis Outline and Contributions

Chapter 2 introduces some basic notions and definitions from network theory, stochastic calculus, and dynamical systems theory that will be used throughout this thesis. It also revisits some properties of graphs, Wiener processes, and linear dynamical systems that will be used in later chapters.

Chapter 3 focuses on theoretical results regarding the structural characteristics of general graphs. It provides a detailed study of the extremal properties of networks,

along with methods on how to build the networks that achieve these bounds. We describe the structure of connected graphs with the minimum and maximum average distance, radius, diameter, betweenness centrality, efficiency and resistance distance, and average clustering, given their order and size. We find tight bounds on these graph qualities for any arbitrary number of nodes and edges and analytically derive the form and properties of such networks. We determine if a graph with one or more of these extremal properties is unique or not, depending on the property and the graph's order and size. We also measure the sensitivity to rewiring of each architecture, and how robust each structure is with regard to changes in the graph.

Chapters 4 and 5 are devoted to the study of networks where information is transmitted through a means that is accessible by all the individual units of the network. Such networks include biological and chemical reaction networks, where all reactions take place in a solution in which all molecules may physically interact with all others, based on their physical proximity. Crosstalk is defined as the set of unwanted interactions among the different constituents of the network and is present in various degrees in every such system. Using concepts from graph theory, we introduce a quantifiable measure for sensitivity to crosstalk, and analytically derive the structure of the networks in which it is minimized. It is shown that networks with an inhomogeneous degree distribution are more robust to crosstalk than corresponding homogeneous networks. We provide a method to construct the graph with the minimum possible sensitivity to crosstalk, given its order and size. For networks with a fixed degree sequence, we present an algorithm to find the optimal interconnection structure among their vertices.

In Chapter 5, we describe how noise propagates through a network. Using stochastic calculus and dynamical systems theory, we study the network topologies that accentuate or alleviate the effect of random variance in the network for both directed and undirected graphs. Given a linear tree network, we show that the variance in the output is a convex function of the poles of the individual nodes, so it can eas-

ily be optimized with existing techniques [17]. Cycles create correlations which in turn increase the variance in the output. Feedforward and feedback have a limited effect on noise propagation when the respective cycle is sufficiently long. Crosstalk between the elements of different pathways helps reduce the output noise, but makes the network slower. Next, we study the differences between disturbances in the inputs and disturbances in the network parameters, and how they propagate to the outputs. Finally, we show how noise correlation can affect the steady state of the system in chemical reaction networks with reactions of two or more reactants, each of which may be affected by independent or correlated noise sources.

Chapter 6 concludes the analysis by giving an overview of the results presented here, and by showing future directions and ideas for further research.

Chapter 2

Background and Preliminaries

This chapter provides a brief introduction to the mathematical preliminaries that will be used throughout this thesis. We will revisit some basic definitions and properties from graph theory, linear control systems, and stochastic calculus.

2.1 Graph Theory

A *graph* (also called a *network*) is an ordered pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ comprised of a set $\mathcal{V} = \mathcal{V}(\mathcal{G})$ of *vertices* together with a set $\mathcal{E} = \mathcal{E}(\mathcal{G})$ of *edges* that are unordered 2-element subsets of $\mathcal{V}$. Two vertices u and v are called *neighbors* if they are connected through an edge ($(u, v) \in \mathcal{E}$) and in this case we write $u - v$, otherwise we write $u \neq v$. A graph is *simple* when all edges connect two different vertices, there is at most one edge connecting any pair of vertices, and edges have no direction. A *weighted graph* associates a weight with every edge. In this thesis, when a graph is weighted, all weights will be restricted to positive real numbers. The *neighborhood* $\mathcal{N}_u$ of a vertex u is the set of its neighbors. The *degree* of a vertex is the number of its neighbors. A vertex is said to have *full degree* if it is connected to every other vertex in the network.

A network is *assortative* with respect to its degree distribution when the vertices with large degrees are connected to others that have large degrees and vertices with small degrees are connected to vertices with small degrees [47]. When vertices with

small degrees connect to vertices with large degrees and vice versa, then the network is called *disassortative*. The *order* $N = N(\mathcal{G})$ of a graph $\mathcal{G}$ is the number of its vertices, $N = |\mathcal{V}(\mathcal{G})|$. A graph's *size* (denoted by $m = |\mathcal{E}(\mathcal{G})|$), is the number of its edges. We will denote a graph $\mathcal{G}$ of order N and size m as $\mathcal{G}(N, m)$ or simply $\mathcal{G}_{N,m}$. A *complete graph* is a graph in which each vertex is connected to every other. The *edge density* of a graph is defined as $\rho = m/\binom{N}{2}$, representing the number of present edges, as a fraction of the size of a complete graph, which is the total number of vertex pairs. A *clique* in a graph is a subset of its vertices in which every vertex pair in the subset is connected. The *clique order* is the number of vertices that belong to it. A clique that consists of three vertices (and three edges among them) is called a *triangle*. A *path* is a sequence of consecutive edges in a graph and the length of the path is the number of edges traversed. A path with no repeated vertices is called a *simple path*. A *tree* is a graph in which any two vertices are connected by exactly one path.

The *distance* between two vertices u and v (usually denoted by $d = d(u, v)$), is the length of the shortest path that connects these two vertices. A *cycle* is a closed (simple) path, with no other repeated vertices or edges other than the starting and ending nodes. A cycle is called *chordless* when there is no edge joining two nodes that are not adjacent in the cycle. A *full cycle* is a cycle that includes all the vertices of the network. A graph $\mathcal{G}$ is *connected* if for every pair of vertices $u \in \mathcal{V}(\mathcal{G})$ and $v \in \mathcal{V}(\mathcal{G})$, there is a path from u to v. Otherwise the graph is called *disconnected*. We will be focusing exclusively on connected graphs, since every disconnected graph can be analyzed as the sum of its connected *components*. If the distance between u and v is equal to k, then these vertices are called *k-neighbors*, and the set of all pairs in the graph that are $k-$neighbors is denoted by $\mathcal{E}_k$. The *eccentricity* of a vertex u is the maximum distance of u from any other vertex in the graph. A *central vertex* of a graph is a vertex that has eccentricity smaller or equal to any other node. A network may have many central vertices, all of which are considered its centers. The

eccentricity of a central vertex is called the graph *radius*. The graph *diameter* is defined as the maximum of the distances among all vertex pairs in the network.

A *cut* is a partition of the vertices of a graph into two disjoint subsets. A *cut set* of the cut is the set of edges whose end points are in different subsets of the cut. A *cut vertex* of a connected graph is a vertex that if removed, (along with all edges incident to it) produces a graph that is disconnected. An edge is *rewired* when we change the vertices it is adjacent to. A *single rewiring* takes place when we change one of the vertices that is adjacent to it, and a *double rewiring* occurs when we change both of them. A subgraph $\mathcal{H}$ of a graph $\mathcal{G}$ is called *induced* if $\mathcal{V}(\mathcal{H}) \subseteq \mathcal{V}(\mathcal{G})$ and for any pair of vertices u and v in $\mathcal{V}(\mathcal{H})$, $(u, v) \in \mathcal{E}(\mathcal{H})$ if and only if $(u, v) \in \mathcal{E}(\mathcal{G})$. In other words, $\mathcal{H}$ is an induced subgraph of $\mathcal{G}$ if it has the same edges that appear in $\mathcal{G}$ over the same vertex set. Furthermore, if the vertex set of $\mathcal{H}$ is the subset $\mathcal{S}$ of $\mathcal{V}(\mathcal{G})$, then $\mathcal{H}$ can be written as $\mathcal{G}[\mathcal{S}]$ and is said to be induced by $\mathcal{S}$. Finally, two graphs $\mathcal{G}$ and $\mathcal{H}$ are called isomorphic if there exists a bijective function $f : \mathcal{V}(\mathcal{G}) \to \mathcal{V}(\mathcal{H})$ such that

$$(u, v) \in \mathcal{E}(\mathcal{G}) \iff (f(u), f(v)) \in \mathcal{E}(\mathcal{H}). \tag{2.1}$$

Two graphs that are isomorphic have by definition the same order and size, and are considered identical. A thorough treatment of the graph theory notions can be found in any introductory graph theory text, including [31] and [48].

2.2 Wiener Processes

In this section, we will be describing some elementary properties of the Wiener process that will be used. Let ξ_n, $n \in \mathbb{N}$ be a sequence of independent identically distributed random variables with zero mean and unit standard deviation. Their sum is

$$S_n = \sum_{k=1}^{n} \xi_n. \tag{2.2}$$

We now define the piecewise constant function

$$W_t = \lim_{n\to\infty} \frac{S_{\lfloor nt \rfloor}}{\sqrt{n}} \tag{2.3}$$

with $t \in \mathbb{R}^+$. According to the Central Limit Theorem, the distribution of W_t is *independent* of the distribution of the sequence of ξ_n, as long as they have finite variance, and are identically distributed and independent of each other. The random process W_t is normally distributed with variance equal to the time interval it which it is measured:

$$W_t = \lim_{n\to\infty} \frac{S_{\lfloor nt \rfloor}}{\sqrt{nt}} \frac{\sqrt{nt}}{\sqrt{n}} \implies W_t \sim \mathcal{N}(0,t). \tag{2.4}$$

The difference of two sums $S_b - S_a$ with $a < b$ has the same distribution of the random variable S_{b-a} and as a result

$$W_b - W_a \sim W_{b-a} \qquad 0 \le a < b. \tag{2.5}$$

Lastly, the random variables $W_b - W_a$ and $W_d - W_c$ are independent when $0 \le a < b \le c < d$, since the respective sums consist of independent random variables. More details on the properties of the Wiener process can be found in [44]. An excellent treatment of stochastic methods in physics, chemistry, and biology, along with examples, can be found in [26].

2.3 General Response of Linear Systems

In this section, we briefly revisit some basic tools from control systems theory. Consider a linear time invariant system with impulse response $h(t,s)$. The general form of the output when the input signal is $u(t)$ is

$$y(t) = \int_{-\infty}^{t} h(t,s)u(s)ds, \tag{2.6}$$

where $h(t,s)$ is the impulse response of the dynamical system [6]. A system with m inputs, n states, and p outputs can be written in the form

$$\mathcal{S}: \begin{cases} \frac{dx}{dt} = Ax + Bu \\ \\ y = Cx, \end{cases} \tag{2.7}$$

where the dimensions of matrices A, B, and C are $n \times n$, $n \times m$, and $p \times n$, respectively. We will always assume that the systems we study are stable, which in this context means that the A matrix has eigenvalues with strictly negative real parts. The output of the system at time t when the input is an impulse applied at time s is

$$h(t,s) = Ce^{A(t-s)}B \tag{2.8}$$

and equation (2.6) can be simplified to

$$y(t) = C\int_{-\infty}^{t} e^{A(t-s)}Bu(s)ds. \tag{2.9}$$

When the network in question is comprised of elements whose outputs obey linear time-invariant differential equations, we can also find the Fourier transform of the network output:

$$H(\omega) = \int_{-\infty}^{+\infty} h(t)e^{-j\omega t}dt, \tag{2.10}$$

where $h(t) = h(t,0)$ is the impulse response of the system and $\omega = 2\pi f$ is the angular frequency. If the system is causal ($h(t) = 0$ for $t < 0$), then the expression above can be simplified by replacing the lower limit of the integral with zero.

When the input is a stationary stochastic process, its output will be a stochastic process as well. We are interested in the mean, the variance, and occasionally the higher central moments of the system output once the system has reached its equi-

librium state. The mean $\mathbb{E}[y(t)]$ and the variance $\mathbb{V}[y(t)]$ of the output $y(t)$ in the steady state will be denoted as $\mathbb{E}[y]$ and $\mathbb{V}[y]$, respectively:

$$\mathbb{E}[y] = \lim_{t\to\infty} \mathbb{E}[y(t)] \qquad \text{and} \qquad \mathbb{V}[y] = \lim_{t\to\infty} \mathbb{V}[y(t)]. \tag{2.11}$$

If we know the impulse response of the system, the mean of the output vector can be expressed as

$$\begin{aligned} \mathbb{E}[y(t)] &= \mathbb{E}\left[\int_{-\infty}^{t} h(t-s)u(s)ds\right] \\ &= \int_{-\infty}^{t} h(t-s)\cdot \mathbb{E}\left[u(s)\right] ds, \end{aligned} \tag{2.12}$$

where in the last equation we have interchanged the expectation with the integration operator, assuming that the input functions are non-pathological, and the quantities are finite, such that all the integrands are measurable in the respective measure space (Fubini's theorem, [44]). In what follows, we will always assume that all such conditions are satisfied. Furthermore, we will disregard any nonzero mean values in the outputs when the system is linear. The covariance matrix of the outputs, when applying the same input, is

$$\begin{aligned} \mathbb{V}[y(t)] &= \mathbb{E}[y(t)\cdot y^T(t)] - \mathbb{E}[y(t)]\cdot\mathbb{E}[y^T(t)] \\ &= \int_{-\infty}^{t}\int_{-\infty}^{t} h(t-r)\left(\mathbb{E}\left[u(r)u^T(s)\right] - \mathbb{E}\left[u(r)\right]\mathbb{E}\left[u^T(s)\right]\right) h^T(t-s)drds. \end{aligned} \tag{2.13}$$

Assuming that $u(t) = 0$ for $t < 0$, and according to equation (2.11),

$$\mathbb{V}[y] = \lim_{t\to\infty} \int_{0}^{t}\int_{0}^{t} h(t-r)\left(\mathbb{E}\left[u(r)u^T(s)\right] - \mathbb{E}\left[u(r)\right]\mathbb{E}\left[u^T(s)\right]\right) h^T(t-s)drds. \tag{2.14}$$

Chapter 3

Extremal Properties of Complex Networks

3.1 Introduction

Complex networks, as abstract models of large dynamical systems, match the structure of real-world networks in many diverse areas. These include both natural and engineered systems such as gene regulation, protein interaction networks, food webs, economic and social networks and the internet, to name a few (see [65] and references therein). Complex systems can be described as interconnections of simpler elements, which in turn can be analyzed abstractly as graphs.

In this chapter, we are interested in the structural properties of networks, regardless of the nature of their individual parts. This allows the results developed here to be applicable in a wide range of different disciplines, such as neuroscience, biology, social sciences, and engineering. The properties studied are of general interest, since many network functions are sensitive to them: the average distance, betweenness centrality, radius, diameter, efficiency, graph resistance, and average clustering. Depending on the application, we usually want to minimize or maximize one or more of the above, because they are directly implicated in some performance metric of the network. They are correlated with how fast the system responds to different input stimuli [65] and how robust it is to the failure of individual subsystems, due to ran-

dom failures or targeted attacks [1, 18, 28]. They also indicate how efficient message propagation is across a network [7, 41], how easy it is for dynamical processes that require global coordination and information flow (like synchronization or computation) to take place, and how reliable a transmitted message is in the presence of noise [65]. Although these structural properties do not take into account the specifics of the various systems, focusing on the structural patterns of the network topology can give a valuable insight on how to optimize the network function, while obeying other limitations. In general, networks need to obey many different constraints, and taking into account all of them may result in different optimal structures, depending on the importance (weight) given to each constraint [14].

3.2 Networks with the Minimum and Maximum Average Distance

3.2.1 Minimum Average Distance

The average distance of a network is an important property, since it is a direct indicator of how different parts of the network communicate, and exchange information. A small average distance is a proxy for improved synchronizability, efficient computation and signal propagation across the network [65]. In this section, we will analytically compute the minimum average distance of a graph of fixed order and size, and find sufficient conditions in order to achieve that minimum.

Lemma 1. *If two connected graphs G and H with $\mathcal{V}(\mathcal{G}) = \mathcal{V}(\mathcal{H})$ have edge sets $\mathcal{E}(\mathcal{G})$ and $\mathcal{E}(\mathcal{H})$, respectively, such that $\mathcal{E}(\mathcal{G}) \subseteq \mathcal{E}(\mathcal{H})$, then $\bar{D}(\mathcal{G}) \geq \bar{D}(\mathcal{H})$, where $\bar{D}$ denotes the average distance of the graph.*

Proof. If we start with graph $\mathcal{G} = \mathcal{G}(N, m)$ with average distance $\bar{D}(\mathcal{G})$, and introduce one additional edge, the new graph $\mathcal{G}' = \mathcal{G}'(N, m+1)$ will have an average distance $\bar{D}(\mathcal{G}') < \bar{D}(\mathcal{G})$, for $N - 1 \leq m \leq \binom{N}{2} - 1$. The additional edge will connect two

previously non-neighboring vertices s and t, changing their distance to $d'(s,t) = 1$. Since they were not connected before, their distance was $d(s,t) \geq 2$, so $d'(s,t) < d(s,t)$. For every other pair of vertices u and v, the new edge can only create new shortest paths, so $d'(u,v) \leq d(u,v)$. The total average shortest path length of the new graph is:

$$\bar{D}(\mathcal{G}') = \frac{1}{\binom{N}{2}} \sum_{\substack{(u,v)\in\mathcal{V}^2(\mathcal{G}') \\ u\neq v}} d'(u,v) < \frac{1}{\binom{N}{2}} \sum_{\substack{(u,v)\in\mathcal{V}^2(\mathcal{G}) \\ u\neq v}} d(u,v) = \bar{D}(\mathcal{G}). \tag{3.1}$$

Adding new edges as above, we can start from graph $\mathcal{G}$, and successively construct the graph $\mathcal{H}$, which will have a strictly smaller average distance. □

Lemma 2. *The star graph is the only tree of order N that has the smallest average distance equal to $\bar{D}_{star} = 2 - \frac{2}{N}$.*

Proof. A tree has exactly $N-1$ edges among its N vertices. There will be exactly $N-1$ pairs of vertices with distance $d = 1$, and $\binom{N-1}{2}$ vertex pairs that are not connected, with distances $d(u,v) \geq 2$. The star graph achieves this lower bound, and has the minimum possible average distance.

$$\bar{D}_{star} = \frac{1}{\binom{N}{2}} \sum_{\substack{(u,v)\in\mathcal{V}^2 \\ u\neq v}} d'(u,v) = \frac{1}{\binom{N}{2}} \left(N - 1 + 2\binom{N-1}{2} \right) = 2 - \frac{2}{N}. \tag{3.2}$$

It is also unique: If a tree is not a star, there is no vertex that is connected to all the remaining vertices. In this case, there are at least two vertices with distance $d \geq 3$, since in every tree there is a unique path connecting each vertex pair, and at the same time the number of neighboring vertices is the same as in the star graph. □

Using the same method as above, we can find the smallest average distance of a graph with N vertices and m edges, which we denote as $\bar{D}_{min}(N,m)$.

Theorem 1. *The minimum possible average distance of a graph $\mathcal{G}(N,m)$ is equal to $\bar{D}_{min}(N,m) = 2 - \frac{m}{\binom{N}{2}}$.*

Proof. The graph $\mathcal{G}(N, m)$ has m pairs of vertices with distance exactly 1, and the rest of the pairs of vertices (u, v) have distances $d(u, v) \geq 2$. Consequently, its average distance is

$$\mathcal{L}_{\mathcal{G}} \geq \frac{m + 2\left(\binom{N}{2} - m\right)}{\binom{N}{2}} = 2 - \frac{m}{\binom{N}{2}}. \tag{3.3}$$

This lower bound can always be achieved. A connected graph $\mathcal{G}(N, m)$ with at least one vertex with degree $d = N - 1$ has the star graph as an induced subgraph, so all non-neighboring vertices will have distance equal to 2. All connected vertices have distance equal to 1, leading to the lower bound of equation (3.3). □

Corollary 1. *If a graph $\mathcal{G}$ has at least one vertex pair (u, v) with distance $d(u, v) \geq 3$, then its average distance is $\mathcal{L}_{\mathcal{G}} > \bar{D}_{min}(N, m)$.*

Proof. The number of pairs with distance 1 is fixed, equal to the graph's size. All other vertices have a distance of at least 2, and the minimum is achieved when *all* non-neighboring pairs have distance equal to 2. □

The next three corollaries present sufficient conditions for a graph to have the smallest average shortest path length.

Corollary 2. *In a network with the smallest average distance, all vertex pairs are either connected, or connected to a common third vertex.*

Proof. From Corollary 1, all vertices that are not connected through an edge have distance equal to 2, which means that they have a common neighbor. □

Corollary 3. *A cut of a minimum average distance graph $\mathcal{G}$ divides its vertices into two disjoint sets where, in at least one of the sets, all vertices have at least one neighbor in the other.*

Proof. Assume that in both sets of a graph $\mathcal{G}$ there is at least one vertex which has no neighbors to the other set. The distance between these two vertices is at least 3, and according to Corollary 1, graph $\mathcal{G}$ will not have the smallest average shortest path length. □

Corollary 4. *Assume that the graph $\mathcal{G}$ of order N has the smallest average distance. The average degree $\bar{g}_{\mathcal{N}_u}$ of the neighbors of vertex u which has degree equal to d_u satisfies the inequality*

$$\bar{g}_{\mathcal{N}_u} \geq \frac{N-1}{d_u}. \tag{3.4}$$

Proof. Since every vertex u of $\mathcal{G}$ has distance exactly 2 with all its non-neighbors, the vertices in its neighbor set $\mathcal{N}_u = \{V_1, \ldots, V_{d_u}\}$ should be connected to all the remaining vertices. In other words, all the remaining $N-1-d_u$ vertices of the graph should have at least one common neighbor with u. Each neighbor V_k of u with degree g_k has $g_k - 1$ neighbors other than u, some of which may belong to $\mathcal{N}_u$. If we sum up the number of neighbors of all these vertices (excluding u), we get:

$$\begin{aligned} \sum_{k \in \mathcal{N}_u} (g_k - 1) &\geq N - 1 - d_u \\ \sum_{k \in \mathcal{N}_u} g_k &\geq N - 1 \\ d_u \bar{g}_{\mathcal{N}_u} &\geq N - 1 \\ \bar{g}_{\mathcal{N}_u} &\geq \frac{N-1}{d_u}. \end{aligned} \tag{3.5}$$

□

Corollary 5. *Networks that have the smallest possible average shortest path length are disassortative with respect to their degrees.*

Proof. It follows directly from equation (3.5) and the definition of a disassortative network. □

All nonisomorphic networks of order $N = 5$ and size $4 \leq m \leq 10$ with the minimum average shortest path length are shown in Figure 3.1.

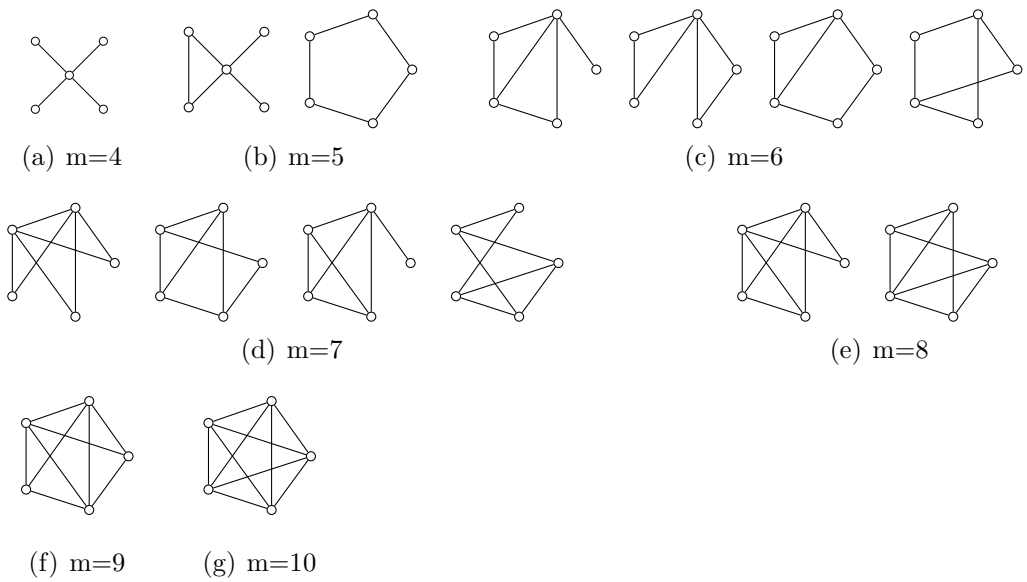


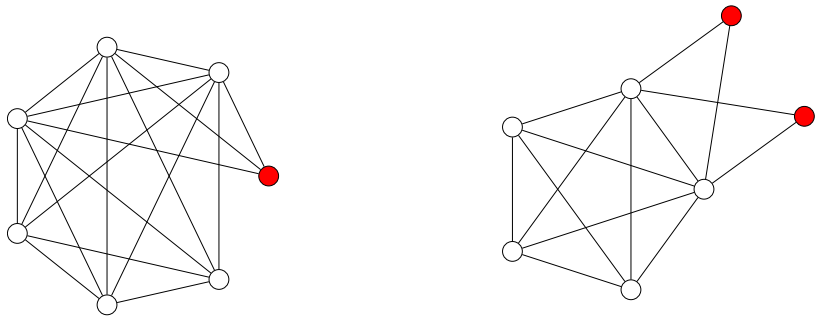
Figure 3.1: All connected networks with 5 vertices and $4 \leq m \leq 10$ edges with the smallest average shortest path length. In these graphs, all nodes that are not connected are second neighbors.

3.2.2 Maximum Average Distance

The networks with the largest average distance have a very different topology. They consist of two distinct connected subgraphs, and if we remove any edge, the network either becomes disconnected, or the previously connected vertices become second neighbors.

Definition 1. *We call a connected graph of order N and size m almost complete when its largest clique has order $N-1$ or $N-2$. We distinguish these two cases by calling them type I and type II, respectively. In order to be almost complete, a graph needs to have $\binom{N-1}{2}+1 \leq m \leq \binom{N}{2}-1$ (type I), or $\binom{N-2}{2}+2 \leq m \leq \binom{N-1}{2}$ (type II) edges. The vertices of the largest clique are called central vertices, whereas the vertices not belonging to it are called peripheral vertices.*

The two types of almost complete graphs are shown in Figure 3.2.



(a) Type I (b) Type II

Figure 3.2: **(a)** The type I almost complete graph consists of a clique of $N-1$ vertices, and one peripheral vertex (shown in red) that connects to them. **(b)** The type II almost complete graph of order N consists of a clique of order $N-2$, and two additional vertices that connect to it (and possibly to each other).

Lemma 3. *Assume that a vertex u with degree d_u is added to a network, with its neighbor set being $\mathcal{N}_u$. Rewiring edges of $\mathcal{G}$ such that they connect previously non-neighboring vertices in $\mathcal{N}_u$ cannot decrease its eccentricity or the average distance of u with the other vertices in the network.*

Proof. Connecting any two vertices in $\mathcal{N}_u$ will not change the distance of u with any of them. Furthermore, disconnecting a pair of vertices, at least one of which is not in $\mathcal{N}_u$, can only increase the distance of u with any of the vertices that do not belong to the set of its neighbors. □

More generally, connecting two non-neighboring vertices has the smallest impact on their average distance if they have a common neighbor. Rewiring an edge in $\mathcal{G}$ will increase the distance of the initially connected pair (u_1, v_1) to d (where $d \geq 2$), and decrease the distance of the new pair of vertices (u_2, v_2) with a common neighbor by

1. The overall difference will be

$$\begin{aligned}\Delta d(u_1, v_1) + \Delta d(u_2, v_2) &= (d_{new}(u_1, v_1) - d_{old}(u_1, v_1)) \\ &\quad + (d_{new}(u_2, v_2) - d_{old}(u_2, v_2)) \\ &= d - 2 \\ &\geq 0.\end{aligned} \tag{3.6}$$

Combining Lemma 3 with equation (3.6), we can easily see that for a fixed neighborhood $\mathcal{N}_u$ of a vertex u, we can increase the eccentricity of u and at the same time the average distance of the graph it belongs to, simply by rewiring edges to connect vertices in $\mathcal{N}_u$, until they form a clique.

Lemma 4. *All connected graphs of order $N \geq 2$ and size $\binom{N-1}{2} + 1 \leq m \leq \binom{N}{2}$ have the same average distance, equal to*

$$\bar{D}(N, m) = 2 - \frac{m}{\binom{N}{2}}. \tag{3.7}$$

Proof. Assume that the largest clique in $\mathcal{G}$ consists of C vertices, which we will call *central* vertices. The rest of the nodes belong to the set $\mathcal{P}$ of *peripheral* vertices, with $|\mathcal{P}| = P$ and they may form connections to the central vertices and among themselves. Since $m \geq \binom{N-1}{2} + 1$, every vertex in the graph is either a central or a peripheral vertex, and as a result

$$C + P = N. \tag{3.8}$$

The average distance of equation (3.7) is equal to the minimum possible distance of a graph as in equation (3.3), and it is achieved if and only if all non-neighboring vertices have distance equal to 2. The only way that the network will not have an average distance equal to $\bar{D}_{min}$ is when there is a pair of vertices A and B with shortest path length of at least 3. If there exist two such vertices, then from equation (3.3) and

Corollary 1 we conclude that the maximum average distance of the graph will be

$$D(\mathcal{G}) > 2 - \frac{m}{\binom{N}{2}}. \tag{3.9}$$

The central vertices are by definition fully connected to each other, and any peripheral vertex has distance two with all the central vertices it is not connected with. So, the only case where two non-neighboring vertices do not have any common neighbors is when both of them are peripheral vertices. We will now show that this is not possible.

For every peripheral vertex u, there are γ_u central vertices that are *not* connected to it. Also, let h be the total number of non-neighboring peripheral vertices. The total number of non-neighboring vertex pairs is

$$\gamma = h + \sum_{u \in \mathcal{P}} \gamma_u \tag{3.10}$$

with

$$\begin{aligned} \gamma &= \binom{N}{2} - m \\ &\leq \binom{N}{2} - \binom{N-1}{2} - 1 \\ &= N - 2. \end{aligned} \tag{3.11}$$

In addition,

$$h \geq 1 \tag{3.12}$$

since A and B are not connected. Combining all the equations above:

$$\begin{aligned} h + \sum_{u \in \mathcal{P}} \gamma_u \leq N - 2 \implies & \sum_{u \in \mathcal{P}} \gamma_u \leq N - 3 \\ \implies & \gamma_A + \gamma_B + \sum_{\substack{u \in \mathcal{P} \\ u \neq A, u \neq B}} \gamma_u \leq N - 3. \end{aligned} \tag{3.13}$$

Every peripheral vertex in $\mathcal{P}$ has at least one central vertex that it is *not* connected

to, so

$$\gamma_u \geq 1 \quad \forall u \in \mathcal{P} \tag{3.14}$$

and

$$\sum_{\substack{u \in \mathcal{P} \\ u \neq A, u \neq B}} \gamma_u \geq P - 2. \tag{3.15}$$

Based on the last two inequalities combined with inequality (3.13), we can derive an upper bound for the sum of γ_A and γ_B:

$$\begin{aligned} \gamma_A + \gamma_B &\leq N - P - 1 \\ &\leq N - 3 \end{aligned} \tag{3.16}$$

because $P \geq 2$. But A and B have by assumption no common neighbors in the clique or among any peripheral vertices, which means that

$$\gamma_A + \gamma_B \geq N - 2 \tag{3.17}$$

which is clearly a contradiction. □

Corollary 6. *There are exactly* $\left\lfloor \frac{N-2}{2} \right\rfloor$ *non-isomorphic graphs of order* N *and size* $m = \binom{N-1}{2}$ *with the largest possible average distance, equal to*

$$\bar{D}_{max}(N, m) = 2 - \frac{m - 1}{\binom{N}{2}}. \tag{3.18}$$

All other graphs of the same order and size have the minimum possible average distance among their vertices, equal to

$$\bar{D}_{min}(N, m) = 2 - \frac{m}{\binom{N}{2}}. \tag{3.19}$$

Proof. In a graph of size $m = \binom{N-1}{2}$, the total number of missing edges among all

the pairs of vertices is

$$\gamma = \binom{N}{2} - \binom{N-1}{2} = N - 1. \tag{3.20}$$

Keeping the same notation as before, we sum up all the missing edges among the peripheral vertices, and among peripheral and central vertices.

$$h + \gamma_A + \gamma_B + \sum_{\substack{u \in \mathcal{P} \\ u \neq A, u \neq B}} \gamma_u = N - 1 \tag{3.21}$$

under the constraints

$$\gamma_A + \gamma_B \geq N - P, \quad \sum_{\substack{u \in \mathcal{P} \\ u \neq A, B}} \gamma_u \geq P - 2 \quad \text{and} \quad h \geq 1. \tag{3.22}$$

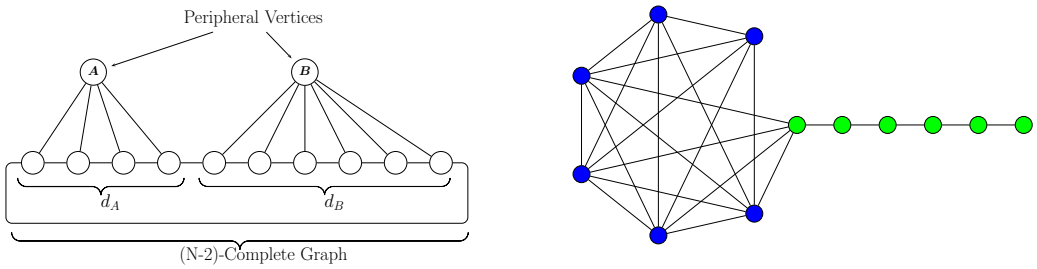
These inequalities can only be satisfied in equation (3.21) if all variables are equal to their respective lower bounds, namely

$$\gamma_A + \gamma_B = N - P, \quad \sum_{\substack{u \in \mathcal{P} \\ u \neq A, B}} \gamma_u = P - 2 \quad \text{and} \quad h = 1. \tag{3.23}$$

The only unknown variable above is P. Since A and B are not neighbors, there is only one ($h = 1$) edge missing among peripheral vertices. If we assume that $P \geq 3$, then A and B have exactly $P - 2$ common neighbors, which are peripheral vertices that are connected to both of them. This would clearly contradict our assumption. Consequently, A and B are the only peripheral vertices and $P = 2$. Such a graph is shown in Figure 3.3(a). It is clear from the previous analysis that

$$d_A + d_B + \gamma_A + \gamma_B = 2(N-2) \implies d_A + d_B = N - 2 \tag{3.24}$$

with $d_A, d_B \geq 1$ because the graph is connected. Setting $d_A \leq d_B$ in order to count



(a) (b)

Figure 3.3: **(a)** A network with size $m = \binom{N-1}{2}$ and largest possible average distance. Vertices A and B are the only vertices without any common neighbors, and $d_A + d_B = N-2$, the number of central vertices. **(b)** The graph of order $N = 12$ and size $m = 24$, with the largest average shortest path length. It consists of a complete graph of order $C = 6$ (blue), and a path graph of order $P = N - C = 6$ (green). Four edges ($\alpha = 4$) connect the complete subgraph to one of the two ends of the path graph.

only the non-isomorphic graphs, it is clear that there are exactly $\left\lfloor \frac{N-2}{2} \right\rfloor$ pairs of degrees d_A, d_B that satisfy the last equation. □

Theorem 2. *The graph of order N and size $N - 1 \leq m \leq \binom{N-1}{2}$ with the largest average distance among its vertices consists of a complete subgraph of order C, and a path subgraph of order $P = N - C$. The two subgraphs are connected through α edges, as shown in Figure 3.3(b). In addition, the graph with the maximum average shortest path length is unique for $N - 1 \leq m \leq \binom{N-1}{2} - 1$.*

Proof. Every arbitrary cut S will produce two disjoint subgraphs, both of which need to be maximum distance graphs for the respective orders and sizes. More formally, if $\mathcal{A}$ is the set of all networks of all orders and sizes with the maximum possible average shortest path length and $\mathcal{H}$ is an induced subgraph of a graph $\mathcal{G}$, then

$$\mathcal{G} \in \mathcal{A} \iff \mathcal{G} - \mathcal{H} \in \mathcal{A} \qquad \forall \quad \mathcal{H} \subseteq \mathcal{G}. \tag{3.25}$$

The above equation is a necessary and sufficient condition for maximum average

distance. If it does not hold for some subgraph $\mathcal{J} \subseteq \mathcal{G}$, then we would be able to rearrange the edges in it, so that the average distance among the vertices in the subgraph is increased. Since this would also increase the average distance of $\mathcal{G} - \mathcal{J}$ with the vertices of $\mathcal{J}$, the overall average distance of $\mathcal{G}$ would increase.

Now suppose that we want to find the maximum average distance graph of order N. According to the equation above, and setting one of the vertices u as the chosen subgraph (of unit order), a graph with order N and size m has the largest possible average distance (in which case it is denoted $\mathcal{G}_{max}$) when

$$\mathcal{G}_{max}(N,m) = \underset{\mathcal{G}\in\mathcal{C}(N,m)}{\arg\max} \left[\sum_{(u,v)\in\mathcal{V}^2(\mathcal{G})} d(u,v) \right], \tag{3.26}$$

where $\mathcal{C}(N,m)$ is the set of all possible connected graphs of order N and size m. But from equation (3.25), and considering a subgraph $\mathcal{H}$ of order 1, we can write the last condition as

$$\mathcal{G}_{max}(N,m) = \max_{\mathcal{N}_u} \Big[\mathcal{G}_{max}(N-1, m-|\mathcal{N}_u|) \cup \mathcal{H}(1,\mathcal{N}_u) \Big]. \tag{3.27}$$

We will now find the neighborhood $\mathcal{N}_u$ of vertex u in order to yield the graph with the largest average distance. We will use induction. For $N < 4$, the theorem holds trivially. For order $N = 4$, it is easy to check that graphs of all sizes have the structure of the theorem.

Assume that all the maximum average distance graphs up to order N_0 and size m_0 have the same form described above, where

$$N_0 = N - 1 \text{ and } N_0 - 1 \leq m_0 \leq \binom{N_0}{2}. \tag{3.28}$$

It will be shown that all networks of order N also have that same form, making use of equations (3.25) and (3.26). If $d_u = 1$, then we can connect it to the vertex w with the

largest eccentricity. In the resulting graph, u will now have the largest eccentricity and average distance to the other vertices. At the same time the new graph will have the form stated in the theorem and the sum of distances of u with the rest of the vertices will be

$$D_u = \sum_{\substack{v \in \mathcal{V}(\mathcal{G}) \\ v \neq u}} d(u,v) = 1 + \sum_{\substack{v \in \mathcal{V}(\mathcal{G}) \\ v \neq u,w}} (1 + d(w,v)). \tag{3.29}$$

If the degree of u is equal to the order of the clique, the resulting graph will have the largest average distance if we connect it to all the vertices of the clique, as shown in Lemma 3.

If d_u is smaller than the order of the clique, then u could be connected to clique vertices only, path vertices only, or a combination of both. None of the above is an optimal configuration, since they do not satisfy condition (3.25). The same argument holds when d_u is larger than the size of the clique. In this case we can subtract the order of the clique C, and consider a new vertex with degree $d_u - C$, repeating the process if needed. According to the above analysis, the new graph will either have the form stated in the theorem, or it will not have the largest average distance.

Finally for graphs with size $N - 1 \leq m \leq \binom{N-1}{2}$, the structure that yields the largest average distance is unique. Using induction again, we see that for $N = 4$, the claim holds. For $N \geq 5$, the graph with maximum average distance is unique for $N - 1$ by the induction hypothesis, and adding one extra vertex u with $d_u = 1$ or $d_u = C$ yields the same graph in both cases:

$$\mathcal{G}_{max}(N-1, m-C) \cup \mathcal{H}(1, C) \equiv \mathcal{G}_{max}(N-1, m-1) \cup \mathcal{H}(1, 1). \tag{3.30}$$

□

Note that according to condition (3.25), the network should have the same form no matter which subset of vertices we remove. The form of a graph with the largest average distance as stated in Theorem 2 is the one that satisfies that requirement.

The networks with the maximum average distance can be described as a combination of a type I almost complete subgraph and a path subgraph. We can now summarize the form of the networks with the largest average distance for any number of edges.

Corollary 7. *A graph $\mathcal{G}(N,m)$ with the largest average distance consists of a clique connected to a path graph as described in Theorem 2 (see Figure 3.3(b)) and is unique for $N-1 \leq m \leq \binom{N-1}{2}-1$. If $m=\binom{N-1}{2}$, then it consists of a clique of order $N-2$ and two peripheral vertices as shown in Figure 3.3(a). If $m \geq \binom{N-1}{2}+1$, then all graphs have the same average distance.*

Corollary 8. *Networks with the largest average shortest path length are assortative with regard to their degrees.*

The maximum possible average shortest path length of a graph is computed in the next corollary, where we also find the order of its clique and path subgraphs.

Corollary 9. *The average shortest path length among the vertices of a network with the largest possible average distance $\mathcal{G}_{max}(N,m)$ of order N and size m, is equal to*

$$\bar{D}_{max}(N,m)=\frac{\binom{C}{2}+\binom{P+1}{2}+(C-\alpha)P+\binom{P+1}{3}}{\binom{N}{2}}, \tag{3.31}$$

where

$$C=\left\lfloor \frac{3+\sqrt{9+8m-8N}}{2} \right\rfloor \tag{3.32}$$

is the number of vertices that belong to the clique,

$$P=N-C \tag{3.33}$$

is the number of vertices of the path subgraph, and

$$\alpha=m-P+1-\binom{C}{2} \tag{3.34}$$

is the number of edges that connect the clique with the path graph.

Proof. We will find the lengths of the shortest paths among all vertices, add them, and finally divide them by their number to find the average. First, we need to find the order of the clique. Summing up all the edges of the network, we have

$$\binom{C}{2} + \alpha + (P-1) = m. \tag{3.35}$$

The total number of vertices is

$$C + P = N \tag{3.36}$$

and replacing P in equation (3.35), we get

$$\binom{C}{2} + \alpha + (N-C-1) = m \tag{3.37}$$

where C and α are integers satisfying the inequalities

$$1 \leq C \leq N-1 \quad , \quad 1 \leq P \leq N-1 \tag{3.38}$$

and

$$1 \leq \alpha \leq C-1, \tag{3.39}$$

respectively. Solving for C:

$$C^2 - 3C + (2N - 2m + 2 - 2\alpha) = 0. \tag{3.40}$$

One way to find the solution of the second-order equation above, is to set α equal to its smallest possible value, and solve for C, keeping in mind that it is always a positive integer. As we add more edges, α increases while C stays unchanged, until the vertex of the path subgraph is connected to all the vertices of the clique. At this point, C increases by one and α changes from $C-1$ to 1. We set $\alpha = 1$, and taking

into account that $C \in \mathbb{N}^*$,

$$C = \left\lfloor \frac{3 + \sqrt{9 + 8m - 8N}}{2} \right\rfloor. \tag{3.41}$$

We can now compute the number of the vertices that do not belong to the clique, and the number of edges between the two subgraphs α from equation (3.35). The distance among each pair of the C vertices of the clique is 1, so the sum of the pairwise distances is

$$D_1 = \binom{C}{2}. \tag{3.42}$$

The sum of the shortest path lengths of the path subgraph vertices to the clique vertices is

$$\begin{aligned} D_2 &= \sum_{x=1}^{P} [x\alpha + (x+1)(C-\alpha)] = \sum_{x=1}^{P} [(C-\alpha) + xC] \\ &= P(C-\alpha) + C\binom{P+1}{2}. \end{aligned} \tag{3.43}$$

Finally, the sum of the shortest path lengths of nodes of the path subgraph is

$$D_3 = \sum_{x=1}^{P} \sum_{y=1}^{x} (y-x) = \sum_{x=1}^{P} \sum_{z=0}^{x-1} z = \sum_{x=1}^{P} \binom{x}{2} = \binom{P+1}{3}. \tag{3.44}$$

Adding all the sums of all the shortest path lengths, and dividing by the total number of vertex pairs, we get

$$\bar{D}_{max}(N,m) = \frac{\binom{C}{2} + C\binom{P+1}{2} + P(C-\alpha) + \binom{P+1}{3}}{\binom{N}{2}}. \tag{3.45}$$

□

It is easy to show that when $m \geq \binom{N-1}{2}$, the formula for the minimum and maximum average distance give the same result for the average distance, in accordance with Lemma 4. In that case, equation (3.31) assumes that the network is an almost complete graph, but this graph has the same average distance as any other graph of

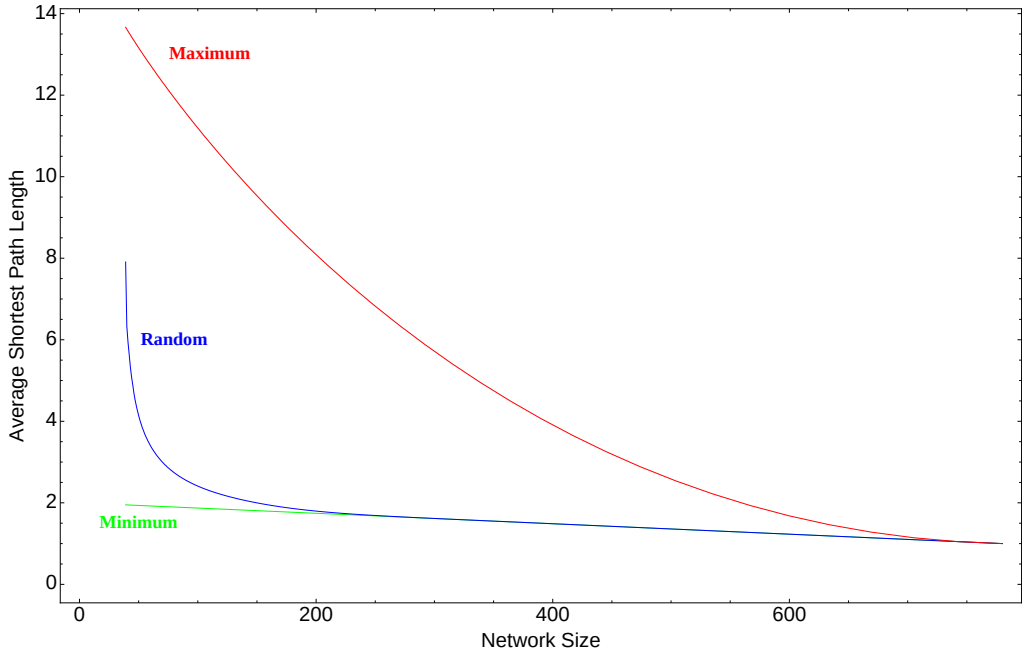


Figure 3.4: Tight bounds on the average distance of a graph with $N = 40$ vertices and $39 \leq m \leq 780$ edges. These bounds have been computed analytically. The average shortest path length for random graphs has been estimated by finding the mean shortest path length of 10^4 randomly generated graphs of the same order and size. The expected average distance of a random graph is very close to the minimum, even for relatively sparse networks. For graphs with edge density $\rho > 0.25$, it is virtually identical to the minimum one.

the same order and size. An example that shows the tight upper and lower bounds of the average distance of a graph with $N = 40$ and $39 \leq m \leq 780$ vertices is shown in Figure 3.4.

3.3 Betweenness Centrality

The betweenness centrality of a vertex or an edge is a measure of how important this vertex or edge is for the communication and information propagation among the different parts of the network. It is based on counting the number of shortest paths among all pairs of vertices a given vertex or edge is a part of [48]. The betweenness centrality of a network is particularly important when we have message passing (or

more generally signal propagation) among various nodes of a network, since it indicates how important each vertex or edge is for the function of such a network, and how robust it is with respect to vertex or edge removal [28]. The vertex betweenness centrality is defined as

$$\mathcal{B}(u) = \sum_{\substack{(s,t)\in\mathcal{V}^2(\mathcal{G}) \\ s\neq u\neq t}} \frac{\sigma_{st}(u)}{\sigma_{st}}, \tag{3.46}$$

where σ_{st} is the number of shortest paths between vertices s and t and $\sigma_{st}(u)$ is the number of shortest paths between s and t that go through vertex u. Equation (3.46) computes the total number of shortest paths of all the pairs of vertices in the graph that go through a given vertex u. If there is more than one such path, we divide by their total number σ_{st}, since they are assumed to be equally important. The betweenness centrality of a vertex is sometimes normalized by the total number of all vertex pairs that we took into account for computing it, which is equal to $\binom{N-1}{2}$.

$$\mathcal{B}_{norm}(u) = \frac{1}{\binom{N-1}{2}} \sum_{\substack{(s,t)\in\mathcal{V}^2(\mathcal{G}) \\ s\neq u\neq t}} \frac{\sigma_{st}(u)}{\sigma_{st}}. \tag{3.47}$$

The vertex betweenness is always nonnegative. The only vertices with betweenness centrality equal to zero are the ones with degree equal to 1. In order to assess the betweenness centrality of a network, we find the average of all vertices:

$$\mathcal{B}^v(\mathcal{G}) = \frac{1}{N} \sum_{u\in\mathcal{V}(\mathcal{G})} \mathcal{B}(u). \tag{3.48}$$

Networks with a large betweenness centrality usually have few vertices that play a major role in the communications among every other vertex. Conversely, small betweenness centrality indicates that the vertices of the network tend to be equally important or that there are many different shortest paths among the various parts of the network.

The edge betweenness centrality is similarly defined as the sum of the fraction of

shortest paths of all vertex pairs in the network that go through a given edge:

$$\mathcal{B}(f) = \sum_{\substack{(s,t)\in\mathcal{V}^2(\mathcal{G}) \\ s\neq t}} \frac{\sigma_{st}(f)}{\sigma_{st}} \tag{3.49}$$

where in this case $\sigma_{st}(f)$ is the number of shortest paths between s and t that go through edge f. The edge betweenness centrality of the network is defined in the same manner as before:

$$\mathcal{B}^e(\mathcal{G}) = \frac{1}{m} \sum_{f\in\mathcal{E}(\mathcal{G})} \mathcal{B}(f). \tag{3.50}$$

The betweenness of an edge is always positive for a connected network.

The betweenness centrality of a graph is an important proxy of how robust the network is to random vertex or edge removals. Removing a vertex or an edge with large betweenness centrality means that the communication among many vertex pairs will be affected, since they will now be forced to exchange information through alternative, possibly longer paths. Graphs which include nodes or edges with large betweenness centralities are sensitive to random removal of that set of vertices or edges. The vertex or edge betweenness centrality of a graph does not give any information about the centralities of individual vertices or edges, which may largely vary from edge to edge. For networks with the same betweenness centrality, large variations among vertices or edges reveal a sensitivity to targeted attacks, since removing the most central vertices may significantly disrupt the network function. In this section we show that the betweenness centrality of a graph is inherently related to its average shortest path length.

Theorem 3. *The average betweenness centrality of a network $\mathcal{G}(N, m)$ is a linear function of its average distance,*

$$\mathcal{B}(\mathcal{G}) = \frac{(N-1)(\bar{D}(\mathcal{G})-1)}{2}. \tag{3.51}$$

Proof.

$$\begin{aligned}
\mathcal{B}(\mathcal{G}) &= \frac{1}{N}\sum_{u\in\mathcal{V}(\mathcal{G})}\mathcal{B}(u) = \frac{1}{N}\sum_{u\in\mathcal{V}(\mathcal{G})}\sum_{\substack{(s,t)\in\mathcal{V}^2(\mathcal{G})\\ s\neq u\neq t}}\frac{\sigma_{st}(u)}{\sigma_{st}} \\
&= \frac{1}{2N}\sum_{u\in\mathcal{V}(\mathcal{G})}\sum_{\substack{s\in\mathcal{V}(\mathcal{G})\\ s\neq u}}\sum_{\substack{t\in\mathcal{V}(\mathcal{G})\\ t\neq u\\ t\neq s}}\frac{\sigma_{st}(u)}{\sigma_{st}} = \frac{1}{2N}\sum_{s\in\mathcal{V}(\mathcal{G})}\sum_{\substack{t\in\mathcal{V}(\mathcal{G})\\ t\neq s}}\frac{1}{\sigma_{st}}\sum_{\substack{u\in\mathcal{V}(\mathcal{G})\\ u\neq s\\ u\neq t}}\sigma_{st}(u) \\
&= \frac{1}{2N}\sum_{s\in\mathcal{V}(\mathcal{G})}\sum_{\substack{t\in\mathcal{V}(\mathcal{G})\\ t\neq s}}\frac{1}{\sigma_{st}}\sigma_{st}\left(|\mathcal{P}(s,t)|-1\right) = \frac{1}{2N}\sum_{s\in\mathcal{V}(\mathcal{G})}\sum_{\substack{t\in\mathcal{V}(\mathcal{G})\\ t\neq s}}\left(d(s,t)-1\right) \\
&= \frac{1}{2N}\left[2\binom{N}{2}\bar{D}(\mathcal{G}) - 2\binom{N}{2}\right].
\end{aligned} \tag{3.52}$$

Simplifying the last equation, the average betweenness centrality of a graph becomes the one stated in the theorem. □

It is worth mentioning that the average betweenness centrality of the network is only dependent on its size indirectly, through the average distance of the graph. For a fixed order, the average betweenness centrality of a network decreases as we add new edges (Lemma 1).

Corollary 10. *A network has minimum (maximum) average betweenness centrality if and only if it has minimum (maximum) average distance. The minimum possible average betweenness centrality of a graph of order N and size m is equal to*

$$\mathcal{B}_{min}(\mathcal{G}) = \frac{N-1}{2} - \frac{m}{N} \tag{3.53}$$

and the maximum possible average betweenness centrality of such a graph is

$$\mathcal{B}_{max}(\mathcal{G}) = \frac{\binom{C}{2} + C\binom{P+1}{2} + P(C-\alpha) + \binom{P+1}{3}}{N} - \frac{N-1}{2} \tag{3.54}$$

where C, P, and α are defined in equations (3.32), (3.33), *and* (3.34), *respectively.*

Proof. The networks with the smallest or largest average betweenness centrality are the graphs with the smallest or largest average distance respectively. Replacing them from equations (3.3) and (3.31), the bounds for the average betweenness centrality of graphs follow. □

Corollary 11. *The minimum sum of betweenness centralities of all the vertices of a network is equal to the number of vertices that are not neighbors.*

Proof. From equations (3.48) and (3.53), we see that

$$\min_{\mathcal{G}\in\mathcal{C}(N,m)}\left[\sum_{u\in\mathcal{V}(\mathcal{G})}\mathcal{B}(u)\right] = N\cdot\mathcal{B}_{min}(\mathcal{G}) = \binom{N}{2} - m. \tag{3.55}$$

□

Theorem 4. *The average edge betweenness centrality of a network is directly proportional to the average distance of the network, equal to*

$$\mathcal{B}^e(\mathcal{G}) = \frac{1}{m}\binom{N}{2}\bar{D}(\mathcal{G}). \tag{3.56}$$

Furthermore, the minimum and maximum average edge betweenness centrality of a network of order N, and size m are, respectively

$$\mathcal{B}^e_{min}(N,m) = \frac{N(N-1)}{m} - 1 \tag{3.57}$$

and

$$\mathcal{B}^e_{max}(N,m) = \frac{\binom{C}{2} + C\binom{P+1}{2} + P(C-\alpha) + \binom{P+1}{3}}{m} \tag{3.58}$$

where C, P and α are the same as in equations (3.32), (3.33), *and* (3.34).

Proof. We follow the same method as in the proof of the vertex betweenness centrality:

$$\begin{aligned}
\mathcal{B}_e(\mathcal{G}) &= \frac{1}{m}\sum_{e\in\mathcal{E}(\mathcal{G})}\mathcal{B}(e) = \frac{1}{m}\sum_{e\in\mathcal{E}(\mathcal{G})}\sum_{\substack{(s,t)\in\mathcal{V}^2(\mathcal{G})\\ s\neq t}}\frac{\sigma_{st}(e)}{\sigma_{st}} \\
&= \frac{1}{m}\sum_{(s,t)\in\mathcal{V}^2(\mathcal{G})}\sum_{e\in\mathcal{E}(\mathcal{G})}\frac{\sigma_{st}(e)}{\sigma_{st}} = \frac{1}{m}\sum_{(s,t)\in\mathcal{V}^2(\mathcal{G})}\frac{1}{\sigma_{st}}\sum_{e\in\mathcal{E}(\mathcal{G})}\sigma_{st}(e) \\
&= \frac{1}{m}\sum_{(s,t)\in\mathcal{V}^2(\mathcal{G})}\frac{1}{\sigma_{st}}\sigma_{st}d(s,t) \\
&= \frac{1}{m}\binom{N}{2}\bar{D}(\mathcal{G}).
\end{aligned} \tag{3.59}$$

Replacing the average distance by its minimum and maximum bounds, we get equations (3.57) and (3.58), respectively. □

3.4 Efficiency

The efficiency of a network (as defined in [41]) is a metric that shows how fast a signal travels on average in the network, assuming constant speed from one vertex to another. It is the sum of the inverse distances of all vertex pairs in a network, normalized by the total number of such pairs:

$$\mathcal{F}(\mathcal{G}) = \frac{1}{N(N-1)}\sum_{\substack{u,v\in\mathcal{V}(\mathcal{G})\\ u\neq v}}\frac{1}{d_{u,v}}. \tag{3.60}$$

Network efficiency is also correlated with the fault tolerance of the network, in the sense of how the average distance of a network changes when one or more vertices are removed from the network. Network efficiency has been used to assess the quality of neural, communication, and transportation networks [41]. Below we are going to show that the most and least efficient networks are the ones with the smallest and largest average distance among their individual parts.

Theorem 5. *A graph $\mathcal{G} = \mathcal{G}(N, m)$ has the highest efficiency among all other graphs*

with the same order and size if and only if it is a graph of minimum average distance. The highest efficiency of a network of N vertices and m edges is equal to

$$\mathcal{F}_{max}(N,m) = \frac{1}{2} + \frac{m}{N(N-1)}. \tag{3.61}$$

Proof. We assign a distance matrix to every graph, with its (k,m) element being the distance between vertices k and m. For a graph $\mathcal{G} = \mathcal{G}_{N,m}$ with distance matrix D and the minimum average distance, the sum of all the distances among all the pairs of vertices is smaller or equal to that of any other random graph $\mathcal{R} = \mathcal{R}_{N,m}$ with distance matrix H.

$$\sum_{k<m} d_{km} \leq \sum_{k<m} h_{km}. \tag{3.62}$$

The function to be maximized is convex, which means that the maximum lies on one of the boundaries. Since we will be comparing only networks of the same order, we will focus on the sum of inverse distances among the vertices of each network.

$$\mathcal{F}'(\mathcal{G}) = \binom{N}{2}\mathcal{F}(\mathcal{G}). \tag{3.63}$$

If a network $\mathcal{R}$ is not a minimum average distance graph, then according to Corollary 1 there exists at least one pair of vertices (a,b) with $d(a,b) \geq 3$. The sum of the inverse shortest path lengths of such a network is

$$\begin{aligned} \mathcal{F}'(\mathcal{R}) &= \sum_{\substack{(u,v)\in\mathcal{V}^2(\mathcal{G}) \\ u\neq v}} \frac{1}{h_{uv}} = \sum_{k\geq 1} \frac{1}{k}|\mathcal{E}_k(\mathcal{G})| \\ &= m + \frac{1}{2}|\mathcal{E}_2(\mathcal{R})| + \sum_{k\geq 3} \frac{1}{k}|\mathcal{E}_k(\mathcal{R})|. \end{aligned} \tag{3.64}$$

On the other hand, the sum of the inverse distances of a minimum average distance

network $\mathcal{G}$ is

$$\begin{aligned}\mathcal{F}'(\mathcal{G}) &= \sum_{\substack{(u,v)\in\mathcal{V}^2(\mathcal{G})\\ u\neq v}} \frac{1}{d_{uv}} = m + \frac{1}{2}|\mathcal{E}_2(\mathcal{G})| \\ &= m + \frac{1}{2}\left(\binom{N}{2} - m\right) = \frac{1}{2}m + \frac{1}{2}\binom{N}{2}.\end{aligned} \tag{3.65}$$

The difference is therefore

$$\begin{aligned}\mathcal{F}'(\mathcal{G}) - \mathcal{F}'(\mathcal{R}) &= \left(\frac{1}{2}m + \frac{1}{2}\binom{N}{2}\right) - \left(m + \frac{1}{2}|\mathcal{E}_2(\mathcal{R})| + \sum_{k\geq 3}\frac{1}{k}|\mathcal{E}_k(\mathcal{R})|\right) \\ &\geq \frac{1}{2}\binom{N}{2} - \frac{1}{2}m - \frac{1}{2}|\mathcal{E}_2(\mathcal{R})| - \frac{1}{3}\sum_{k\geq 3}|\mathcal{E}_k(\mathcal{R})| \\ &= \frac{1}{2}m + \frac{1}{2}\binom{N}{2} - \frac{1}{2}|\mathcal{E}_2(\mathcal{R})| - \frac{1}{3}\left(\binom{N}{2} - m - |\mathcal{E}_2(\mathcal{R})|\right) \\ &= \frac{1}{6}\left(\binom{N}{2} - m - |\mathcal{E}_2(\mathcal{R})|\right) \\ &= \frac{1}{6}\sum_{k\geq 3}|\mathcal{E}_k(\mathcal{R})| \\ &> 0.\end{aligned} \tag{3.66}$$

This shows that a maximum efficiency graph is a minimum distance graph. Normalizing by the total number of vertex pairs, equation (3.61) follows. □

Theorem 6. *A network has the lowest possible efficiency if and only if it is a largest average distance graph.*

Proof. We will use the same method as in the proof for the form of networks with the largest average distance. When $\binom{N-1}{2}+1 \leq m \leq \binom{N}{2}$, then all networks have m pairs of connected vertices, $\binom{N}{2} - m$ pairs of vertices that are second neighbors, and there is no graph in which two vertices do not have any common neighbors, as shown in Lemma 4. This clearly shows that all networks of this size have the same efficiency, given by equation (3.61). For smaller size graphs, when $N-1 \leq m \leq \binom{N-1}{2}$, a

necessary and sufficient condition will be

$$\mathcal{G} \in \mathcal{I} \iff \mathcal{G} - \mathcal{H} \in \mathcal{I} \qquad \forall \quad \mathcal{H} \subseteq \mathcal{G} \tag{3.67}$$

with $\mathcal{I}$ being the set of networks with the lowest efficiency. If we consider a subgraph of order 1 (a single vertex), its average distance to all other vertices will be the largest when its degree is equal to 1. So, if $\mathcal{G} = \mathcal{H} \cup \{u\}$, with $\mathcal{H} \in \mathcal{I}$ and u is only connected to one other vertex in the graph (its distance to which is equal to 1), it is evident that it has to be connected to one of the vertices with the largest average distance, which at the same time has the largest eccentricity.

$$\begin{aligned} \mathcal{F}(\mathcal{G}) &= \mathcal{F}(\mathcal{H}) + \sum_{\substack{k \in \mathcal{V}(\mathcal{G}) \\ k \neq u}} \frac{1}{d_{ku}} \\ &= \mathcal{F}(\mathcal{H}) + 1 + \sum_{\substack{k \in \mathcal{V}(\mathcal{G}) \\ k \neq u, k \neq v}} \frac{1}{1 + d_{vu}}. \end{aligned} \tag{3.68}$$

The last equation shows that if v is the vertex of $\mathcal{H}$ with degree $d_v = 1$, then the new graph has the smallest possible efficiency. It follows that a network has the smallest efficiency if and only if it is a maximum average distance network. □

Corollary 12. *The smallest efficiency of a network of order N and size m is equal to*

$$\mathcal{F}_{min}(N, m) = \frac{\binom{C}{2} + \sum_{x=1}^{N-C} \frac{\alpha + xC}{x(x+1)} + \sum_{x=1}^{N-C} \sum_{y=1}^{x-1} \frac{1}{x-y}}{\binom{N}{2}} \tag{3.69}$$

with

$$C = \left\lfloor \frac{3 + \sqrt{9 + 8m - 8N}}{2} \right\rfloor \qquad \textit{and} \qquad \alpha = m + 1 - N + C - \binom{C}{2} \tag{3.70}$$

The proof Corollary 12 is similar to the proof of Corollary 9. The lowest and

highest efficiency bounds of a connected network with $N = 40$ vertices and $39 \leq m \leq 780$ edges is shown in Figure 3.5. The same figure also shows the expected efficiency of a random graph.

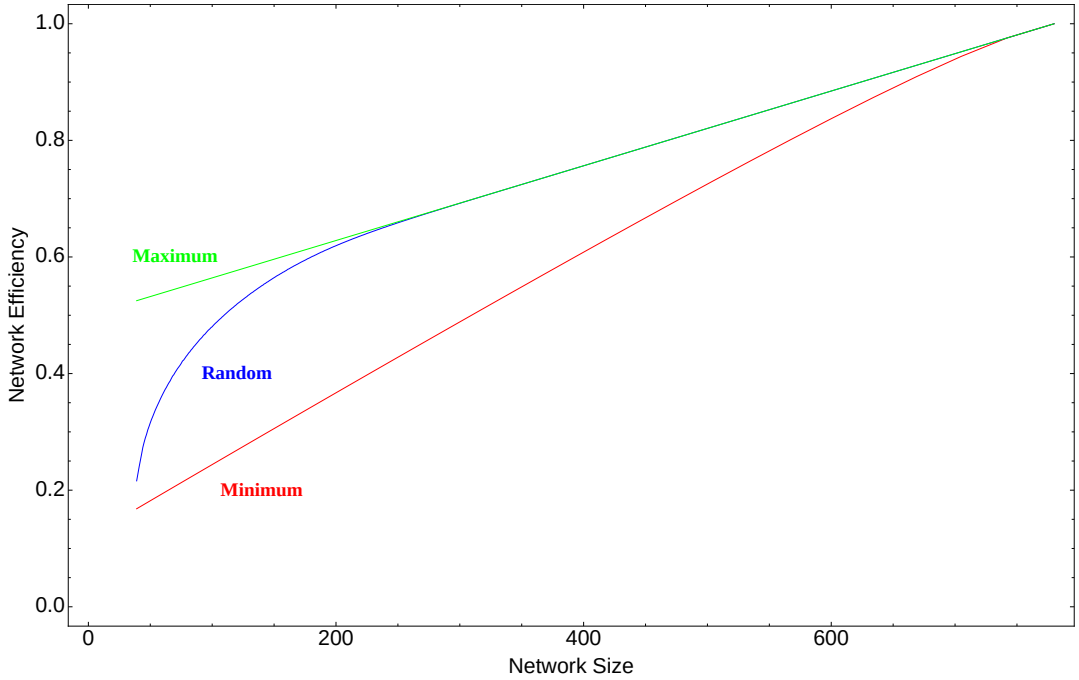


Figure 3.5: The smallest, average and largest efficiency of graphs of order $N = 40$ as a function of their size, $39 \leq m \leq 780$. The minimum and maximum bounds are analytically computed. The statistical average is approximated by the average of the efficiency of 10^4 random graphs for each size.

3.5 Radius and Diameter

The radius and the diameter of a graph are also measures that have to do with distance. In order to define the radius of a graph, we need to find a central vertex in the network, the one that is the closest to all other vertices. A network may have more than one central vertex. We are often interested in the radius of a network when information is aggregated and distributed from a vertex high in the hierarchy to other vertices lower in the hierarchy. The importance of a node is correlated with

how central it is. Important vertices are usually the ones closest to the network center.

On the other hand, the diameter of a network becomes important when we have a flat hierarchy, where communication or signal propagation takes place with the same frequency among any given pair of vertices in the network. There are applications in which we want our network to have very small or very large diameter. Usually for signal propagation or in general diffusion phenomena, the desired network topology has the smallest possible diameter, since the response to different inputs needs to be processed as fast as possible. When considering a virus spreading in the network during a fixed time interval, in order to ensure that as few nodes as possible get infected before appropriate action is taken, the network diameter needs to be as large as possible. In this section, we are going to show the structure of the networks with the largest and smallest radius and diameter. As we will see below, these graphs do not always have the same form.

3.5.1 Networks with the Smallest and Largest Radius

We will now find tight bounds for the radius of graphs of arbitrary order and size. The networks that achieve these bounds are generally not unique. The radius of a network is correlated with its average distance and diameter. Graphs with the smallest radius have the smallest average distance and the smallest diameter. Surprisingly, graphs with the largest radius may or may not have the largest average distance or diameter, as we will see next.

Lemma 5. *If* $(u, v) \in \mathcal{E}(\mathcal{G})$, *then*

$$ecc(u) - 1 \leq ecc(v) \leq ecc(u) + 1. \tag{3.71}$$

Proof. For every connected pair of vertices u, v such that $(u, v) \in \mathcal{E}(\mathcal{G})$, and a third

vertex $w \in \mathcal{V}(\mathcal{G})$,

$$\begin{aligned} & |d(v,w) - d(u,w)| \leq 1 \\ & \implies d(u,w) - 1 \leq d(v,w) \leq d(u,w) + 1 \\ & \implies \max_{w \in \mathcal{V}(\mathcal{G})} \{d(u,w) - 1\} \leq \max_{w \in \mathcal{V}(\mathcal{G})} d(v,w) \leq \max_{w \in \mathcal{V}(\mathcal{G})} \{d(u,w) + 1\} \\ & \implies ecc(u) - 1 \leq ecc(v) \leq ecc(u) + 1. \end{aligned} \tag{3.72}$$

□

Theorem 7. *A connected network of order N and size $m \geq N - 1$ has the smallest radius if and only if it has an induced subgraph which is the star graph. Such a network has a radius equal to one, regardless of its size.*

Proof. The radius $R(\mathcal{G})$ of any graph $\mathcal{G}$ is a natural number, with $R(\mathcal{G}) \geq 1$. If a star of the same order as $\mathcal{G}$ is an induced subgraph, then the central vertex has eccentricity equal to one, which is the minimum possible. Conversely, if the radius is equal to one, then there exists at least one vertex with full degree, which along with its neighbors forms a star subgraph. □

Corollary 13. *A network with the smallest radius also has the smallest average distance among its vertices. The opposite is not necessarily true, since there exist minimum average distance networks with no vertices of full degree (Corollaries 2–4).*

Lemma 6. *The maximum radius of graphs of order N is a non-increasing function of their size m.*

Proof. Assume that the graph $\mathcal{G} = \mathcal{G}(N, m)$ has the maximum radius among all other graphs with the same order and size. If we add one extra edge among any pair of currently non-neighboring vertices it will create a shorter path between at least two nodes, so the eccentricity of every vertex in $\mathcal{G}$ will either be unchanged or will decrease. The new graph will consequently have radius smaller or equal to $\mathcal{G}$, which

is also smaller or equal to the one of the maximum radius graph $\mathcal{H} = \mathcal{H}(N, m+1)$. So adding one or more edges cannot increase the maximum radius. □

Lemma 7. *Assume that $\mathcal{G}(N,m)$ has radius $R(\mathcal{G}) = r$, and c is a central vertex. If $d(a,c) = r$ for some $a \in \mathcal{V}(\mathcal{G})$, then there exists a vertex $b \in \mathcal{V}(\mathcal{G})$ such that*

$$d(b,c) \geq r-1 \quad \textit{and} \quad \mathcal{P}(a,c) \cap \mathcal{P}(b,c) = \{c\}. \tag{3.73}$$

Proof. Suppose that there does not exist such a vertex. If the first condition is not satisfied, then

$$d(u,c) \leq r-2 \quad \forall u \in \mathcal{V}(\mathcal{G}) \implies R(\mathcal{G}) \leq r-1 \tag{3.74}$$

which contradicts the assumption that $\mathcal{G}$ has radius r.

If there do not exist any vertices a and b with distances at least r and $r-1$, respectively, from c whose shortest paths to c have no other common vertex, then there exists a different vertex $w \in \mathcal{P}(a,c) \cap \mathcal{P}(b,c)$ such that

$$d(a,w) \leq r-1 \quad \text{and} \quad d(b,w) \leq r-2 \tag{3.75}$$

meaning that c is not a central vertex. □

Lemma 8. *A path graph has a radius larger or equal to any other network,*

$$R_{max}(N, m = N-1) = \left\lfloor \frac{N}{2} \right\rfloor. \tag{3.76}$$

A cycle graph has also radius larger or equal to any other network,

$$R_{max}(N, m = N) = R_{max}(N, m = N-1). \tag{3.77}$$

Proof. According to Lemma 7, the order of a network $\mathcal{G}$ with radius $R \geq \left\lfloor \frac{N}{2} \right\rfloor + 1$,

will be

$$|\mathcal{V}(\mathcal{G})| \geq 1 + 1 + \left\lfloor \frac{N}{2} \right\rfloor + \left\lfloor \frac{N}{2} \right\rfloor \geq N + 1 \tag{3.78}$$

which is clearly a contradiction. If the path graph has an odd number of vertices, the central vertex is the middle vertex, with distance $\frac{N-1}{2}$ from both extreme vertices. If the order is even, then both middle vertices are graph centers, and their eccentricities are equal to $\frac{N}{2}$. Connecting the two vertices that are furthest from the center through an edge does not have an impact to the graph radius, so a cycle has the largest possible radius (Lemma 6). In this case, because of the symmetry of the network, all vertices have the same eccentricity. □

Lemma 9. *A graph of order N and size $\binom{N}{2} - \left\lceil \frac{N}{2} \right\rceil + 1 \leq m \leq \binom{N}{2}$ has radius equal to 1.*

Proof. It suffices to prove that there exists at least one vertex with full degree. If all vertices have degree less than or equal to $N - 2$, the graph size is at most

$$m \leq m_0 = \left\lfloor \frac{N(N-2)}{2} \right\rfloor \tag{3.79}$$

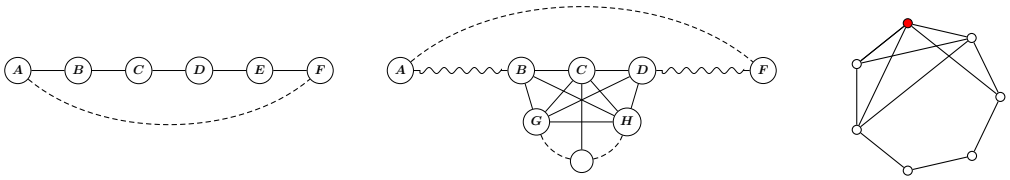
which is not possible, since for all $N \geq 2$

$$\binom{N}{2} - \left\lceil \frac{N}{2} \right\rceil + 1 > m_0. \tag{3.80}$$

□

Lemma 10. *The largest possible radius for a graph of order N and size $\binom{N-2}{2} + 1 \leq m \leq \binom{N}{2} - \left\lceil \frac{N}{2} \right\rceil$ is equal to 2.*

Proof. It suffices to prove that graphs with size $m \geq \binom{N-2}{2} + 1$ cannot have radius of 3 or larger, since we can find at least one network of this size in which no vertex has full degree [33]. Refering to Figure 3.6(a), let C be one of the central vertices.



(a) (b) (c)

Figure 3.6: **(a)** Necessary induced subgraph for a network to have radius larger or equal to 3. The vertices shown are not allowed to have any direct connections, and vertices with distance 3 are not allowed to have any common neighbors. **(b)** For order N and size $m \geq N$, there is always at least one graph $\mathcal{G}(N, m)$ that has the maximum possible radius, and has a full cycle as an induced subgraph. **(c)** A network with largest possible radius with $N = 7$ and $m = 11$. It consists of an almost complete subgraph of order $M = 5$ and a path graph of order $L = 4$. The two induced subgraphs share two vertices, so that $N = L + M - 2$. The red vertex has the largest degree, and is the center of the network.

According to Lemma 7, there exist at least two vertices A and F (possibly connected to each other) such that

$$d(C, A) \geq 2 \quad \text{and} \quad d(C, F) \geq 3. \tag{3.81}$$

In order to respect these distance conditions and the centrality of C,

$$C \not\sim F, \quad B \not\sim E, \quad \text{and} \quad A \not\sim D. \tag{3.82}$$

Because $d(C, F) \geq 3$, nodes C and F cannot have a common neighbor, so there are $N - 2$ edges that are *not* present in the graph. In addition, B and E may not have a common neighbor either, otherwise the radius would be at most equal to 2. There are another $N - 4$ edges that cannot be present, since there are $N - 2$ possible common neighbors of B and E, and we have already counted two of them previously. So the graph is missing at least $m_s = (N - 2) + (N - 4) + 3$ edges, and its size is at most

$$m \leq \binom{N}{2} - m_s = \binom{N}{2} - (2N - 3) = \binom{N - 2}{2}. \tag{3.83}$$

□

Lemma 11. *The maximum possible radius of a connected graph $\mathcal{G}$ of order N and size $N+1 \leq m \leq \binom{N-2}{2} - 2$ is*

$$R_{max}(N,m) = \left\lceil \frac{2N - 1 - \sqrt{1 + 8m - 8N}}{4} \right\rceil. \tag{3.84}$$

Proof. For every $N \geq 3$, there is at least one graph with radius equal to $R_{max}(N,m)$ that includes a full circle as an induced subgraph. To see why, suppose that C is a central vertex and

$$d(C,A) = R_{max} \quad \text{and} \quad d(C,F) \geq R_{max} - 1 \qquad A, F \in \mathcal{V}(\mathcal{G}) \tag{3.85}$$

as shown in Figure 3.6(b). We pick vertices B and D such that

$$d(C,B) = 1 \quad d(C,D) = 1, \text{ and } B \in \mathcal{P}(C,A), G \in \mathcal{P}(C,F). \tag{3.86}$$

Also, all other vertices can be connected to B, D, and C without changing the radius, when $m \leq \binom{N-2}{2} - 2$. Vertices A and F can be connected without changing the maximum radius of the network, as shown in Lemma 8. Also, there are no edges among vertices in $\mathcal{P}(C,A)$ or vertices in $\mathcal{P}(F,A)$, otherwise condition (3.85) would be violated. Thus, the network described has a full circle as an induced subgraph.

We now need to compute the maximum radius of such a graph. Since a new edge always creates new shortest paths, we have to connect vertices with distance equal to two, such that we only create a single new shortest path between vertices that are second neighbors (see also equation (3.6)). A simple method to find the graph with the largest radius is to start from a cycle graph, and keep adding edges such that we have a complete or almost complete graph connected with both ends of a path graph, as shown in Figure 3.6(c). Because of the symmetry of this network, the largest degree vertex is always a central vertex. If we denote with M the order of the

complete or almost complete subgraph, and with L the order of the path subgraph, we can find the order of these subgraphs by solving the following system of equations:

$$\begin{aligned} m &= (L-1) + \binom{M-1}{2} + \alpha \\ L + M &= N + 2 \end{aligned} \tag{3.87}$$

with

$$1 \leq \alpha \leq M-1, \quad L, C \in \mathbb{N}^*. \tag{3.88}$$

We compute M (and subsequently L) in the same way as in the proof of Corollary 9, by setting α equal to its maximum value, and then choosing the smallest integer M from the solution of the second-order equation.

$$M = \left\lfloor \frac{3 + \sqrt{1 + 8m - 8N}}{2} \right\rfloor, \quad L = N + 2 - \left\lfloor \frac{3 + \sqrt{1 + 8m - 8N}}{2} \right\rfloor. \tag{3.89}$$

This graph has radius equal to

$$R_{max}(N, m) = \left\lfloor \frac{L + \delta(a < M-1)}{2} \right\rfloor = \left\lfloor \frac{N + 2 + \delta(a < M-1) - \left\lfloor \frac{3+\sqrt{1+8m-8N}}{2} \right\rfloor}{2} \right\rfloor \tag{3.90}$$

where δ is the Kronecker delta. After simplifying, the last expression becomes:

$$R_{max}(N, m) = \left\lceil \frac{2N - 1 - \sqrt{1 + 8m - 8N}}{4} \right\rceil. \tag{3.91}$$

□

An example of the form of the function above for $N = 40$ is shown in Figure 3.7.

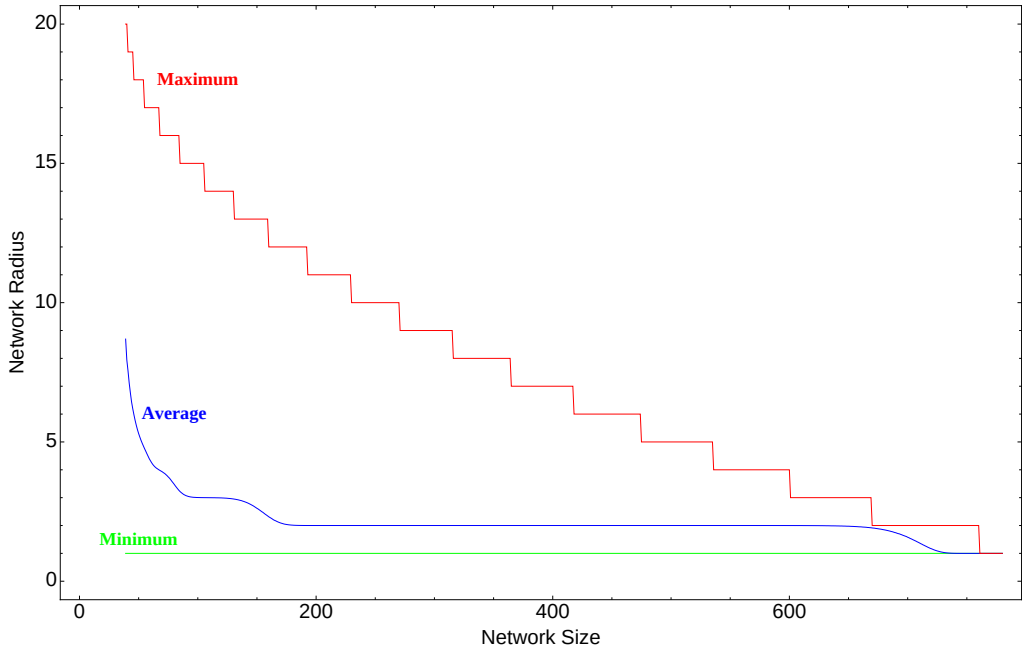


Figure 3.7: The largest, smallest, and average radius of graphs of order $N = 40$ as a function their size $39 \leq m \leq 780$. The minimum and maximum bounds are analytically computed. The statistical averages are estimated by the mean radius of 10^4 random graphs for each size. The average radius also has a "stepwise" form.

Theorem 8. *The maximum radius of a network of order N and size m is*

$$R_{max}(N,m) = \begin{cases} \lfloor \frac{N}{2} \rfloor & N-1 \leq m \leq N \\ \left\lceil \frac{2N-1-\sqrt{1+8m-8N}}{4} \right\rceil & N+1 \leq m \leq \binom{N-2}{2} \\ 2 & \binom{N-2}{2}+1 \leq m \leq \binom{N}{2} - \lceil \frac{N}{2} \rceil \\ 1 & \binom{N}{2} - \lceil \frac{N}{2} \rceil + 1 \leq m \leq \binom{N}{2}. \end{cases} \tag{3.92}$$

Note the same stair-like form of both the maximum radius and the statistical averages. The statistical average curve exhibits fewer and smoother "steps". Networks

with the largest radius of order $N = 5$ are listed in Figure 3.8.

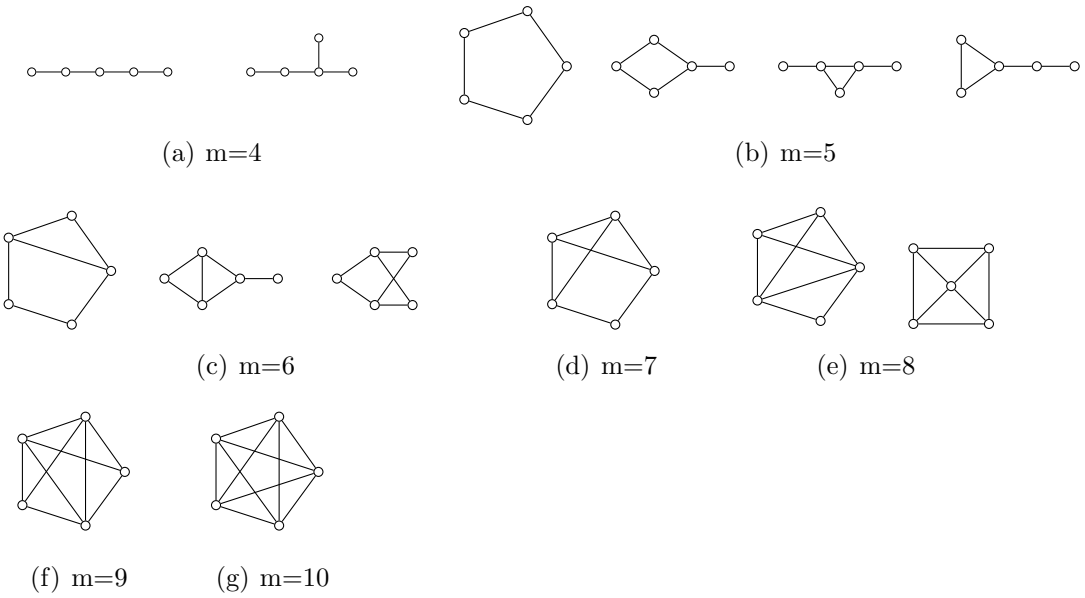


Figure 3.8: All connected networks with 5 vertices and $4 \leq m \leq 10$ edges with the largest possible radius. The first graph of each group is constructed by the method described in the text.

3.5.2 Networks with the Smallest and Largest Diameter

In this subsection, we are going to study the form of the networks with the minimum and maximum diameter. Computing the minimum diameter of a network is fairly straightforward. In the case of the maximum possible diameter, we first prove two lemmas that will help us show that we can find the structure of the networks recursively.

Theorem 9. *A network has the smallest possible diameter if and only if it is a smallest average distance graph.*

Proof. The diameter of a complete graph is trivially equal to one. If the graph is not complete, the diameter is at least 2, since there is at least one pair of non-

neighboring vertices. In a graph with the smallest average distance, all vertices that are not connected have at least one common neighbor, and the maximum eccentricity is equal to 2. Conversely, if the largest distance among any vertex pair is equal to 2, then by Corollary 1, the graph has the smallest average distance. □

Corollary 14. *A network with the minimum radius ($R(\mathcal{G}) = 1$) also has minimum diameter ($T(\mathcal{G}) = 2$) regardless of its interconnection topology. The inverse is not always true: There are networks with minimum diameter, and radius $R(\mathcal{G}) = 2 > R_{min}(N, m)$.*

An example of a network with minimum diameter and radius equal to 2 would be one where there is no vertex of full degree, but in which the distance among all pairs of vertices is equal to either 1 or 2.

Lemma 12. *A network of order N and size $\binom{N-1}{2} + 1 \leq m \leq \binom{N}{2} - 1$ has diameter equal to 2. A complete graph has diameter equal to 1.*

Proof. In a complete graph, all vertices are connected to each other, so the eccentricity of every vertex is trivially equal to 1. In a graph of size $m \geq \binom{N-1}{2} + 1$, all vertices that do not share an edge have at least one common neighbor, as shown in the proof of Lemma 4. Consequently, every vertex has eccentricity either 1 or 2, so the diameter is equal to 2 regardless of the graph topology. □

Lemma 13. *The maximum diameter $T_{max}(N, m)$ of a network of order N is at most one larger than the maximum diameter of a network with order $N - 1$ and smaller size.*

$$T_{max}(N, m) \leq T_{max}(N - 1, m - d) + 1, \quad 1 \leq d \leq N - 1. \tag{3.93}$$

Proof. Assume that the graph $\mathcal{G} = \mathcal{G}_{N,m}$ has diameter $T(\mathcal{G}) \leq T_{max}(N, m)$. Define as $\mathcal{D}$ the set of unordered vertex pairs whose distance is equal to the graph diameter. We now remove an arbitrary vertex u with degree $d = d_u$ from $\mathcal{G}$, and the resulting

graph is $\mathcal{H}$ with order $N-1$. If $u \in \mathcal{D}$, then no shortest path between any vertex pair in $\mathcal{D}$ passes through u. We distinguish two cases:

- If u is in every vertex pair in $\mathcal{D}$, then the diameter of $\mathcal{H}$ is

$$\begin{aligned} T(\mathcal{G}) &= T(\mathcal{H}) + 1 \qquad \forall \quad \mathcal{G}, \mathcal{H} \\ \implies T_{max}(N, m) &\leq T_{max}(N-1, m-d) + 1. \end{aligned} \tag{3.94}$$

- If there exists at least one vertex pair in $\mathcal{D}$ that does not include u, then removing u will result in a graph $\mathcal{H}$ that if it is connected, it has the same diameter as $\mathcal{G}$. If u is a member of all paths among of all vertex pairs with distance equal to the diameter, then $\mathcal{H}$ will be disconnected, and its diameter will be by definition infinite.

$$\begin{aligned} T(\mathcal{G}) = T(\mathcal{H}) &\leq T_{max}(N-1, m-d) \\ \implies T_{max}(N, m) &\leq T_{max}(N-1, m-d). \end{aligned} \tag{3.95}$$

Combining the two cases, the result follows. □

Corollary 15. *If we remove a vertex u from a graph $\mathcal{G}$, resulting in graph $\mathcal{H}$, then*

$$T_{\mathcal{H}} = T_{\mathcal{G}} - 1 \implies \mathcal{D}_{\mathcal{G}} = \{(u, w_1), (u, w_2), \ldots, (u, w_d)\}. \tag{3.96}$$

Conversely, if we add a vertex u with degree d to $\mathcal{H}$, then

$$T_{\mathcal{G}} = T_{\mathcal{H}} + 1 \implies \{(v_1, w_1), (v_2, w_2), \ldots, (v_d, w_d)\} \subseteq \mathcal{D}_{\mathcal{H}} \tag{3.97}$$

with $w_1, \ldots, w_d, v_1, \ldots v_d \in \mathcal{V}(\mathcal{H}), \mathcal{V}(\mathcal{G})$ and $(u, v_1), \ldots, (u, v_d) \in \mathcal{E}(\mathcal{G})$ and $\mathcal{D}$ the set of unordered vertex pairs whose distance is equal to the graph diameter.

Theorem 10. *The largest possible diameter of a network of order N and size m is*

equal to

$$T_{max}(N,m) = N - \left\lfloor \frac{1+\sqrt{9+8m-8N}}{2} \right\rfloor. \tag{3.98}$$

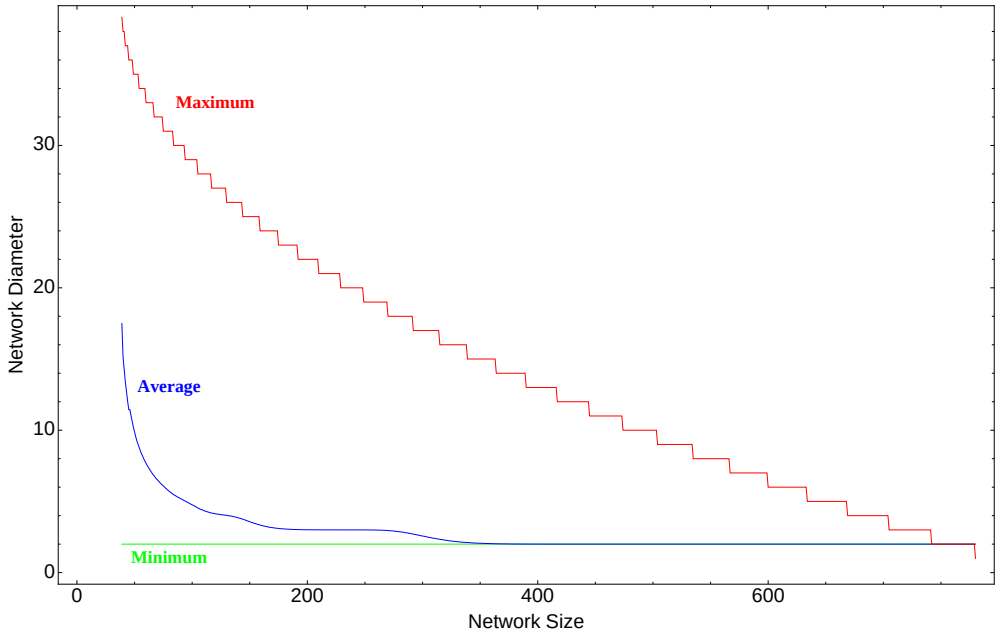


Figure 3.9: The smallest, average, and largest diameter of graphs of order $N = 40$ as a function of their size, $39 \leq m \leq 780$. The minimum and maximum bounds are analytically computed. The statistical average is approximated by the mean diameter of 10^4 random graphs of the respective size. Note that even in the average diameter, there is the "stepwise" form of the maximum diameter graphs, although much smoother.

Proof. Lemma 13 and Corollary 15 readily show an easy way to find the largest possible diameter of a graph of fixed order and size. According to Corollary 15, if we add a vertex u with degree $d_u = 1$ to a maximum diameter graph $\mathcal{H}$, and we connect it to a vertex with the largest eccentricity, the resulting graph $\mathcal{G}$ has also the largest diameter for its order and size (see also Lemma 5). As a result, we can write the following recursive relation:

$$T_{max}(N,m) = 1 + T_{max}(N-1, m-1). \tag{3.99}$$

Repeating the process as many times as possible,

$$T_{max}(N,m) = k + T_{max}(N-k, m-k). \tag{3.100}$$

The only reason why the recursion can only be implemented a finite number of times is that there cannot exist a network with $N-k-1$ vertices and $m-k-1$ edges with

$$m-k-1 > \binom{N-k-1}{2}. \tag{3.101}$$

This means that the subgraph $\mathcal{H}$ with $N-k$ vertices and order $m-k$ has size

$$\binom{N_{\mathcal{H}}-1}{2} \leq m_{\mathcal{H}} < \binom{N_{\mathcal{H}}}{2} \tag{3.102}$$

and consequently has a diameter of 2 (almost complete subgraph), so the maximum diameter of the graph is:

$$T_{max}(N,m) = k+2 \tag{3.103}$$

where $k = N - N_{\mathcal{H}}$. One of the graphs with the maximum diameter consists of a path graph with $N - N_{\mathcal{H}}$ vertices, and a subgraph of order $N_{\mathcal{H}}$ and size $m_{\mathcal{H}} \geq \binom{N_{\mathcal{H}}-1}{2} + 1$. If we assume without loss of generality that $\mathcal{H}$ is a type I almost complete graph, then $\mathcal{G}$ is a maximum distance network. It consists of a path graph of order $P = k+1$ and a complete graph of order C, such that $P + C = N$, as shown in Figure 3.3(a). If we denote by C and P the order of the clique and the path subgraphs as in equations (3.32) and (3.33), and combine them with equation (3.103), the maximum diameter of a graph $\mathcal{G}(N,m)$ is

$$\begin{aligned} T_{max}(N,m) &= k+2 = P+1 = N-C+1 \\ &= N - \left\lfloor \frac{1+\sqrt{9+8m-8N}}{2} \right\rfloor. \end{aligned} \tag{3.104}$$

□

Corollary 16. *A maximum average distance graph also has the largest possible diameter. The converse is not always true.*

An example of the minimum, maximum, and average diameter of graphs with 40 vertices and increasing number of edges is shown in Figure 3.9.

From the previous analysis we can conclude that the only tree with the largest diameter is the path graph. A graph with the largest diameter is not necessarily unique for any size $N \leq m \leq \binom{N}{2} - 2$. The list of all the graphs of order $N = 5$ and size $4 \leq m \leq 10$ are shown in Figure 3.10.

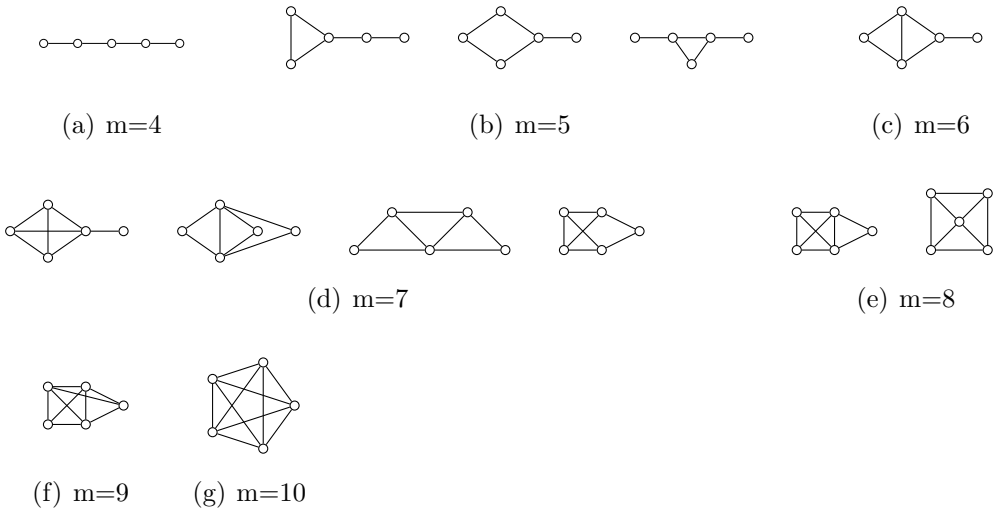
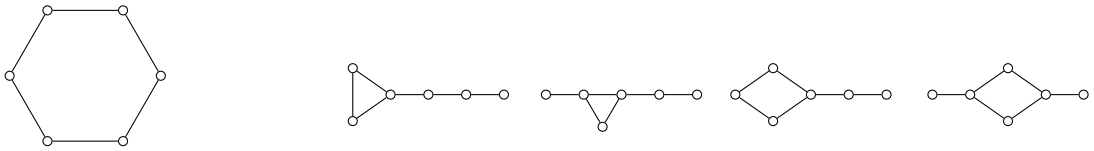


Figure 3.10: All connected networks with 5 vertices, $4 \leq m \leq 10$ edges, and the largest diameter. The first graph of each group is also a maximum average distance graph.

Comparing the graphs with the largest radius with the graphs with the largest diameter, we find that graphs with the largest radius and graphs with the largest diameter do not always have the same form.

Corollary 17. *A network with the largest diameter does not necessarily have the largest radius. Conversely, a network with the largest radius does not necessarily have the largest diameter.*

An example is given in Figure 3.11.



(a) (b)

Figure 3.11: **(a)** The only graph with $(N, m) = (6, 6)$ and radius $R_{max}(6, 6) = 3$. **(b)** All graphs with the same order and size and maximum diameter, equal to 4. No graph of the largest diameter coincides with the maximum radius network.

3.6 Resistance Distance

The resistance distance of a graph is a property that shows how easy it is to transfer information among different parts of the network, taking into account both the number of independent paths available and their length. It is an important quantity for the design of many kinds of networks, including electrical networks of all scales, from power networks to microelectronic circuits. First, we define the resistance distance between two nodes of a network, and the overall resistance distance of a graph.

Definition 2. *The resistance distance $s(u, v)$ between two vertices $u, v \in \mathcal{V}(\mathcal{G})$ of a graph $\mathcal{G}$ is equal to the total resistance between them when every edge represents a unit resistor.*

Definition 3. *The resistance distance of a graph is equal to the minimum among of the resistance distance of all the node pairs of the network.*

$$S(\mathcal{G}) = \min_{\substack{u,v \in \mathcal{V}(\mathcal{G}) \\ u \neq v}} s(u, v). \tag{3.105}$$

We can find the networks with the smallest resistance distance if we choose two connected vertices and make sure that they have the largest possible number of common neighbors, as the next theorem shows.

Theorem 11. *The smallest possible resistance distance of a simple connected graph*

$\mathcal{G} = \mathcal{G}_{N,m}$ *is*

$$S_{min}(N,m) = \begin{cases} \frac{2}{m-N+3} & N-1 \leq m \leq 2N-3 \\ \\ \frac{2}{N} & 2N-2 \leq m \leq \binom{N}{2}. \end{cases} \tag{3.106}$$

Proof. If $m = N-1$, $\mathcal{G}$ is a tree, and it is not possible to have any resistors connected in parallel, so $\mathcal{S}_{min}(N, N-1)$=1. Every tree is a minimum resistance distance graph as long as the endpoints of the circuit are two adjacent vertices. When $N \leq m \leq 2N-3$, the network will have $c = m - N + 1$ independent cycles. In order to make the resistance as small as possible, we need to choose the endpoints of the circuit to be two adjacent nodes, and in addition to be parts of cycles that are as short as possible. This is because the smaller the resistance of each branch of the cycle, the smaller the resistance between the two endpoints. Since a cycle has at least 3 vertices (two vertices cannot be connected with more than one edge, since the graphs are assumed to be simple), the cycles need to be of length 3, and all of them have a common edge, the endpoints of which are the endpoints of the circuit. As a result, the total resistance will be the combination of a unit resistor in parallel to $m - N + 1$ pairs of resistors connected in parallel, which means that

$$S_{min}(N,m) = \frac{1}{1 + \frac{1}{2}(m-N+1)} = \frac{2}{m-N+3}\ , \quad N-1 \leq m \leq 2N-3. \tag{3.107}$$

Adding more edges to the network has no effect on the total impedance, since all vertices except for the circuit endpoints will have the same potential, equal to half of the voltage difference applied to the circuit ends (assuming that we have arbitrarily set the potential of one of the endpoints equal to zero). Consequently, no current would flow among them, and we can write

$$S_{min}(N,m) = \frac{1}{1 + \frac{1}{2}((2N-3)-N+1)} = \frac{2}{N}\ , \quad m \geq 2N-3. \tag{3.108}$$

□

Corollary 18. *A graph with size $N-1 \leq m \leq 2N-3$ has minimum resistance distance if and only if it has a subgraph consisting of $m-N+1$ triangles that have one edge in common. The endpoints of the common edge are the endpoints of the circuit. For $2N-2 \leq m \leq \binom{N}{2}$, any graph with at least two vertices of full degree is a minimum resistance distance graph.*

We now turn our attention to the structure of the networks with the largest resistance distance. We will start from networks of relatively large size (almost complete graphs) and then move on to finding their form in the general case.

Lemma 14. *The maximum resistance of a network of fixed order is a decreasing function of the graph size.*

$$S_{max}(N,m) \geq S_{max}(N,m+g) \quad \textit{for } g \geq 0. \tag{3.109}$$

Proof. If $S_{max}(N,m) < S_{max}(N,m+g)$, we remove g edges from the network of larger size. This cannot decrease the total resistance, since there is now no voltage drop between the vertices that were previously connected. Therefore, the resulting graph will have larger resistance than the resistance of the initial network of smaller size, which contradicts the initial assumption. □

Lemma 15. *The largest possible resistance distance of a network with size $m \geq \binom{N-1}{2}+1$ is*

$$S_{max}(N,m) = \frac{\alpha + C + 1}{\alpha C} \tag{3.110}$$

where

$$C = N-1 \quad \textit{and} \quad \alpha = m - \binom{N-1}{2}. \tag{3.111}$$

Proof. The almost complete graph will have the largest possible resistance when the endpoints of the circuit are the peripheral vertex and a central vertex it is *not* connected to. The reason is that any other graph will have more shorter paths to the ground vertex, and thus smaller total resistance [22]. In the almost complete graph, because of the symmetry, all the vertices that are neighbors of the peripheral vertex have the same potential. Similarly, all vertices that are *not* neighbors of the peripheral vertex, except for the ground vertex also have the same potential. The ground vertex has potential equal to zero. Thus, we may remove the edges between the α neighbors of the peripheral vertex, and the $N-2-\alpha$ non-neighbors of the peripheral vertex, excluding the ground vertex. Then, we merge the edges that connect these three sets of vertices, as if they were connected in parallel. The result is shown in Figure 3.12. The total resistance of the transformed network is

$$\begin{aligned} S_{max}(N,m) &= \frac{1}{\alpha} + \frac{1}{\frac{1}{\frac{1}{\alpha}} + \frac{1}{\frac{1}{\alpha(N-2-\alpha)} + \frac{1}{N-2-\alpha}}} \\ &= \frac{1}{\alpha} + \frac{1}{\alpha + \frac{\alpha(N-2-\alpha)}{\alpha+1}} \\ &= \frac{N+\alpha}{\alpha(N-1)}. \end{aligned} \tag{3.112}$$

□

Lemma 16. *The maximum resistance of a network $\mathcal{G} = \mathcal{G}(N,m)$ can be at most one larger than the maximum resistance $S_{max}(N-1,m-1)$, provided that both networks are connected. More precisely*

$$S_{max}(N,m) \leq S_{max}(N-1,m-1) + 1. \tag{3.113}$$

Proof. We start from the network $\mathcal{H}(N-1,m-1)$ with the largest resistance, and add one extra vertex with unit degree to one of its endpoint vertices. If the Lemma

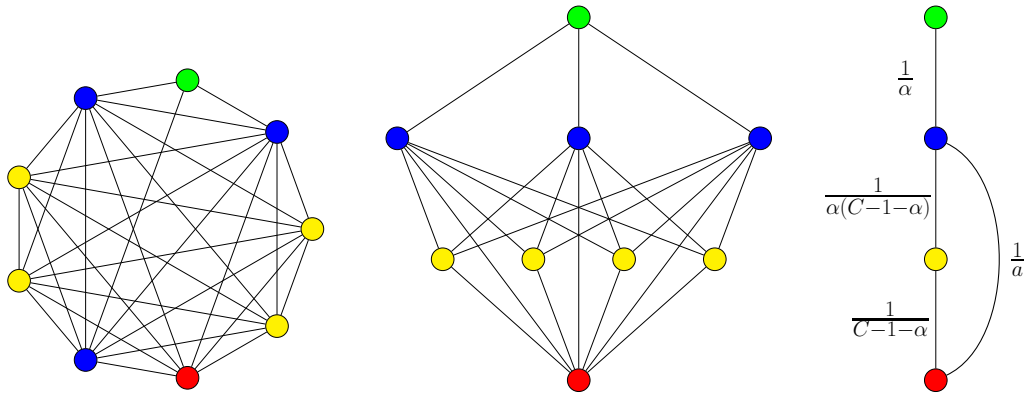


Figure 3.12: **(a)** An almost complete graph has the largest resistance distance among all other graphs of the same order and size. The endpoints with that resistance are the peripheral vertex (green) and an arbitrary central vertex that is *not* connected to it (red). The central vertices that the peripheral vertex connects to (blue), all have the same potential. All central vertices that are not connected to the peripheral vertex (yellow) also have the same potential. **(b)** Taking into consideration the symmetry of the circuit, we can remove the edges among vertices that belong to the same group. **(c)** The previous circuit can be simplified by collapsing all nodes of one group to one "super-node", considering all resistors that connect different groups to be connected in parallel. The weights of the edges correspond to the resistances among the "super-nodes".

does not hold, then the resulting graph $\mathcal{G} = \mathcal{G}(N, m)$ will have resistance

$$S(\mathcal{G}) = 1 + S_{max}(N-1, m-1) < S_{max}(N, m). \tag{3.114}$$

Now assume that the network $\mathcal{K}$ with the maximum resistance has vertex u with degree d_u as an endpoint. If $d_u = 1$, then network $\mathcal{H}$ was not a maximum resistance network and the Lemma is proved. If $d_u > 1$, then the potential difference between u and all its neighbors will be smaller or equal to one, since the current flowing through u will be divided among all the resistors that are adjacent to u. Consequently, removing vertex u along with its edges, we can find a relation with the resulting network $\mathcal{M}$

$$S_{max}(N, m) \leq \epsilon + S(\mathcal{M}) \quad \leq \epsilon + S_{max}(N-1, m-d_u) \tag{3.115}$$

with $0 \leq \epsilon \leq 1$. Combining equations (3.114) and (3.115), it is evident that

$$S_{max}(N-1, m-1) < S_{max}(N-1, m-d_u) - (1-\epsilon) \tag{3.116}$$

which, according to Lemma 14, is not possible. □

Theorem 12. *The largest possible resistance distance of a simple connected graph* $\mathcal{G} = \mathcal{G}_{N,m}$ *is*

$$S_{max}(N, m) = P - 1 + \frac{\alpha + C + 1}{\alpha C} \tag{3.117}$$

where C, P *and* α *are defined in equations* (3.32)*,* (3.33)*, and* (3.34)*, respectively.*

Proof. Using the Lemmas above, we can analytically compute the largest resistance, and find the form of the graphs that have it. We repeatedly apply Lemma 16, until we are left with a type I almost complete graph, at which point we can make use of Lemma 15.

$$S_{max}(N, m) \leq k + S_{max}(N-k, m-k). \tag{3.118}$$

We can achieve the equality in the last equation by connecting a new vertex with unit degree to one of the endpoints of the previous graph, such that

$$k = P - 1 \quad \text{and} \quad N - k = C + 1 \tag{3.119}$$

where the total resistance is the sum of a path graph with P vertices, and an almost complete graph with the respective values of C and α (see also condition (3.101)). The form of these maximum resistance networks are easy to identify: We recursively connect in series a unit resistor in one of the two endpoints of the previous circuit. □

An example showing the minimum, average and maximum resistance for networks of order $N = 40$ and increasing size is shown in Figure 3.13.

Corollary 19. *The number of non-isomorphic graphs with the largest resistance dis-*

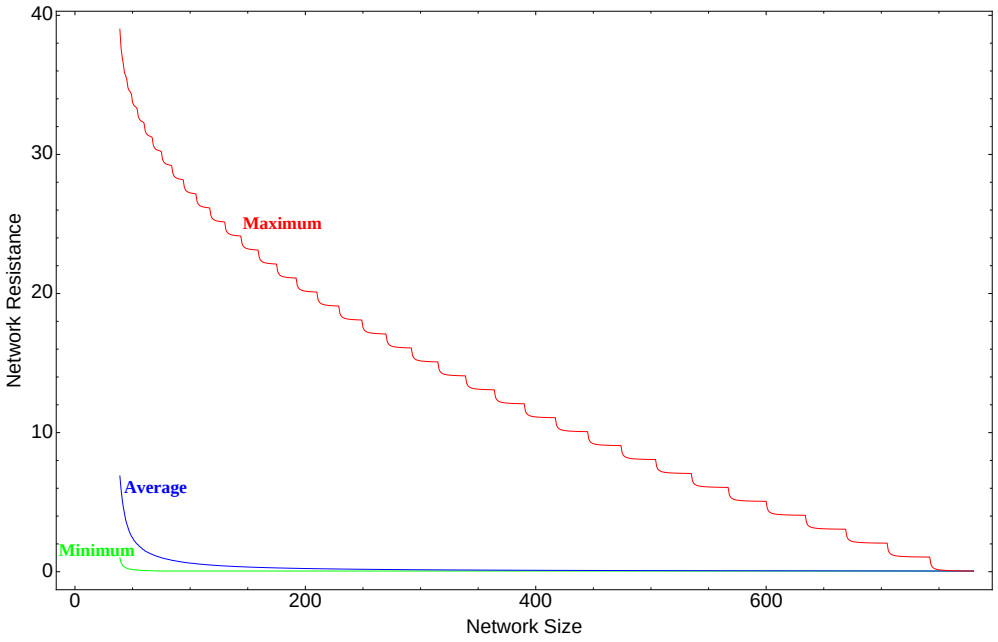


Figure 3.13: The largest, smallest, and average resistance of graphs of order $N = 40$ and size $39 \leq m \leq 780$. The minimum and maximum bounds are analytically computed. The average resistance of each graph is approximated by the mean of 10^4 random graphs of the respective size. The statistical average of a network of relatively small size is very close to the minimum resistance.

tance is

$$I_{S_{max}}(N,m) = \begin{cases} P+1 & \alpha = 1 \text{ or } \alpha = C-1 \\ \left\lceil \frac{P+1}{2} \right\rceil & 2 \leq \alpha \leq C-2 \end{cases} \tag{3.120}$$

where C, P and α are defined in equations (3.32), (3.33), *and* (3.34), *respectively.*

Proof. The order in which we place linear resistors in series has no effect on the total resistance of the circuit. We can serially place n_1 and n_2 resistors at each side of the almost complete subgraph, such that

$$n_1 + n_2 = P \quad \text{with } n_1 \geq 0 \text{ and } n_2 \geq 0. \tag{3.121}$$

The last equation has $P + 1$ solutions. When $a = 1$ or $a = C - 1$, then the almost complete graph is symmetric, which means that we count every non-isomorphic graph twice. Adjusting for this special case the number of non-isomorphic graphs, we get the desired result. □

Corollary 20. *The networks with the largest average distance have the largest resistance distance. The converse does not always hold.*

3.7 Average Clustering

One of the prominent features that distinguishes many large natural and engineered networks from random ones is the clustering among their individual units, and at the same time the small average distance among them [52, 70]. These properties were thought to be mutually exclusive, but in almost all real networks, having a small average distance does not greatly interfere with the presence of a large clustering coefficient, as shown in [70]. Despite the considerable amount of literature in the area of complex networks, there has been no study exploring the properties of a graph required in order for it to have both large average clustering and small average distance. In this section, we will find the topology of the networks with the largest average clustering, and show that at the same time their average distance is smaller or equal to the average distance of any other graph. We will also study their resilience to vertex and edge removal, and will solve the problem of finding the graph which has the largest average clustering coefficient, for given order N and size m. There is usually a unique graph with the largest clustering, which also has the smallest average distance. Finally, we devise a method to make these networks more robust with respect to vertex removal.

The local clustering coefficient of a vertex u is defined as the number of connections between vertices that are neighbors of u, divided by the total number of pairs of neighbors of u. In other words, it is the number of triangles in which u partici-

pates divided by the number of all possible triangles it could participate in, if all its neighbors were connected to each other. More formally, if d_u is the degree of a vertex u, and t_u is the number of edges among its neighbors, its clustering coefficient is

$$C(u) = \begin{cases} 0 & \text{if } d_u = 0 \\ 1 & \text{if } d_u = 1 \\ \frac{t_u}{\binom{d_u}{2}} & \text{if } d_u \geq 2. \end{cases} \tag{3.122}$$

An example is shown in Figure 3.14. The clustering coefficient of a vertex can only take values in the interval $[0, 1]$. Note that we deliberately choose to define the clustering coefficient of a vertex u with degree $d_u = 1$ as equal to 1. The graphs with the largest clustering under this convention may be different when we assume that vertices with degree 1 have zero clustering. The method to find the largest clustering graphs under the latter assumption is similar and will be described later.

The average clustering coefficient for a graph $\mathcal{G}$ is simply the average of all the local clustering coefficients in its vertex set $\mathcal{V}(\mathcal{G})$. A large average clustering coefficient is a proxy for increased robustness of the network, "local" structure, and increased connection density among vertices in a neighborhood [5]. If N is the order of the network, the average clustering coefficient is defined as

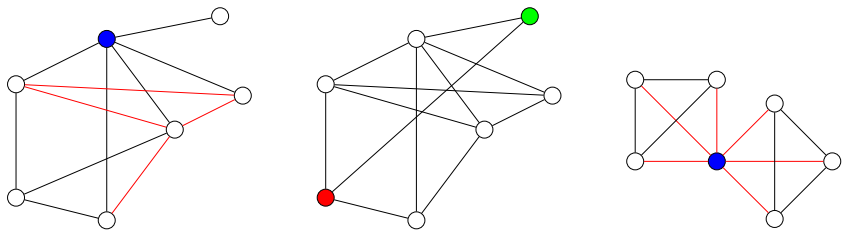
$$\bar{C}(\mathcal{G}) = \frac{1}{N} \sum_{u \in \mathcal{V}(\mathcal{G})} C(u). \tag{3.123}$$

Since we will only be comparing graphs with the same number of vertices and edges, in order to make our analysis easier, we will be considering the sum of the clustering

coefficients of all the vertices:

$$C_S(\mathcal{G}) = \sum_{u \in \mathcal{V}(\mathcal{G})} C(u). \tag{3.124}$$

Maximizing $C_S(\mathcal{G})$ is equivalent to maximizing $\bar{C}(\mathcal{G})$. If a network has both a high average clustering coefficient, and a small average shortest path length among its nodes, it is called a *"small world"* network [70]. This architecture is conjectured to have other desired properties, like enhanced signal propagation speed, synchronizability and computational power [65, 70]. As it turns out, the networks with the largest average clustering are "small world" networks, since they also have the smallest possible average distance.



(a) (b) (c)

Figure 3.14: **(a)** The clustering coefficient of a vertex is the number of connections between its neighbors, divided by all pairs of neighbors, whether they are connected by an edge or not. The clustering coefficient of the blue vertex u is $C(u) = 4/\binom{5}{2} = 0.4$. The average shortest path length is $\bar{L}_{\mathcal{G}} = 1.48$. **(b)** After rewiring one edge, and decreasing the distance between the red and the green vertex, the network has the smallest possible average distance, equal to 1.43. All vertex pairs now have at most a distance of 2. **(c)** The network with the largest possible average clustering and smallest average shortest path length. The blue vertex is the central vertex of the induced star subgraph.

3.7.1 Recursive Computation of Graph Clustering

Assume that we add one vertex u with degree d to a graph $\mathcal{G}(N, m)$, by connecting it to the vertices in a set $\mathcal{D}$ ($|\mathcal{D}| = d$) and the result is a new graph $\mathcal{G}'(N+1, m+d)$.

The difference of the sum of clustering coefficients of the two graphs will be

$$\Delta C_S(\mathcal{G}', \mathcal{G}) = C_S(\mathcal{G}') - C_S(\mathcal{G}) = C(u) + \Delta C_S(\mathcal{G}', \mathcal{G}, \mathcal{D}) \tag{3.125}$$

where $C(u)$ is the clustering coefficient of the new vertex, and

$$\Delta C_S(\mathcal{G}', \mathcal{G}, \mathcal{D}) = \sum_{v \in \mathcal{D}} (C'(v) - C(v)) \tag{3.126}$$

is the sum of the differences of the clustering coefficients of the vertices in $\mathcal{D}$, before and after they acquire their new edge. For all other vertices, the clustering coefficients remain unchanged.

Lemma 17. *Assume that we have a graph $\mathcal{G}(N, m)$ with sum of clustering coefficients equal to $C_S(\mathcal{G})$, and we add one more vertex u, with degree d. The difference in the clustering of the two graphs $\Delta C_S(\mathcal{G}', \mathcal{G}) = C(u) + \Delta C_S(\mathcal{G}', \mathcal{G}, \mathcal{D})$ can only be the largest possible if the vertices in $\mathcal{D}$ are part of a clique Q.*

Proof. If all vertices in $\mathcal{D}$ are part of a clique, they form an induced complete subgraph. The clustering coefficient of u will be the maximum possible ($C(u) = 1$), since all the possible connections among its neighbors will be present. Also, each of the d vertices of Q (with $|Q| = q \geq d$) that u is connected to, will increase the number of connections among their neighbors to the maximum extent (given the degree of u), and they will have $d - 1$ additional triangles each. The clustering coefficient of the rest of the $q - d$ vertices of the clique will not be affected. If, on the other hand, u forms connections with vertices that do not form a clique, its clustering coefficient will be $C(u) < 1$, and the vertices it is connected to will have less extra connections among their neighbors. □

Note that when $d < q$, the new clustering coefficient of the vertices that will be connected to u will be less than 1, possibly smaller than it was before connecting the new vertex. In order to minimize the effect of the missing triangles on the overall

clustering coefficient, we need to make sure that we connect u to those vertices of the clique with the largest degree (which may have additional connections outside the clique). Also, the size of the clique is an important factor. The larger the clique, the smaller the impact of the additional vertex, since the degrees of the connected vertices are larger, but on the other hand we may have more missing triangles in total.

Lemma 18. *Assume that we have a graph $\mathcal{G}$ that consists of a clique Q and two additional vertices A and B that each connects to a subset $\mathcal{D}_1$ and $\mathcal{D}_2$, with $d_1 = |\mathcal{D}_1| \leq |\mathcal{D}_2| = d_2$. The average clustering of $\mathcal{G}$ is maximized when $\mathcal{D}_1 \subseteq \mathcal{D}_2$.*

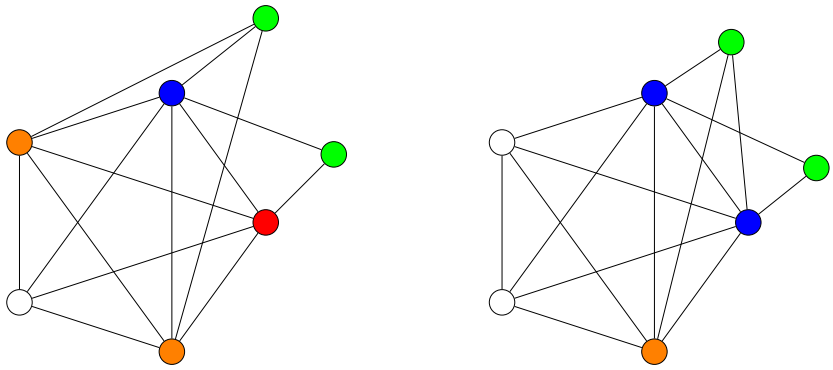
Proof. The number of missing triangles stays the same regardless of the specific vertices of the clique that A and B are connected to. So, the goal is to redistribute the missing triangles to vertices of the largest possible degree. When $\mathcal{D}_1 \subseteq \mathcal{D}_2$, the number of vertices of Q with clustering equal to 1 is maximized, and at the same time, the vertices of Q with the largest degree have the maximum number of missing triangles. □

The above lemma can be used recursively for any number of edges connecting to a clique. An example is shown in Figure 3.15.

Assume we have a graph $\mathcal{G}$ which belongs to the set $\mathcal{S}_{N,m}$ of all graphs with N vertices and m edges. We denote the clustering sum of the optimal graph with $C_S(N,m)$. Its clustering sum will be

$$\begin{aligned} C_S(N,m) &= \max_{\mathcal{G}\in\mathcal{S}_{N,m}} \left\{ \sum_{u\in\mathcal{V}(\mathcal{G})} C(u) \right\} \\ &= \max_{\mathcal{G}\in\mathcal{S}_{N,m}} \left\{ \left(\sum_{u\in\mathcal{V}(\mathcal{G})\setminus v} C(u) \right) + C(v) \right\} \end{aligned} \tag{3.127}$$

for any vertex $v \in \mathcal{V}(\mathcal{G})$. The reasoning behind the last equation is that if we pick any vertex from the graph, it should be "optimally" connected to a smaller graph



(a) (b)

Figure 3.15: An example of a graph with two vertices connected to a clique. The graph in (a) has a smaller sum of clustering coefficients C_S than the one in (b). By rewiring one edge in the graph on the left, we shift the missing triangles of one orange vertex to a blue one, with a higher degree. So the missing edges between a green and a white vertex have a smaller effect on the overall clustering.

that is itself optimal. This in turn means that we need to make sure that the chosen vertex has the largest possible clustering coefficient with regard to the vertices it is connected to, the rest of the graph needs to have the largest clustering coefficient, and the potential decrease of the clustering coefficient of the vertices after connecting the last one is minimal. This method provides an easy way to find the maximum clustering of a graph $\mathcal{G}$ of order N, by using a graph $\mathcal{G}_\prime$ of order $N-1$, and connecting one additional vertex to it. The above can be algebraically expressed by conditioning on the set $\mathcal{D}$ of vertices with which the additional new vertex v of degree $d = |\mathcal{D}|$ forms connections.

$$C_S(N,m) = \max_{\mathcal{D}\subseteq\mathcal{V}(\mathcal{G}_0)} \{C_S(N-1, m-d) + C(v) + \Delta C_S(N-1, m-d, \mathcal{D})\} \quad (3.128)$$

where $\Delta C_S(N-1, m-d, \mathcal{D})$ is the change in the clustering coefficient of the d vertices of the set $\mathcal{D}$ of the graph of order $N-1$ and size $m-d$ when we connect one vertex (vertex v) with degree d to them. The last equation shows that the graph with

the largest possible clustering coefficient can be found by connecting a vertex to an optimal graph with fewer vertices and making sure that the algebraic value of the change is as large as possible.

3.7.2 Clustering of Almost Complete Graphs

Lemma 19. *The total clustering for a type I almost complete graph $\mathcal{G}(N,m)$ is $C_S(\mathcal{G}) = N - \frac{\alpha(N-1-\alpha)}{\binom{N-1}{2}}$, where α is the degree of the peripheral vertex.*

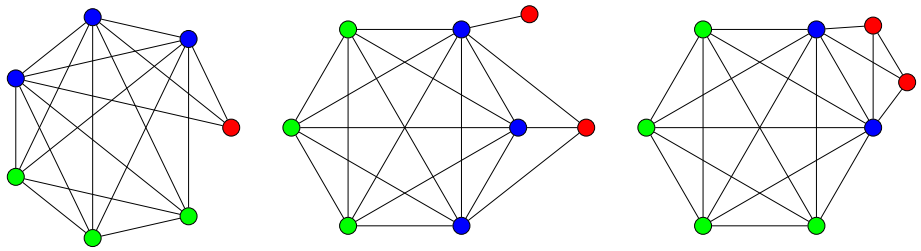
Proof. The peripheral vertex of the almost complete graph will have a clustering coefficient equal to 1, the α vertices with degree $N-1$ will have a clustering coefficient $C(u) = \frac{\binom{N-2}{2}+(\alpha-1)}{\binom{N-1}{2}}$, and the $N-1-\alpha$ vertices with degree $N-2$ will have a clustering coefficient equal to 1 (see Figure 3.16(a)). The sum of the clustering coefficients for this graph will be

$$\begin{aligned} C_S(\mathcal{G}) &= 1 + \alpha\frac{\binom{N-2}{2} + (\alpha-1)}{\binom{N-1}{2}} + (N-1-\alpha) \\ &= N + \frac{-\alpha\binom{N-1}{2} + \alpha\binom{N-2}{2} + \alpha(\alpha-1)}{\binom{N-1}{2}} \\ &= N + \frac{-\alpha(N-2) + \alpha(\alpha-1)}{\binom{N-1}{2}} \\ &= N - \frac{\alpha(N-1-\alpha)}{\binom{N-1}{2}}. \end{aligned} \tag{3.129}$$

□

The average clustering, as a function of α, is convex and symmetric around $\frac{N-1}{2}$. It decreases as α goes from 1 to $\lfloor\frac{N-1}{2}\rfloor$, and then it increases as α goes from $\lceil\frac{N-1}{2}\rceil$ to $N-1$.

Lemma 20. *A type I almost complete graph has larger clustering coefficient than any other nonisomorphic graph of the same order and size.*



(a) (b) (c)

Figure 3.16: Clustering for the almost complete graphs. **(a)** The peripheral vertex (red) has unit clustering since it is connected to vertices of a clique. The vertices that are not connected to it (green) also have clustering equal to 1, as members of the complete graph with no other edges. The vertices that are connected to the peripheral vertex (blue) have clustering less than 1, since the peripheral vertex is not connected to all of their neighbors. **(b)** A type II almost complete graph with the largest clustering, where the two peripheral vertices are not connected (type IIa). **(c)** A maximum clustering coefficient type II almost complete graph, with connected peripheral vertices (type IIb).

Proof. Assume a type I almost complete graph $\mathcal{F}$ that consists of a clique P of $N-1$ vertices, and one peripheral vertex with degree α that is connected to it. Further assume that $\mathcal{G}$ is a graph of the same order and size, whose largest clique Q consists of $N-d \leq N-2$ vertices (otherwise it would be isomorphic to $\mathcal{F}$). The $d \geq 2$ peripheral vertices form connections with the central vertices that are members of Q, and among each other. Define $\gamma = \binom{N}{2} - m$, which corresponds to the number of non-neighboring vertices in a graph. By assumption, $1 \leq \gamma \leq N-2$. It is easy to see that $d \leq \gamma$, since we have only γ edges missing, and for each peripheral vertex u, at least $m_u \geq 1$ edges between itself and Q have to be missing, otherwise Q is not the largest clique. We will add all of the clustering coefficients for all the vertices, and then show that the sum is always greater for $\mathcal{F}$. We note the following:

- The number of vertices with clustering coefficient equal to 1, is $1+\gamma$ for $\mathcal{F}$ and at most $d + \left\lfloor \frac{\gamma}{d} \right\rfloor$ for $\mathcal{G}$. The number of such vertices in $\mathcal{G}$ is smaller or equal to the respective number in $\mathcal{F}$, for all $2 \leq d \leq \gamma$. Conversely, $\mathcal{F}$ has exactly $N-1-\gamma$ vertices with clustering coefficient less than 1, and $\mathcal{G}$, has at least

$N - d - \left\lfloor \frac{\gamma}{d} \right\rfloor$ such vertices.

- All vertices that have a clustering coefficient smaller than 1 in $\mathcal{F}$ have degree $N-1$, the largest possible degree in a graph with N vertices.

We will find the sum of the clustering coefficients for both graphs in terms of the number of missing triangles. Then, we will show that $C_S(\mathcal{F}) > C_S(\mathcal{G})$, for any $\mathcal{G} \neq \mathcal{F}$. Since $\gamma = N - 1 - \alpha$, the sum of the clustering coefficients for $\mathcal{F}$, is

$$C_S(\mathcal{F}) = N - \frac{\gamma(N-1-\gamma)}{\binom{N-1}{2}}. \tag{3.130}$$

The sum of the clustering coefficients for $\mathcal{G}$ is

$$\begin{aligned} C_S(\mathcal{G}) &= N - \sum_{u=1}^{N} \frac{w_u}{\binom{d_u}{2}} \\ &\leq N - \frac{\sum_{u=1}^{N} w_u}{\binom{N-1}{2}} \\ &= N - \frac{s_g}{\binom{N-1}{2}} \end{aligned} \tag{3.131}$$

where w_u is the number of edges missing among the neighbors of vertex u and $s_g = \sum_{u=1}^{N} w_u$. We only need to prove that the missing triangles of $\mathcal{G}$ are more than those missing in $\mathcal{F}$, in other words that

$$\Delta = s_g - s_f = \sum_{u=1}^{N} w_u - \gamma(N-1-\gamma) \tag{3.132}$$

is greater than or equal to zero.

We will now find the minimum number of central vertices that are connected to at least two peripheral vertices in $\mathcal{G}$. This happens when $d-2$ peripheral vertices have the largest possible degree $N-d-1$, and the remaining two have the smallest possible degrees, and consequently the minimum number of clique vertices that they are both connected to. If we assume that b edges are *not* present among peripheral

vertices, we have a total of $t = d(N-d) - \gamma + b$ edges between peripheral vertices and central vertices. If we further assume that all but 2 peripheral vertices have the largest possible number of edges with the clique (which is $N-d-1$), the remaining peripheral vertices will have a degree sum of $r = t - (d-2)(N-d-1) = 2N - d - 2 - \gamma + b$. Since we only have $N-d$ vertices in Q, at least $r - (N-d) = N - 2 - \gamma + b$ of them will be connected to two peripheral vertices. The total number of triangles missing from the central vertices, because of edges missing between peripheral and central vertices, is

$$\begin{aligned} M_{C-P} &= \sum_{u=1}^{d} m_u(N-d-m_u) \\ &= (N-d)\left(\sum_{u=1}^{d} m_u\right) - \sum_{u=1}^{d} m_u^2. \end{aligned} \tag{3.133}$$

The number of triangles missing because of the absence of edges among peripheral vertices, as shown above, is at least

$$M_{P-P} \geq \sum_{e=1}^{b} (N-2-\gamma+b) = b(N-2-\gamma+b). \tag{3.134}$$

We are now ready to count the number of triangles missing from $\mathcal{G}$. We omit the triangles missing from the peripheral vertices, and assume that all peripheral vertices have clustering equal to 1 (otherwise the number of missing trinangles in $\mathcal{G}$ would further decrease). The total number of missing triangles in $\mathcal{G}$ is

$$\begin{aligned} s_g &\geq M_{C-P} + M_{P-P} \\ &\geq (N-d)\left(\sum_{u=1}^{d} m_u\right) - \sum_{u=1}^{d} m_u^2 + b(N-2-\gamma+b) \\ &= (N-d)(\gamma-b) - \sum_{u=1}^{d} m_u^2 + b(N-2-\gamma+b) \end{aligned} \tag{3.135}$$

with $\sum_{u=1}^{d} m_u = \gamma - b$, since γ edges in total are missing. In order to minimize s_g, we need to maximize the sum of squares $\sum_{u=1}^{d} m_u^2$, under the constraints $m_u \geq 1$ for

$u = 1, 2, \ldots d$, and $\sum_{u=1}^{d} m_u = \gamma - b$. The maximum is achieved when $m_u = 1$ for $u = 1, 2, \ldots d-1$, and $m_d = \gamma - b - (d-1)$. As a result,

$$s_g \geq (N-d)(\gamma - b) - \left((d-1) + (\gamma - b - d + 1)^2\right) + b(N - 2 - \gamma + b) \tag{3.136}$$

$$s_g \geq \gamma N - \gamma^2 + \gamma d - bd + d + \gamma b - 2\gamma - d^2. \tag{3.137}$$

The difference between s_g and s_f is

$$\begin{aligned} \Delta &= s_g - s_f \\ &= \gamma N - \gamma^2 + \gamma d - bd + d + \gamma b - 2\gamma - d^2 - \gamma N + \gamma^2 + \gamma \\ &\geq \gamma d - bd + d + \gamma b - \gamma - d^2 \\ &= (\gamma - d)(b + d - 1). \end{aligned} \tag{3.138}$$

By assumption, $d \geq 2$ and $b \geq 0$. The above product is always positive, when $d < \gamma$. If $d = \gamma$, we have many fully connected peripheral vertices, each of which connect to all but one vertex of Q. No two peripheral vertices are connected to the same central ones, for in this case Q would not be maximal. As a result, there are less than γ vertices in Q that have unit clustering, and more than $N - 1 - \gamma$ vertices in Q that are missing one or more triangles, so $s_g > s_f$. In every case, $C_S(\mathcal{F}) > C_S(\mathcal{G})$. □

There are two cases of type II almost complete graphs, depending on whether or not there is an edge between the two peripheral vertices. If there is no such edge (type IIa), the graph with the largest clustering takes the form of Figure 3.16(b). When the two peripheral vertices are connected (type IIb), the graph with the form shown in Figure 3.16(c) has the largest average clustering coefficient, as shown in the next lemma.

Lemma 21. *Assume a type II almost complete graph $\mathcal{G}(N, m)$ with two peripheral vertices u and v. If u and v are not connected, the average clustering coefficient is maximized when u and v have the smallest number of common neighbors. If u and v*

are connected, the graph with the largest clustering coefficient is the one where they have the largest number of common neighbors.

Proof. Let $c = m - \binom{N-2}{2}$ be the total number of edges connecting the two peripheral vertices to the rest of the graph. Without loss of generality, assume that u has smaller degree than v, $d(u) \leq d(v)$. Then, $c = d_u + d_v = 2a + b$, where a is the number of their common neighbors, and $b = d_v - d_u$ the neigbors of v that are not connected to u. The sum of clustering coefficients will be

$$\begin{aligned} C_s(\mathcal{G}) &= N - a \cdot \frac{1 + (N-2-a) + (N-2-a-b)}{\binom{N-1}{2}} - b \cdot \frac{N-2-a-b}{\binom{N-2}{2}} \\ &= N - \frac{c-b}{2} \cdot \frac{2N-3-c}{\binom{N-1}{2}} - b \cdot \frac{N-2-\frac{c-b}{2}-b}{\binom{N-2}{2}}. \end{aligned} \tag{3.139}$$

Since c is a constant, the sum of clustering coefficients is a function of b. By differentiating, we find that it is increasing with b for all $2 \leq c \leq N-2$ and $0 \leq b \leq c-2$.

Similarly, when there is an edge between u and v, $c = 2a + b + 1$, and the sum of vertex clustering coefficients is

$$\begin{aligned} C_s(\mathcal{G}) &= N - a \cdot \frac{(N-2-a) + (N-2-a-b)}{\binom{N-1}{2}} - b \cdot \frac{N-2-a-b}{\binom{N-2}{2}} + \frac{b}{\binom{a+b+1}{2}} \\ &= N - \frac{c-b-1}{2} \cdot \frac{2N-3-c}{\binom{N-1}{2}} - b \cdot \frac{N-2-\frac{b+c-1}{2}}{\binom{N-2}{2}} + \frac{b}{\binom{\frac{b+c+1}{2}}{2}}. \end{aligned} \tag{3.140}$$

The last equation is a decreasing function of b for all $2 \leq c \leq N-2$ and $0 \leq b \leq c-3$. From the above, we find that if a type II almost complete graph has maximum average clustering coefficient, there will be a peripheral vertex with degree 1 (type IIa), or both vertices will have the maximum number of common neighbors (type IIb). □

For a given size m of a type II almost complete graph, we need to decide which of the two variations has the largest clustering. The first two terms in both equations differ in the fact that c and $c-1$ edges connect a peripheral with a central vertex, respectively. In the IIb type, the third term in equation (3.140) becomes very large for

$b \geq 1$, compared to the other terms for any $N > 5$, and a necessary condition in order to have a type IIb almost complete graph is that $b = 0$ (the two peripheral vertices have exactly the same neighbors), which means that c has to be odd, $c = 2a + 1$. Then, a simple comparison of the two equations shows that if

$$c \geq \frac{3N - 1 + \sqrt{2\left(N^3 - 4N^2 - N + 12\right)}}{N + 1} \tag{3.141}$$

then the type IIa has larger average clustering than type IIb. The above number scales proportionally to the square root of the order N.

Lemma 22. *A type II almost complete graph has larger clustering than any other graph of the same order and size.*

Proof. We will use induction. The claim is true for graphs of order $N = 4$, shown by exhaustive enumeration of all the graphs with 4 vertices. Now, we will assume that it is true for all graphs of order up to $N - 1$, and will show that this is still true for a graph of order N. The optimal graph will be found by using equation (3.128). The connected graph of order $N - 1$ with the largest clustering coefficient is either a type I or type II almost complete graph for all possible degrees d of the additional vertex w, because $1 \leq d \leq N - 1$, and by assumption $\binom{N-2}{2} + 2 \leq m \leq \binom{N-1}{2}$, so

$$\binom{N-3}{2} \leq m - d \leq \binom{N-1}{2} - 1. \tag{3.142}$$

If it is a type I almost complete graph, addition of one vertex will transform it into an almost complete graph of type II, regardless of its degree. Consequently, the lemma holds in this case. Now assume that the existing graph is a type II almost complete graph. Assume that the two already existing peripheral vertices u and v have α and β edges with vertices of the largest clique. If $\alpha + \beta + d = N - 1$, then it is easy to see that the optimal graph consists of a full graph of order $N - 2$, and two vertices of degree 1 that connect to it. For every other value of $\alpha + \beta + d$, since the initial

graph is type II almost complete, $d \geq 2$, and w has at least one neighbor in the clique. Considering u, v, and w in pairs, we can show that a graph with three peripheral vertices cannot be optimal, in other words, not all three can have less than $N-3$ edges to the clique of order $N-3$. The reason is that according to equation (3.127), any choice of a single vertex from the graph should yield the same result, in terms of maximizing the sum of clustering coefficients. By the induction hypothesis, in each pair, one of the peripheral vertices should have one connection to the clique, (not possible under the constraint $\alpha + \beta + d > N-1$) or all of them should be connected to each other and to the same vertices of the clique (also impossible for the same reason). So, in every case, a type II almost complete graph will be optimal. □

3.7.3 Graphs with the Largest Clustering

In this subsection, we will combine the previous results to show the form of the graphs with the largest clustering for a graph of arbitrary order and size.

Lemma 23. *The largest clustering graph with N vertices and $0 \leq m \leq N-2$ edges consists of complete components of 2 or 3 vertices each.*

Proof. For m=0, we have no edges and the clustering coefficient is equal to zero. If $m > 0$, we first connect pairs of vertices, until all of them have degree 1. If we have any edges left, we start forming triangles, by trying to keep the vertices that do not have any edges at a minimum. In a triangle, we have the same number of vertices and edges, and since $m \leq N-2$, the number of edges is not enough to connect all the vertices in triangles. The above procedure will guarantee that the disconnected graph will have a sum of clustering coefficients equal to 1. □

Lemma 24. *The tree that has the largest clustering (according to the convention that a vertex with degree 1 has clustering 1) is the star graph.*

Proof. Since there are no cycles in a tree (which also means no triangles), a vertex with degree larger than 1 will have a clustering coefficient equal to zero. By minimizing

the number of such vertices (one vertex is the minimum number since the graph needs to be connected), we achieve the largest clustering for the star graph. □

An important point to note is that the star graph is a graph with one cut vertex, connected to several complete graphs of order 2.

Theorem 13. *The graph with N vertices and $N \leq m \leq \binom{N-2}{2} + 1$ edges that has the largest possible clustering coefficient consists of one cut vertex that is shared by complete or almost complete subgraphs.*

Proof. We will use induction on the number of vertices to find the optimal graph for $N \leq m \leq \binom{N-2}{2} + 1$. For N=4, the statement can be found to be true, by evaluating all the possible graphs (for computational considerations see [32]). Assume that the optimal graph for every number of vertices up to $N-1$ and for the respective range of sizes, has the form mentioned above. The graph with N vertices and m edges will be found by connecting a new vertex of degree d to an optimal graph of order $N-1$ and size $m-d$. The resulting graph will have the maximum possible clustering of the new vertex (equal to 1), the maximum clustering of the rest of the graph it is attached to, and the minimal possible decrease in clustering for the vertices it is connected to.

If d is larger than the order of the largest subgraph in $\mathcal{G}(N-1, m-d)$, then it will be connected to at least two smaller complete or almost complete subgraphs. Now consider the subgraph that consists of these subgraphs plus the added vertex of degree d. It has $P < N$ vertices and $R < \binom{P-2}{2} + 2$ edges. In addition, it has a cut vertex, which connects its two or more components. This is a subgraph whose form is *not* optimal, according to the induction hypothesis, meaning that there is a smaller degree d for the vertex u, for which equation (3.128) gave a larger clustering coefficient for order P and size R. So we only need to consider vertices with degree d less than the size of the largest "module" if $\mathcal{G}(N-1, m-d)$ is already connected. Moreover, we only need to try to connect it to one clique. The rest of the cliques will not change, so we focus on the clique where the new vertex with the new edges

is added, and prove that it will still have the same form. If it is a complete graph, then after adding the new vertex u, it will have one peripheral vertex, which makes the claim hold (the complete graph will now be a type II almost complete graph). If it is a type I almost complete subgraph, after the addition of the new vertex, it will become a type II almost complete subgraph, as shown above. If the subgraph is a type II almost complete subgraph, then connecting u to it cannot yield an optimal subgraph, since by increasing the degree of u to the size of this subgraph, we can make a graph with larger clustering, as shown in the lemma about the optimal form of type II almost complete graphs. In every case, the new graph will have the form described in the theorem. □

The above arguments show the form that the optimal graph needs to have, but not the exact arrangement of edges among its vertices. Using equation (3.128), we can find the optimal graph in polynomial time, and we only need to consider a small range of different degrees for the added vertex.

Corollary 21. *The graph $\mathcal{G}(N, m)$ with the largest possible clustering coefficient has one of the following forms, depending on its size m:*

- $0 \leq m \leq N-2$: Disconnected graph, consisting of complete components with 2 or 3 vertices each.
- $N-1 \leq m \leq \binom{N-2}{2} + 1$: Complete or almost complete subgraphs that share one vertex.
- $\binom{N-2}{2} + 2 \leq m \leq \binom{N-1}{2}$: Type II almost complete graph.
- $\binom{N-1}{2} + 1 \leq m \leq \binom{N}{2} - 1$: Type I almost complete graph.
- $m = \binom{N}{2}$: Complete graph, with average clustering $\bar{C} \equiv 1$.

3.7.4 Properties of the Graphs with the Largest Average Clustering Coefficient

1. The structure of real large-scale networks have locally the structure of the networks with the largest average clustering, and this pattern has been observed in many different types of networks, from social to biological [27, 52]. The subnetworks that comprise the larger ones have a few very well connected central vertices, that are part of many complete or almost complete subgraphs. This pattern also helps define a hierarchical structure in natural and engineered networks, if each peripheral vertex that belongs to one clique is also the central vertex to a different subnetwork. This graph form can also be used to facilitate modeling and characterization of large-scale networks [30]. Although networks with the largest clustering coefficient are unique, a large clustering coefficient does not necessarily imply any specific network structure or function, since there are ways to construct random networks with relatively high average clustering coefficient [49].

2. The form of the optimal graphs is recursive. Every combination of clusters (modules) that are connected through a cut vertex themselves form an optimal graph of the respective order and size. This suggests an alternative way to generate an optimal graph, where it can be found as a combination of smaller complete or almost-complete subgraphs. We can find the optimal form recursively with this procedure, too. Since there is always a cut vertex, we can break the network in two smaller parts, that do not affect each other's average clustering coefficient, and optimize them independently.

3. When a new vertex is attached to a graph with the process described above, it is *always* attached to vertices that are members of the same clique. This property can make computation easier, since we only need to consider cliques of size equal or greater to the degree of the candidate additional vertex, according

to equation (3.128).

4. No graph with the largest average clustering has any chordless cycles. This is because a cycle of length more than 3 could be changed to a triangle, therefore increasing the clustering coefficient.

5. No graph with the largest average clustering has any induced bipartite subgraphs, since bipartite subgraphs have average clustering coefficient equal to zero.

6. Every cut set is a complete graph of order 1 or more. In addition, if there exists a cut vertex, it also belongs to all larger cut sets. The structure of the graphs with the largest clustering is very modular, with subgraphs that serve as modules, connected through the cut vertex. Also, graphs that locally have the structure with the largest clustering coefficient have also very high average clustering. Many natural networks also seem to have evolved to have modular structure, which makes the systems more stable and robust [66].

7. For all connected optimal graphs, there is at least one vertex that is connected to every other vertex in the graph. If the graph is not almost complete, then the central vertex is unique. Otherwise, we may have more than one such vertex. This architecture is locally "robust yet fragile" [20], since the structure of the function of the network is unlikely to be affected too much if one of the vertices in one of the almost complete subgraphs is dysfunctional or removed. On the other hand, if the central vertex is removed, the network becomes disconnected. For networks that locally have the form of the networks with the largest average clustering, with many nodes of large degree, the same principle holds. The highly connected nodes are far more likely to affect the network than the nodes of smaller degrees, and this observation is quite independent of the nature or function of the network [53, 69].

8. The networks with the largest clustering coefficient also have the smallest average distance among the various vertices. For every pair of vertices that are not connected, there is a path of length 2 that connects them. These networks are classified as "small world", because they have both high average clustering and small average shortest paths, as mentioned previously.

9. From a computational standpoint, in order to find the optimal graph $\mathcal{G}(N, m)$, we first need to find the optimal graphs of order $N-1$, for all sizes, which in turn means that we need to compute *all* such graphs of *all orders up to* $N-1$. The complexity of this procedure is polynomial, $\mathcal{O}(N^5)$, since we need to find the optimal graphs for all graphs of order up to N, each of which has $\mathcal{O}(N^2)$ edges, and each time we need to add one vertex, trying various degrees $d = \mathcal{O}(N)$, to all appropriate $\mathcal{O}(N)$ cliques. In order to avoid computing recursively the same optimal graphs, we can start from the smallest possible optimal graph (for $N = 3$), and build our way to the desired N, while storing all the optimal graphs to the memory. If we had to perform an exhaustive search, the complexity would be prohibitively large, $\mathcal{O}(2^{\frac{N^2}{2}})$.

3.7.5 Fast Generation of Graphs with Small Distance and Large Average Clustering

In the previous sections, we saw that the form of the graphs with the smallest distance and largest clustering take the form of complete or almost complete subgraphs that are connected through one vertex. In order to generate very large networks that have this form, we can resort to a much simpler algorithm that has almost constant complexity, and can generate networks with arbitrarily many vertices, with the minimal average distance and very close to optimal clustering coefficient. Given the order and size of the graph, we find the largest complete subgraph, which leaves enough edges for the rest of the network to be connected. Then we subtract the number of vertices and

edges used, and repeat the process until all the vertices and edges have been used. If at any point during the process, it is found that we cannot form a subgraph with the number of vertices and edges, we backtrack and reduce the order of the cluster in the previous step (see [32]).

3.7.6 Resilience to Vertex or Edge Removal

The small-world networks studied here are very robust to edge removal. Since almost every vertex is part of a complete graph, we need to remove at least as many edges as the order of the smallest clique it belongs to in order to render the network disconnected. Even in that case, the number of vertices disconnected is at most equal to the number of edges deleted.

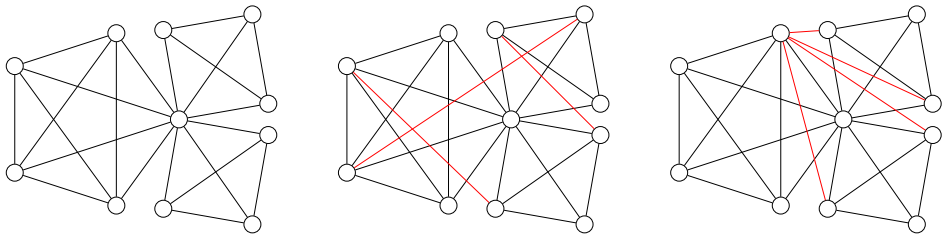


Figure 3.17: **(a)** The graph with the largest clustering coefficient of order 11 and size 22 consists of three complete subgraphs. Removal of the central vertex will render the network disconnected. **(b)** Addition of a few new edges among the different subgraphs creates new alternative communication paths among the various vertices of the network. As a result, the minimum cut set of the network is increased. **(c)** Adding new edges that connect one edge with vertices in the other subgraphs essentially creates a new network with more than one central vertex.

The situation is different for vertex removal. We immediately note that unless we have enough edges to build an almost complete graph, the network becomes disconnected if we remove the central vertex, and the number of disconnected components is the number of modules in the initial network. Real world networks on the other hand, rarely have an articulation point. The robustness of a network to vertex failure

is determined by its smallest cut set, and in this type of network, it consists of a single vertex (the central vertex). Removal of the central vertex will render the network disconnected. Depending on the application, we may be able to add new edges, which will increase the network's robustness to vertex failure. There are many ways to add the new edges, all of which result in reduced clustering. One of them is to distribute the new edges among vertices of the various modules, as shown in Figure 3.17(b). This method ensures that if the central vertex is removed, there are still communication channels among the different subsystems of the network. Another way is shown in figure 3.17(c), where one or more of the vertices forms new connections with the vertices of all the other subgraphs, which in essence increases the number of 'central' vertices. Rewiring or adding new edges will have the least effect when they connect vertices with a large degree.

3.7.7 Alternative Definition for Vertices with Degree One

A common alternative convention for the vertices with degree 1 is to define their clustering coefficient as equal to zero. The process of finding the graphs with the largest clustering under this new convention is very similar to the previous case. The biggest difference is the form of graphs with a small number of edges, when they consist of at least one subgraph that is a tree.

For $m < N - 1$, the graph is disconnected, and the optimal form consists of a group of disconnected triangles. When $m = N - 1$, the only form a connected graph can take is a tree, in which case, any arrangement of the vertices will yield a clustering coefficient of zero, since there can be no triangles. As m increases, we are able to form the first triangles, and the vertices that have nonzero clustering are part of a triangle, and have one more edge to the rest of the graph, keeping their degree low. As the number of edges increases further, and every edge is part of at least one triangle, the form of the optimal graphs resembles the form under the previous case, with only

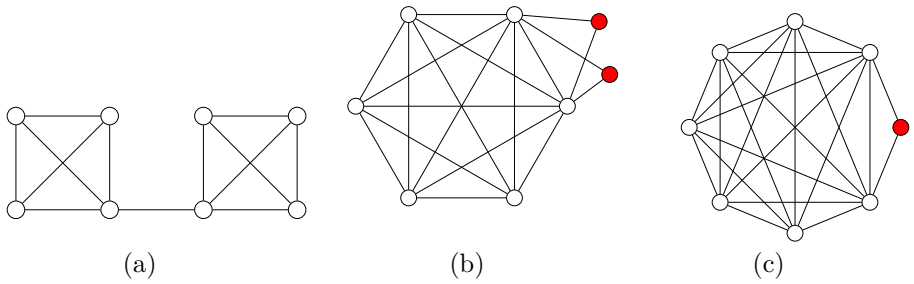


Figure 3.18: Graphs with the largest clustering coefficient, defining the clustering coefficient of vertices with degree $d = 1$ as equal to zero. **(a)** The optimal graph for $\mathcal{G}(8, 13)$ has two cut vertices, connected through a bridge. **(b)** The smallest type II almost complete subgraph consists of a clique with $N - 2$ vertices, and both peripheral vertices have degree at 2. The peripheral vertices are shown in red. **(c)** The smallest type I almost complete graph under the new convention, where the peripheral vertex has degree 2.

one difference, and after which point, no vertex has degree 1. Some graphs, instead of having a unique cut vertex, have two cut vertices that are connected through one bridge edge, as shown in Figure 3.18(a). Under the initial convention, the same graph would consist of the same two modules, plus one vertex with degree 1, connected to the cut vertex. The last difference is that the definition of the almost complete graphs has to be changed so that the peripheral vertices have a degree of at least 2 (Figures 3.18(b) and 3.18(c)). The process of finding the optimal graphs remains otherwise the same, and an example for a graph of order 10 is shown in Figure 3.20. Comparing figures 3.19 and 3.20, we immediately see the similarity of the optimal graphs in both cases.

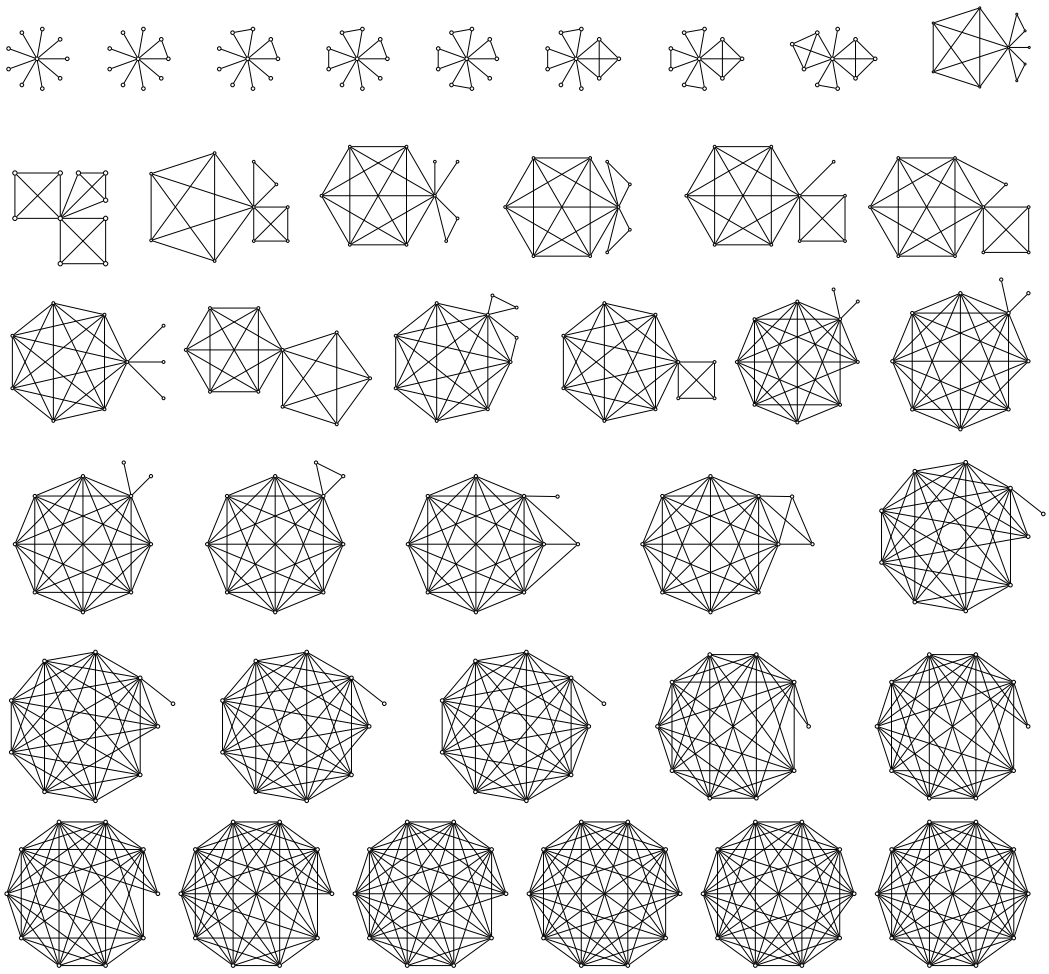


Figure 3.19: Graphs of order 10 and size $9 \leq m \leq 45$ with the largest clustering coefficient, assuming that vertices with degree 1 have clustering coefficient equal to 1.

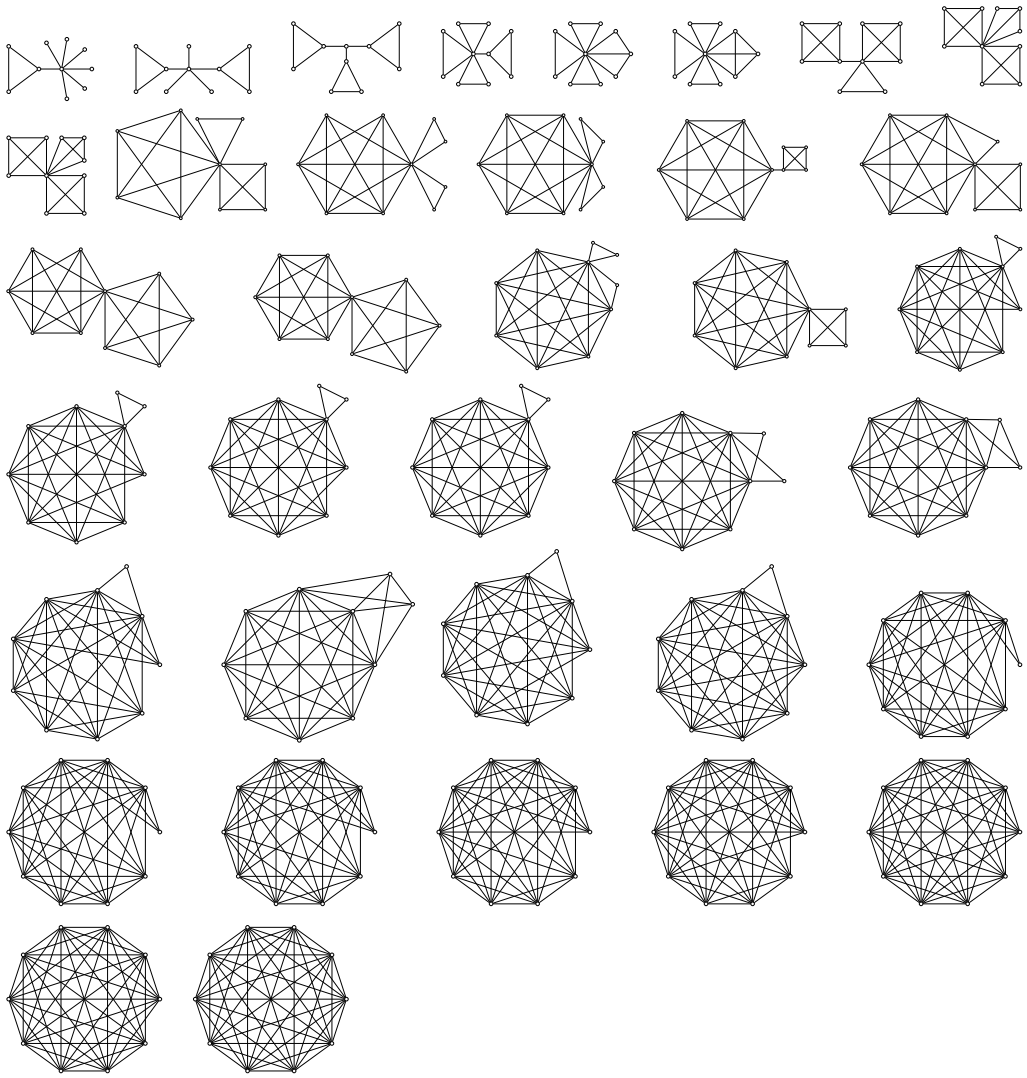


Figure 3.20: Graphs of order 10 and size $10 \leq m \leq 45$ with the largest average clustering coefficient, assuming that vertices with degree 1 have a clustering coefficient equal to zero.

3.8 Relationships Among Graphs with Various Extremal Properties

From the analysis of the various graph properties, we can find relations among the respective classes of graphs. This is important in order to design networks that achieve more than one extrema simultaneously. There are two main relations. The first one includes the set of networks with the minimum average clustering ($\mathcal{C}_{min}$), minimum radius ($\mathcal{R}_{min}$), minimum distance ($\mathcal{D}_{min}$), maximum efficiency ($\mathcal{F}_{max}$), minimum betweenness centrality ($\mathcal{B}_{min}$), and minimum diameter ($\mathcal{T}_{min}$):

$$\mathcal{C}_{min} \subseteq \mathcal{R}_{min} \subseteq \mathcal{D}_{min} = \mathcal{B}_{min} = \mathcal{F}_{max} = \mathcal{T}_{min}. \tag{3.143}$$

We can clearly see that networks with the minimum average clustering also have, by construction, minimum radius (equal to 1), average distance, and diameter(equal to 2). They also have the minimum possible vertex and betweenness centrality, as well as the largest possible efficiency. This is the only class of networks in this equation that are unique. Networks with minimum radius are not unique, since they only require one vertex to have full degree, regardless of the connectivity pattern of all other vertices. The network types with minimum average distance, diameter, betweenness centrality and maximum efficiency are less constrained: all vertices are either connected or have a common neighbor.

The second relation includes networks with the maximum average distance ($\mathcal{D}_{max}$), minimum efficiency ($\mathcal{F}_{min}$), maximum vertex and edge betweenness centrality ($\mathcal{B}_{max}$), and maximum diameter ($\mathcal{T}_{max}$):

$$\mathcal{D}_{max} = \mathcal{F}_{min} = \mathcal{B}_{max} \subseteq \mathcal{T}_{max}. \tag{3.144}$$

As mentioned in the previous sections, a network with maximum average distance also has minimum efficiency and maximum average vertex and edge betweenness centrality.

The networks with maximum diameter are a superset of the last class of networks. The networks with maximum radius are not part of any of the set equations above. The same is true for networks with minimum or maximum resistance distance. This is because of the particular requirements of each structure. The networks with maximum diameter have a structure that has a different effect on the other properties. Networks with minimum resistance distance have constraints on two vertices only, such that the connectivity of the rest of the network can be arbitrary. Finally, networks with maximum resistance are a superset of networks with the maximum distance (and subsets of networks with maximum diameter) only when the size of the network is not too large, in other words when they consist of an almost complete subgraph and one or two path subgraphs.

3.9 Variance of Various Properties for Random Networks

In order to evaluate the structure of a graph which achieves a minimum or maximum of a given property, we need to find not only the expected value of this property for a random network, but also its standard deviation. This way, we can measure how much a given structure deviates from the mean, and estimate if there are many other networks with similar structure. For a fixed order and size, a large standard deviation (and consequently, variance) for a given property would reveal that the respective random networks are more diverse. In contrast, a small standard deviation means that the property in question has the same value for almost all random graphs with the same number of nodes and edges. Small variance would make the design of networks with properties that are further away from the mean harder, because it would indicate that even small changes in their structure would significantly change their properties. The standard deviation of the average distance, radius, diameter, and resistance of a network is shown in Figure 3.21. For the average distance and

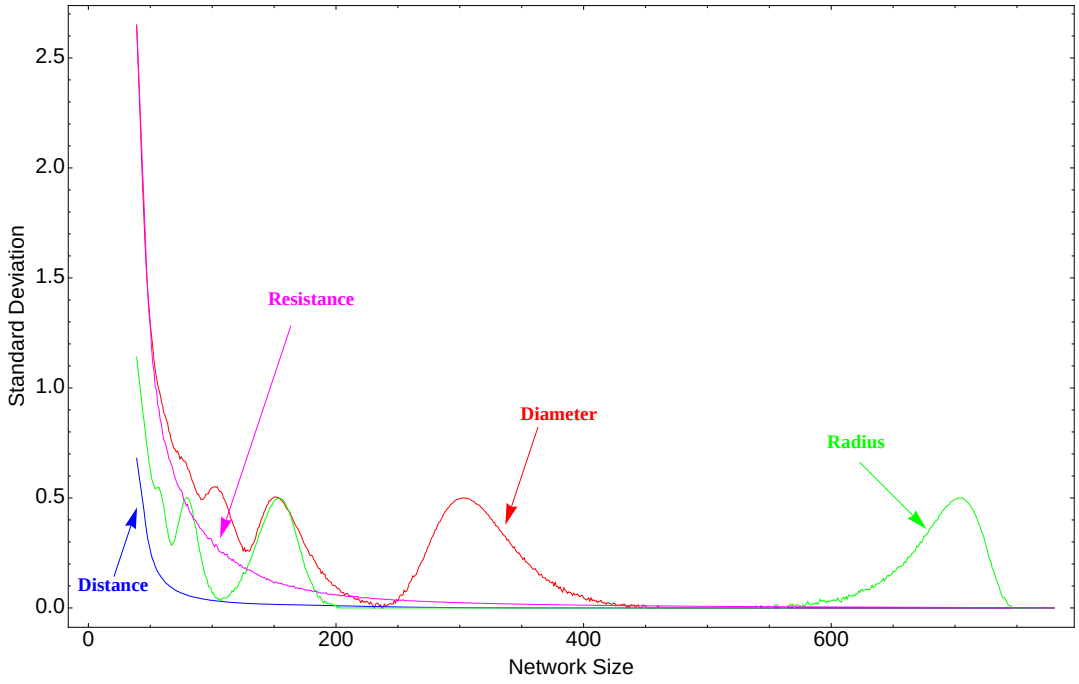


Figure 3.21: Standard deviation of various properties of a network of fixed order as a function of its size. The properties shown here are the average shortest path length, radius, diameter and resistance distance. The plots are for a network of order $N = 40$ and size $39 \leq m \leq 780$. The standard deviation of the average distance and resistance are monotonous and decreasing as the size of the graph increases. The standard deviation of the radius and diameter of the network has several local maxima. The graph sizes where it increases and then goes back to zero are when the average value changes.

resistance, it is a strictly decreasing function of its size, as one would expect, with distance always having variance smaller or equal to every other property shown.

The shape of the curves for both the radius and diameter closely resembles a combination of a strictly decreasing function that quickly drops to zero (like the one for the average distance), and a series of smooth "humps". This is consistent with the shape of the expected value of the radius and diameter for random graphs, as shown in Figures 3.7 and 3.9. The graph sizes for which these local maxima are exactly the graph sizes for which there is a transition from one value to another, between the different "steps" in the respective plots of the averages, and when the mean values do not change, the standard deviation is very close to zero.

3.10 Sensitivity to Rewiring

In most cases when designing a network, there are many constraints that need to be satisfied, and the properties discussed so far are only proxies to determining other desirable qualities of a network. Also, it is reasonable to expect that our networks may not be allowed to have the extremal values of the properties described in the previous sections, especially given that sometimes there are conflicting requirements for the network function. Given these considerations, we are interested in knowing how robust these structures are, in other words, how sensitive these properties are to changes in the interconnection patterns.

3.10.1 Average Distance, Radius, Diameter, and Resistance

We start with two networks with the extremal properties (minimum and maximum), and keep rewiring one of the edges, making sure that the network remains connected. At every step, we measure the properties of each network instance, and monitor its evolution as we introduce more and more randomness in their structure. Eventually, both will resemble a random graph, with the average distance, betweenness centrality, efficiency, radius, diameter, and resistance being close to the statistical average. The question is how fast this state is reached. If the change is very fast, this means that a few changes in the structure are able to negate the advantages that an extremal graph is able to provide. However, if a particular property does not change much after several rewirings, we can afford to build networks with many other desirable properties without the need to follow exactly the specific topology for every given property.

We have tested the change of average distance, radius, diameter and resistance for networks of small order. Refering to Figure 3.22, we find that the average distance of networks with large average shortest path length decreases very quickly after a few rewirings. Convergence to the statistical average is much slower for networks that

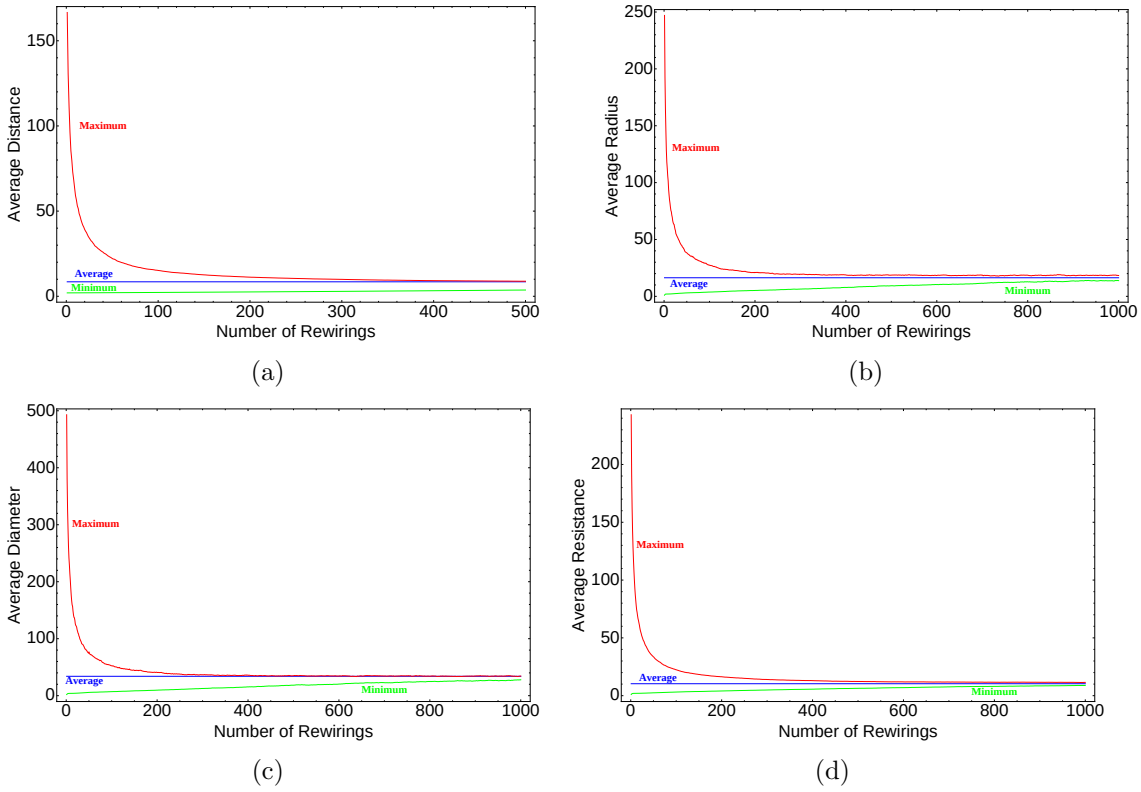


Figure 3.22: Evolution of the average distance, radius, diameter and resistance of networks after successive random rewirings. Each curve is the average of 100 experiments, rewiring networks of the same size and order $N = 500$. The curves for the average shortest path length correspond to a network of size $m = 600$. The curves for the evolution of radius, diameter and resistance correspond to networks of size $m = 525$.

start with the smallest average distance. The number of rewirings required to get close to the statistical average is an increasing function of the order of the network when the edge density is constant, as one would expect, since there are more edges that need to be rewired in order to make a difference. The same can be said for the evolution of the network radius, diameter, and resistance.

The general conclusion is that networks with the smallest average distance, radius and diameter and resistance are much more robust to changes in their topology because they rely on a global pattern of interconnections, and each edge has a small role in ensuring that property, so its conservation is diffused among many edges. On the contrary, networks with maximum distance, radius, diameter, and resistance are

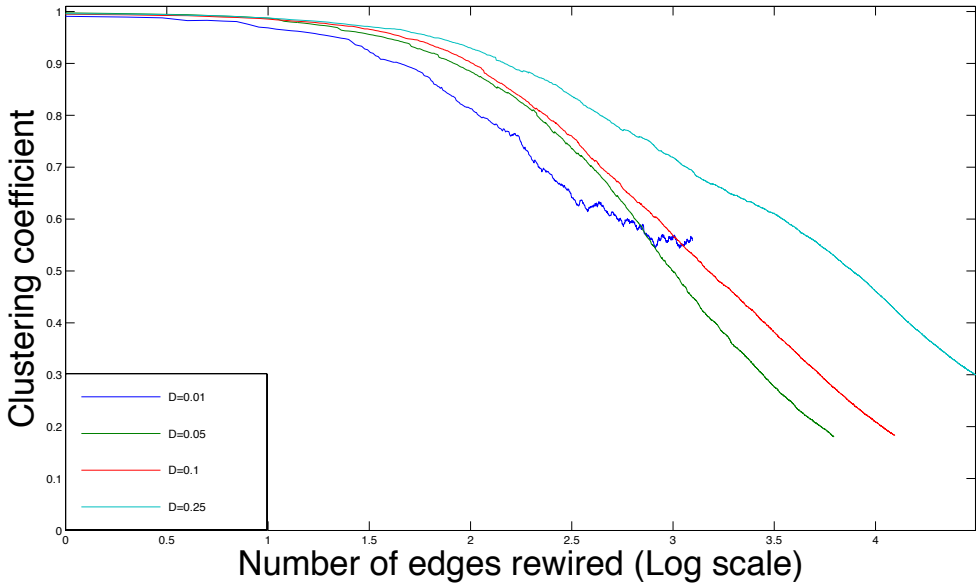


Figure 3.23: Change of the average clustering of a network of order N=500 and various edge densities D . At every step one edge is rewired and the new clustering coefficient is measured. We repeat the process for a number of steps equal to the number of edges in the graph. The rate of decrease is very small at first, but increases as we introduce more randomness in the structure. Larger networks are more robust to rewiring a fixed number of edges.

very sensitive to change, because most vertex connections are prohibited, in the sense that if currently non-neighboring vertices are connected, there is a dramatic trend towards the statistical average.

3.10.2 Clustering Coefficient

In Figure 3.23, we show the evolution of the average clustering coefficient after rewiring a percentage of the edges, for a typical large network, and for different edge densities. If we rewire a small number of edges, this has no notable effect on the average clustering coefficient, since most of the vertices have only a small amount of neighbors that are not in their previous clique. The effect of rewirings depends on the edge density of the network, with more densely connected networks being more

robust, since the relative amount of randomness introduced in the structure becomes smaller as the size of the network increases. This is a similar situation to that found in [70], with the only difference being that the distance is very small from the start, and quite robust to rewirings, increasing less than 0.1% even after we rewire a number of edges equal to the size of the network.

3.11 Conclusions

This chapter focused on structural properties of graphs. We found tight bounds on specific graph properties, and how they relate to the respective expected values for random graphs. We also showed the structure of networks that achieve these bounds, along with when these structures are unique. These properties include the average distance of a graph, the vertex betweenness centrality, edge betweenness centrality, efficiency, radius, diameter, resistance distance and average clustering. Finally, we explored how sensitive these properties are with regards to rewiring a fraction of the total number of edges for each of these structures.

Chapter 4

Quantification and Minimization of Crosstalk Sensitivity in Networks

Crosstalk is defined as the set of unwanted interactions among the different entities of a network. It is present in various degrees in every system where information is transmitted through a means that is accessible by all its individual constituents. Using concepts from graph theory, we introduce a quantifiable measure for sensitivity to crosstalk, and analytically derive the structure of the networks in which it is minimized. It is shown that networks with an inhomogeneous degree distribution are more robust to crosstalk than corresponding homogeneous networks. We provide a method to construct the graph with the minimum possible sensitivity to crosstalk, given its order and size. Finally, for networks with a fixed degree sequence, we present an algorithm to find the optimal interconnection structure among their vertices.

4.1 Introduction

Crosstalk affects the function of many complex engineering systems. In microelectronic circuits that operate in higher frequencies, it is increasingly hard to ensure proper function without isolating the various components from electromagnetic interference. In wireless communications, we need to make sure that the input and output signal frequency spectrums are relatively narrow to avoid crosstalk with other

communication systems. In chemical and biomolecular systems, different molecules interact with each other in a solution without explicit mechanisms that bring them in contact, creating errors that disrupt the function of the network and introduce errors [61, 64]. In social networks, crosstalk which appears in the form of "weak ties" and can be responsible for many macroscopic phenomena, such as diffusion, social mobility, political organization, and social cohesion in general [29].

As a result, there is a potentially large number of spurious interactions, despite which all natural systems seem to be able to function without problems. The plausible amount of unwanted crosstalk interactions increases as a function of the number of the individual elements in the network, so there is the need to design systems that are optimized to minimize or at least decrease the amount of spurious interactions, in order to keep the network functional. In biological networks, this need is more pronounced since signaling is implemented through variations of molecular concentrations.

Crosstalk does not seem to be a problem in the cell, or any other natural biological system, despite the large variety of molecules, and the intricate patterns of interactions. However, in biological design, even the simplest systems suffer from crosstalk between different elements, notwithstanding the small complexity of such artificial systems [57, 58]. If there are molecules that randomly interact with each other without performing any function, the effective concentration of each molecule which is available to participate in any given pathway will tend to decrease, and therefore the efficiency of the circuit will diminish.

In this chapter, we find the structure of graphs with the smallest possible sensitivity to crosstalk, assuming that the crosstalk affinity of each vertex is either specific to each unit in the network, or is a function of its degree.

4.2 Model

4.2.1 General Considerations

Assume that we have a weighted graph that represents a network of interactions. We define crosstalk as additional unmodeled interactions among the vertices of the network. These edges are randomly distributed across the network, and typically have smaller weights than the original ones. We assume that there cannot be any unwanted interactions among vertices that are already connected in the original network. The crosstalk matrix, denoted by R^{xt}, is the weight matrix of all the crosstalk interactions. If there is crosstalk between vertices u and v, the matrix element $w^{xt}_{u,v}$ will contain the intensity of that interaction. We will assume that the crosstalk affinity between any pair of vertices $(u, v) \notin \mathcal{E}(\mathcal{G})$, is bounded:

$$||R^{xt}||_\infty = \max_{(u,v)\notin\mathcal{E}} w^{xt}_{u,v} \leq \epsilon. \tag{4.1}$$

Definition 4. *Given a local measure of the crosstalk interaction $w^{xt}_{u,v} = g(u, v)$ among two vertices u and v, the overall sensitivity to crosstalk of a network of interactions is defined as the sum of the intensities of all the spurious interactions among individual units that are not connected:*

$$K(\mathcal{G}) = \sum_{(u,v)\notin\mathcal{E}} w^{xt}_{u,v}. \tag{4.2}$$

When the network in question is obvious from the context, we will refer to its sensitivity to crosstalk simply as K. The order of the graph is finite, so the overall crosstalk sensitivity is bounded.

$$||R^{xt}||_1 = \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} w^{xt}_{u,v} \leq \left(\binom{N}{2} - m\right)\epsilon = \mathcal{M}. \tag{4.3}$$

We will call $\mathcal{M} = \mathcal{M}(N, m, \epsilon)$ the *maximum possible crosstalk intensity* of the net-

work. It is only a function of the size and order of the network along with the maximum possible crosstalk intensity of any pair of vertices, which depends on the network and is independent of its specific structure. The matrix R^{xt} is constant, assuming that each crosstalk interaction represents the average over a time period.

Definition 5. *We say that a graph $\mathcal{G}^* = \mathcal{G}^*_{N,m}$ has minimum sensitivity to crosstalk, or that it is a minimum crosstalk graph, if for any other graph $\mathcal{G} = \mathcal{G}_{N,m}$ of the same order and size, we have*

$$K(\mathcal{G}^*) \leq K(\mathcal{G}). \tag{4.4}$$

In what follows, we analyze the crosstalk sensitivity of networks in three different cases. The first is when the crosstalk affinity of each vertex depends only on its physical properties, specific to each network. The second and third cases assume that the crosstalk affinity of each vertex is an increasing function of its degree. This is a reasonable assumption, especially in biological networks, natural or man-made. Biological molecules (proteins, for example) which are supposed to interact with many others, have a less specific shape. This is exactly the reason why they can also randomly interact with many other unrelated molecules. The same argument holds for molecules that are very specific, and can interact with a small subset of the rest of the molecules in the network. In DNA systems [25, 38, 58], if a strand is designed to interact with many others, it is usually longer and, as a result it can stick to many unrelated strands that are floating in the solution. In communication networks, transceivers that are sensitive to a large spectrum of frequencies are more likely to pick up noise or other random signals from surrounding transmitters, and the opposite is true for transceivers which have a narrow input and output frequency range.

4.2.2 Crosstalk Specific to Individual Vertices

In this first case, we assume that the amount of crosstalk depends on the specific vertices of the network and the way they interact with each other. For this reason,

we have no control over their intensity. The intensity of interactions may depend on many different properties of each vertex, but in what follows, we will always normalize the crosstalk interaction intensity among any pair of vertices. The larger the size of a graph with constant order, the fewer the pairs of vertices that can spuriously interact with each other, decreasing the overall sensitivity to crosstalk. Two extreme cases are the tree graphs, when the number of edges is equal to the minimum possible ($m = N-1$, given that the graph is required to be connected) and the complete graphs, where $m = \binom{N}{2}$. In a tree graph we have $\binom{N}{2} - (N-1) = \binom{N-1}{2}$ pairs of vertices that may spuriously interact. The complete graph has no crosstalk interactions, simply because every vertex is designed to interact with every other. In a network with size m, the overall crosstalk sensitivity $K(\mathcal{G})$ will be bounded by these two extremes,

$$0 \leq K \leq \binom{N-1}{2} ||R^{xt}||_{\infty}. \tag{4.5}$$

We can optimize the structure of this network, by connecting the m pairs of vertices with the strongest interactions among each other, thus reducing $||R^{xt}||_{\infty}$.

A more interesting problem arises when the intensity of crosstalk interaction is a function of the vertex degrees. We denote the crosstalk affinity of vertex u with degree d_u and vertex v with degree d_v as

$$w^{xt}_{u,v} = g(d_u, d_v) \tag{4.6}$$

which represents the intensity of the crosstalk interaction between u and v. The function g is assumed to be a strictly increasing function in both its arguments, reflecting our assumption that the more neighbors a vertex has, the more likely it is to interact with vertices it is not connected with. We also assume that g is symmetric,

$$g(d_u, d_v) = g(d_v, d_u). \tag{4.7}$$

The affinity $f = f(d_u)$ of a vertex u is similarly defined as its tendency to interact with other vertices of the network with which it is not connected in the initial network, and consequently it is not supposed to interact with. The following sections present two scenarios, the first defines g as the sum of the affinity functions of the two vertices, and the other defines g as the product of their affinities.

4.3 Pairwise Crosstalk Interactions as the Sum of Individual Affinities

If $g(x, y)$ is an additive function, we can write it as

$$g(x, y) = f(x) + f(y) \tag{4.8}$$

where $f : \mathbb{R}^+ \to \mathbb{R}^+$ is defined as the affinity of each individual vertex. We will refer to this type of crosstalk interactions in the network as *additive crosstalk*. The overall crosstalk sensitivity will be

$$\begin{aligned} K(\mathcal{G}) &= \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} w_{u,v} g(u, v) \\ &= \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} w_{u,v}(f(u) + f(v)). \end{aligned} \tag{4.9}$$

If we group all the terms that correspond to each vertex together, we can rewrite the last equation as

$$K(\mathcal{G}) = \sum_{u\in\mathcal{V}(\mathcal{G})} \left(\sum_{v\notin\mathcal{N}_u} w_{u,v} \right) f(u). \tag{4.10}$$

The last equation clearly shows how to reduce crosstalk if vertex affinities are fixed. We either need to make sure that vertices that are not connected do not significantly affect the function of the network (reduce the respective weights), or have these vertices interact with each other, removing the respective weights that

contribute to a large increase in the overall crosstalk sensitivity.

If all weights are equal, in other words, if

$$w_{u,v} = w \quad \forall u, v \in \mathcal{V}(\mathcal{G}), \tag{4.11}$$

then the overall crosstalk can be written as:

$$K(\mathcal{G}) = w \sum_{u \in \mathcal{V}(\mathcal{G})} (N - 1 - d_u)\, f(u). \tag{4.12}$$

The last equation clearly shows that the overall crosstalk is *independent* of the network structure, as it only depends on the affinity of each vertex and its degree, not the neighborhood of each vertex. In what follows, we will assume that all weights are equal to one for simplicity. In addition, we will be assuming that the affinity functions and the crosstalk intensity functions depend on the degree of the respective vertices, as mentioned in the introduction.

4.3.1 Structure of Networks with Minimum Sensitivity to Additive Crosstalk

If the affinity of each vertex is a strictly increasing function of its degree, and crosstalk interactions among all pairs of vertices contribute equally to the overall crosstalk, then we can rewrite equation 4.9 as:

$$K(\mathcal{G}) = \sum_{(u,v) \notin \mathcal{E}(\mathcal{G})} g(u, v) = \sum_{(u,v) \notin \mathcal{E}(\mathcal{G})} g(d_u, d_v) = \sum_{(u,v) \notin \mathcal{E}(\mathcal{G})} \left(f(d_u) + f(d_v)\right). \tag{4.13}$$

We will first present some lemmas that will help us formulate the main theorem of this section.

Lemma 25. *Assume that the function $h : \mathbb{R}^+ \to \mathbb{R}^+$ can be written as the product of two functions f and g, with $f : \mathbb{R}^+ \to \mathbb{R}^+$ being strictly increasing, differentiable, and*

concave function, and $g : \mathbb{R}^+ \to \mathbb{R}^+$ *is strictly decreasing, differentiable, and concave. Then* h *is strictly concave.*

Proof. The second derivative of h is

$$h''(x) = f''(x)g(x) + 2f'(x)g'(x) + f(x)g''(x). \tag{4.14}$$

All three products of the right hand side are negative, since $f(x) > 0$, $f'(x) > 0$, $f''(x) \leq 0$, $g(x) > 0$, $g'(x) < 0$, and $g''(x) \leq 0$, therefore $h''(x) < 0$. □

Lemma 26. *Assume we have a differentiable function* $f : \mathbb{R}^+ \to \mathbb{R}^+$*, with* $f'(x) < 0$ *and* $f''(x) \leq 0$ *for every* $x \in \mathbb{R}^+$*. Further assume that we have four numbers* $a, b, c, d \in \mathbb{R}^+$*, such that* $0 \leq a < c \leq d < b < \infty$*, and for which* $a + b = c + d$*. Then* $f(a) + f(b) \leq f(c) + f(d)$*.*

Proof. According to the mean value theorem, there exist numbers $a \leq \xi_1 \leq c$ and $d \leq \xi_2 \leq b$ such that

$$f^{'}(\xi_1) = \frac{f(c) - f(a)}{c - a} \tag{4.15}$$

and

$$f^{'}(\xi_2) = \frac{f(b) - f(d)}{b - d}. \tag{4.16}$$

Since f is concave,

$$f'(\xi_2) \leq f'(\xi_1). \tag{4.17}$$

Replacing these derivatives from the equations above, we get

$$\frac{f(b) - f(d)}{b - d} \leq \frac{f(c) - f(a)}{c - a}. \tag{4.18}$$

Since $a + b = c + d$, the denominators are equal, and as a consequence

$$f(b) - f(d) \leq f(c) - f(a). \tag{4.19}$$

After rearranging the terms, the result follows. □

Definition 6. *The total affinity h_u of a vertex u is equal to the sum of the affinities of u to all the vertices of the network that it is not connected with.*

Theorem 14. *Assume that the function $f : \mathbb{R}^+ \to \mathbb{R}^+$ that defines the affinity of each vertex of the network is a strictly increasing concave function of the vertex degree. Then the total affinity of every vertex u is equal to*

$$h_u = (N - 1 - d_u) f(d_u) \tag{4.20}$$

and the degree distribution of the network with the minimum additive sensitivity to crosstalk is inhomogeneous.

Proof. As shown in equation (4.13), the crosstalk intensity between any pair of vertices is only a function of their degrees. Furthermore, the affinity of each vertex is only a function of its degree, and independent of the vertices it interacts with. As a result,

$$h_u = \sum_{v \notin \mathcal{N}_u} f(d_u) \implies h_u = (N - 1 - d_u) f(d_u). \tag{4.21}$$

Since h_u is also a function of the vertex degree, we will denote the total affinity as $h_u = h(d_u)$. From Lemma 25, the function h is positive, strictly increasing and concave, since $N - 1 - d_u$ is a decreasing positive linear function for $1 \leq d_u \leq N - 1$ and f is assumed to be positive, increasing, and concave.

The network which is most robust to crosstalk will be the one with degree distribution $\mathbf{d}^* = [d_1, d_2, \ldots, d_N]$ such that

$$\mathbf{d}^* = \underset{\mathbf{d} \in \mathcal{P}}{\arg\min} \left\{ \sum_{(u,v) \notin \mathcal{E}(\mathcal{G})} (f(d_u) + f(d_v)) \right\} \tag{4.22}$$

under the constraints

$$1 \leq d_u \leq N - 1 \quad \forall u \in \mathcal{V}(\mathcal{G}) \tag{4.23}$$

$$\sum_{u=1}^{N} d_u = 2m \geq 2(N-1) \tag{4.24}$$

where $\mathcal{P}$ is the space of graphic degree sequences for connected graphs. Rewriting equation (4.13), we find that

$$K = \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} (f(d_u) + f(d_v)) = \sum_{u\in\mathcal{V}(\mathcal{G})} (N-1-d_u) f(d_u) = \sum_{u\in\mathcal{V}(\mathcal{G})} h(d_u). \tag{4.25}$$

The function K is a sum of concave functions, and thus concave. According to Lemma 26, the optimal $\mathbf{d}^*$ will be achieved for d_u's at the boundaries of the plausible values in the inequality (4.23). □

Corollary 22. *When the crosstalk affinity between two vertices is an additive function of their degrees, the overall crosstalk sensitivity only depends on the vertex degrees, and does not depend on the structure of the network. In other words, how robust the network is to crosstalk only depends on its degree distribution.*

Proof. From equation (4.25), the overall sensitivity of the network only depends on the degree distribution, and is independent of its structure. □

Corollary 23. *The tree graph that is most robust to additive crosstalk is the star graph.*

Proof. The star network has one vertex with full degree and this vertex has no crosstalk interactions. The $N-1$ peripheral vertices have degree 1, and may spuriously interact with each other. Hence,

$$\begin{aligned} K_{star} =& \sum_{u\in\mathcal{V}(\mathcal{G})} (N-1-d_u) f(d_u) \\ =& (N-1)(N-2) f(1) + (N-1)\cdot 0 \\ =& 2\binom{N-1}{2} f(1). \end{aligned} \tag{4.26}$$

Every other graph will have vertices with degrees $1 \leq d \leq N-2$ with at least

one vertex with degree $d \geq 2$. The total number of crosstalk interactions is equal to twice the number of edges that are *not* present in the network, namely $2\binom{N-1}{2}$. The sum of all the crosstalk interactions K_{alt} will then be

$$\begin{aligned} K_{alt} &= \sum_{u \in \mathcal{V}(\mathcal{G})} (N-1-d_u) f(d_u) \\ &> f(1) \sum_{u \in \mathcal{V}(\mathcal{G})} (N-1-d_u) \\ &= f(1)(N(N-1) - 2(N-1)) \\ &= 2\binom{N-1}{2} f(1) \\ &= K_{star}. \end{aligned} \tag{4.27}$$

□

Corollary 24. *Suppose that a vertex u is connected to vertex v with degree d_v, and is not connected to vertex w with degree d_w. If $d_v \leq d_w$, and we can rewire the edge (u, v) to (u, w) while keeping the graph connected, then the sensitivity of the new network to additive crosstalk is smaller.*

Proof. The only vertices whose affinities change are v and w, since the degree of u, along with any other vertex in the network, stays the same. Thus,

$$\begin{aligned} \Delta K =& K_{new} - K_{old} \\ =& (N-d_v) f(d_v - 1) + (N-2-d_w) f(d_w - 1) \\ & - (N-1-d_v) f(d_v) - (N-1-d_w) f(d_w) \\ =& -[h(d_v) - h(d_v - 1)] + [h(d_w + 1) - h(d_w)] \\ <& 0. \end{aligned} \tag{4.28}$$

The last inequality holds because h is concave (Lemma 25) and $d_v < d_w$. □

Corollary 25. *In a graph with minimum sensitivity to additive crosstalk, there is always a vertex with full degree.*

The last corollary guarantees that no matter how we rewire the vertices which do not have full degree, the graph stays connected.

Definition 7. *We define $\mathcal{N}_{x-y}$ as the set of all the neighbors of vertex x except vertex y, such that*

$$\mathcal{N}_{x-y} = \begin{cases} \mathcal{N}_x & \text{if } y \notin \mathcal{N}_x \\ \mathcal{N}_x - \{y\} & \text{if } y \in \mathcal{N}_x. \end{cases} \tag{4.29}$$

Corollary 26. *In a graph with the minimum crosstalk sensitivity, for every pair of vertices a and b, we have*

$$d_a \leq d_b \iff \mathcal{N}_{a-b} \subseteq \mathcal{N}_{b-a}. \tag{4.30}$$

Proof. Counting the elements of the sets $\mathcal{N}_{a-b}$ and $\mathcal{N}_{b-a}$,

$$\mathcal{N}_{a-b} \subseteq \mathcal{N}_{b-a} \implies |\mathcal{N}_{a-b}| \leq |\mathcal{N}_{b-a}|. \tag{4.31}$$

In addition,

$$|\mathcal{N}_a| = |\mathcal{N}_{a-b}| + \delta((a,b) \in \mathcal{E}), \quad |\mathcal{N}_b| = |\mathcal{N}_{b-a}| + \delta((a,b) \in \mathcal{E}) \tag{4.32}$$

where δ is the Kronecker delta function, defined as

$$\delta((u,v) \in \mathcal{E}) = \begin{cases} 1 & \text{if } u \text{ and } v \text{ are connected} \\ 0 & \text{otherwise.} \end{cases} \tag{4.33}$$

As a result,

$$d_a = |\mathcal{N}_a| \leq |\mathcal{N}_b| = d_b. \tag{4.34}$$

The necessity of condition (4.30) is proved, and we will next show the sufficiency of the statement by assuming otherwise and arriving at a contradiction.

Suppose that $d_a \leq d_b$, but $\mathcal{N}_{a-b} \nsubseteq \mathcal{N}_{b-a}$. This means that there is at least one vertex that is connected to a, but is not connected to b. Let $M \in \mathcal{V}(\mathcal{G})$ be a vertex with degree d_M such that

$$d_M \geq d_u \quad \forall u \in \mathcal{N}_a \cap \mathcal{N}_b^c. \tag{4.35}$$

Since the graph $\mathcal{G}$ has minimum sensitivity to crosstalk, by Corollary 24, this means that a is connected to all vertices that have degree larger than M. If it did not, we could rewire the edge (a, M) to connect a with a vertex with a larger degree. The same applies for vertex b, and since $d_a \leq d_b$, the only way that vertex b would not be connected to M, is when it is connected to some other vertex P with degree equal to the degree of M,

$$d_M = d_P = \theta. \tag{4.36}$$

Even in that case, if we rewire the edge that connects b with P, so that after rewiring it connects b with M, the difference of overall crosstalk sensitivity for the graph $\mathcal{G}$ will be negative

$$\begin{aligned} \Delta K &= K_{new} - K_{old} \\ &= h(\theta + 1) + h(\theta - 1) - 2h(\theta) \\ &< 0 \end{aligned} \tag{4.37}$$

because the total crosstalk affinity function h is strictly concave. This means that $\mathcal{G}$ was not a graph with minimum crosstalk sensitivity, which is a contradiction.

The last derivation also shows that in a graph with minimum crosstalk sensitivity,

$$d_a = d_b \iff \mathcal{N}_{a-b} = \mathcal{N}_{b-a}. \tag{4.38}$$

□

4.3.2 Rewiring Algorithm

In this section, we describe an algorithm that can be used to find the structure of the network with the minimum sensitivity to additive crosstalk. We will first explain how it works, and then prove that it always returns the optimal network, regardless of the input graph. The algorithm consists of two steps: The first step (single rewirings) ensures that condition (4.30) holds. In every iteration, it checks if each vertex could be connected to another neighbor with larger degree, therefore decreasing the overall crosstalk sensitivity of the graph, according to Corollary 24. The second step ensures that the required changes that are not possible with single-vertex rewirings actually take place.

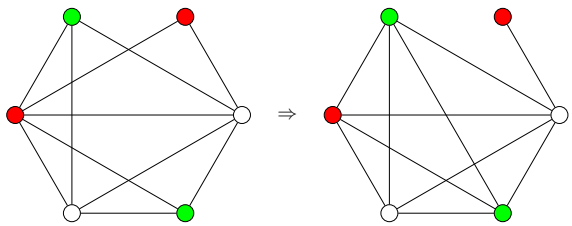


Figure 4.1: An example of a double rewiring needed for the graph to minimize crosstalk sensitivity. If $h_1 + 3h_4 < h_2 + 2h_3 + h_5$ (for example when $h(d) = (N - 1 - d)\sqrt{d}$), there is no edge that can be rewired keeping one of its endpoints, and that would decrease the overall sensitivity. A double rewiring is needed, removing the edge between the red vertices and connecting the green ones with it.

After the first step is finished (there are no more possible single-vertex rewirings), the degrees of all vertices satisfy equation (4.30). The resulting graph is not necessarily one with the minimum crosstalk sensitivity, because we may have a situation like the one depicted in Figure 4.1, where we need to rewire one edge such that we change both vertices it is adjacent to. In this case, we cannot perform a single rewiring, and we need to delete an edge between a pair of vertices, and add it to another. There are only two possible arrangements of the edges among four vertices that may need a double rewiring transformation (shown in Figure 4.2), and the condition that the

degrees of these vertices need to satisfy is shown in Table 4.1.

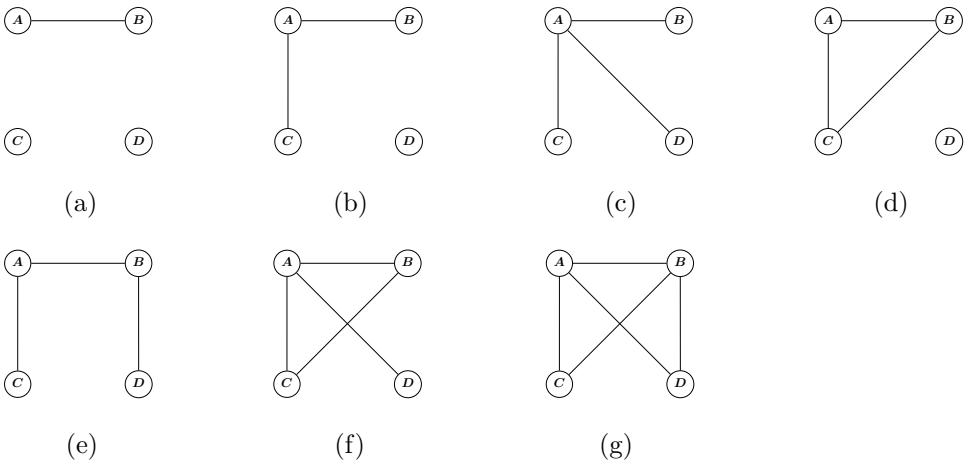


Figure 4.2: All nonisomorphic subgraphs of 4 vertices where $A - B$ and $C \not\sim D$. See Table 4.1 for all the allowed degree correlations among the degrees of the four vertices that satisfy equation (4.30). Subgraphs (c) and (d) may need a rewiring of an edge between A and B to connect vertices C and D. Note that if we apply the double-vertex rewiring, we transform subgraph (c) to subgraph (d) and vice versa.

Theorem 15. *Provided with an arbitrary graph $\mathcal{G}_{N,m}$, Algorithm 1 returns the network $\mathcal{G}^*_{N,m}$ with the minimum sensitivity to additive crosstalk.*

Proof. We need to prove two statements, the first is that the double rewirings do not affect the condition of equation (4.30), and the second is that the algorithm cannot be trapped in a local minimum. Suppose that we perform a double rewiring (removing the edge (A, B) and adding the edge (C,D)) reducing the overall sensitivity to crosstalk by ΔK, and condition (4.30) is violated. This would only be possible if there exists a vertex M with degree d_M such that

$$d_M = d_B - 1 \quad \text{and} \quad \mathcal{N}_{B-A} \neq \mathcal{N}_M, \tag{4.39}$$

so that after removing the edge (A, B), vertices B and M have the same degrees, and condition (4.38) does not hold. The last equation also implies that $(A, M) \in \mathcal{E}$,

Graph	Possible Degree Order	ΔK
a	$d_D \leq d_C \leq d_B \leq d_A$	> 0
b	$d_D \leq d_C \leq d_B \leq d_A$	> 0
	$d_D \leq d_B \leq d_C \leq d_A$	> 0
c	$d_D \leq d_C \leq d_B \leq d_A$	> 0
	$d_D \leq d_B \leq d_C \leq d_A$	> 0
	$d_B \leq d_D \leq d_C \leq d_A$	**?**
d	$d_D \leq d_C \leq d_B \leq d_A$	> 0
	$d_D \leq d_B \leq d_C \leq d_A$	> 0
	$d_D \leq d_B \leq d_A \leq d_C$	**?**
e	$d_D \leq d_C \leq d_B = d_A$	> 0
f	$d_D \leq d_C \leq d_B \leq d_A$	> 0
	$d_D \leq d_B \leq d_C \leq d_C$	> 0
g	$d_D \leq d_C \leq d_B \leq d_A$	> 0

Table 4.1: All the possible degree orderings of the degrees of the four vertices shown in Figure 4.2, with the difference in the overall crosstalk sensitivity if we rewire the edge between A and B to connect vertices C and D, and assuming that all possible single rewirings that decrease crosstalk have already taken place. In all but two cases, the overall sensitivity to crosstalk increases. Cases (c) and (d) may have a positive or negative difference (question marks), depending on the crosstalk function h, and only if the degrees of the vertices A, B, C, D are ordered as shown. Note that case (e) is not possible at any time, but is included here for completeness.

Algorithm 1 Find the Graph with Minimum Crosstalk Sensitivity $\mathcal{G}$

Require: An arbitrary connected graph $\mathcal{G}$ of order N and size m
Ensure: The graph with the minimum sensitivity to crosstalk

changeflag$\leftarrow 1$
while changeflag==1 **do**
  changeflag$\leftarrow 0$
  for $u \in \mathcal{V}(\mathcal{G})$ **do**
    if $\min\limits_{v\in\mathcal{N}_u}\{d_v\} \leq \max\limits_{w\in\mathcal{N}_u^c}\{d_w\}$ **then**
      $x = \arg\min\limits_{v\in\mathcal{N}_u}\{d_v\}$; $y = \arg\min\limits_{w\in\mathcal{N}_u^c}\{d_w\}$
      Remove the edge (u, x);Add the edge (u, y); changeflag$\leftarrow 1$
      if $\mathcal{G}$ becomes disconnected **then**
        Revert the changes
      end if
    end if
  end for
end while

changeflag$\leftarrow 1$
while changeflag==1 **do**
  changeflag$\leftarrow 0$
  for $A, B, C, D \in \mathcal{V}(\mathcal{G})$ **do**
    if ($A-B$ **and** $A-C$ **and** $A-D$ **and** $B\neq C$ **and** $B\neq D$ **and** $C\neq D$) **or** ($A-B$ **and** $A-C$ **and** $B-C$ **and** $A\neq D$ **and** $B\neq D$ **and** $C\neq D$) **then**
      $\Delta h_A = h(d_A) - h(d_A - 1)$;$\Delta h_B = h(d_B) - h(d_B - 1)$
      $\Delta h_C = h(d_C + 1) - h(d_C)$; $\Delta h_D = h(d_D + 1) - h(d_D)$
      $\Delta K_{A,B,C,D} = -\Delta h_A - \Delta h_B + \Delta h_C + \Delta h_D$
    end if
    Find the vertex set (A, B, C, D) with the smallest ΔK
    if $\min \Delta K < 0$ **and** rewiring the edge $A - B$ to $C - D$ keeps $\mathcal{G}$ connected **then**
      Remove the edge (A, B); Add the edge (C, D); changeflag$\leftarrow 1$
    end if
  end for
end while

because otherwise $\mathcal{N}_B$ and $\mathcal{N}_M$ would be identical after removing the edge (A, B). But in that case,

$$\Delta K' = -\Delta h_A - \Delta h_M + \Delta h_C + \Delta h_D < -\Delta h_A - \Delta h_B + \Delta h_C + \Delta h_D = \Delta K \quad (4.40)$$

because h is strictly concave, which means that the edge (a, b) should not be rewired before the edge (a, M).

We will now prove that the algorithm cannot be trapped in a local minimum by showing that all the individual steps needed in order to reach the graph with the minimum overall crosstalk sensitivity can be reordered arbitrarily. This condition is sufficient in order to prove that when there are no such rewirings left, we have reached the optimal graph (see Figure 4.3).

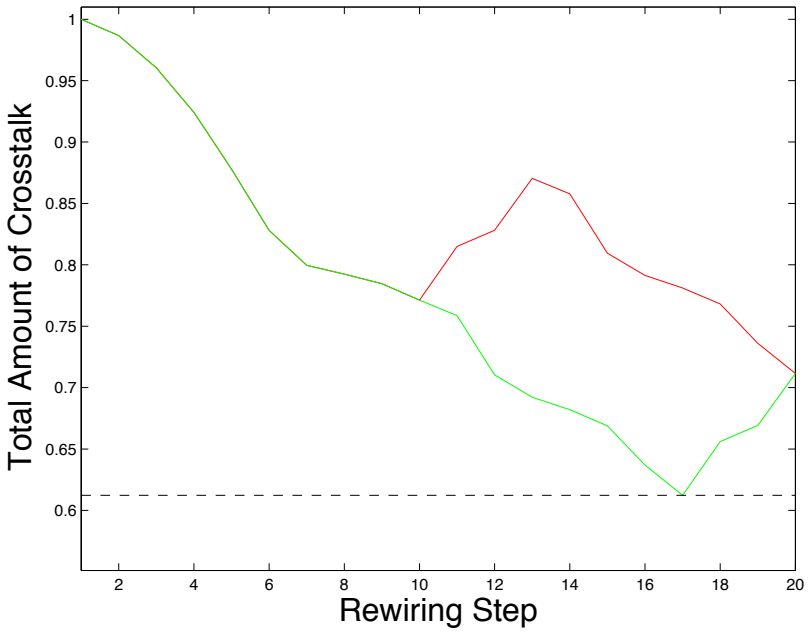


Figure 4.3: Algorithm 1 cannot be trapped in a local minimum. After decreasing the overall crosstalk sensitivity (blue line), suppose it stops at a local minimum. Since we can reorder the steps, and each step decreases the sensitivity at least as much as when taken individually (equation (4.47)), we may perform all the decreasing transformations first and then stop, ending up with a network with smaller overall sensitivity to crosstalk.

The single-vertex rewirings can be performed in an arbitrary order, since the only

way there are no more such rewirings possible is when the network satisfies equation (4.30). So it suffices to prove that the double rewirings can be rearranged. Suppose that we have two or more such rewirings needed. For each pair of vertices that will accept one extra edge, they can have up to one vertex in common, whose degree will increase by at least 2. If they have no vertex in common, the double rewirings are completely independent of each other, so the statement holds. They cannot have the same vertices, because these two nodes will already be connected by an edge after one them is implemented. Now assume that we can perform two double rewirings. The first is moving the edge (A, B) to (C, D), changing crosstalk by ΔK_1, and the second is moving the edge (E, F) to (C, J), changing crosstalk by ΔK_2, based on the current degrees of vertices A, B, C, D, E, F, and J. If C is common in both of them, then if we perform the first rewiring and then perform the second (now changing the overall crosstalk sensitivity by $\Delta K_2'$), we will show that

$$\Delta K_1 < 0, \Delta K_2 < 0 \implies \Delta K_2' \leq \Delta K_2. \tag{4.41}$$

This means that every rewiring makes the subsequent ones decrease the crosstalk even more than their individual contributions, and there is no way that a rewiring that increases crosstalk could enable future rewirings to decrease the overall crosstalk more than they would have otherwise. Denote

$$\Delta K_1 = -\Delta h_A - \Delta h_B + \Delta h_C + \Delta h_D < 0 \tag{4.42}$$

where

$$\Delta h_A = h(d_A) - h(d_A - 1) > 0, \qquad \Delta h_C = h(d_C + 1) - h(d_C) > 0 \tag{4.43}$$

and similarly for the differences of the crosstalk affinities of the other vertices. By

the same token,

$$\Delta K_2 = -\Delta h_E - \Delta h_F + \Delta h_C + \Delta h_J < 0. \tag{4.44}$$

If we perform the first double rewiring ((A, B) is rewired to (C, D)), the new degree of vertex C will increase by 1, and as a result

$$\Delta K_2' = -\Delta h_E - \Delta h_F + \Delta h_C' + \Delta h_J \tag{4.45}$$

where

$$\Delta h_C' = h(d_C + 2) - h(d_C + 1) < \Delta h_C \tag{4.46}$$

since h is strictly concave. It follows that in every case we have

$$\Delta K_2' \leq \Delta K_2. \tag{4.47}$$

The same is true when we have more than one double rewiring that decreases the overall sensitivity and removes two or more edges from a vertex. We can similarly show that the difference of the crosstalk affinity of this vertex increases, and since it is negated, the decrease in crosstalk of the subsequent rewirings will be greater. □

Theorem 16. *We can find the graph with the lowest sensitivity to crosstalk in $\mathcal{O}(N^4)$ time.*

Proof. The first part of the algorithm checks if each edge should be rewired, which takes $\mathcal{O}(N^2)$ steps, because we need to check all the vertex pairs, possibly more than once. The second part of the algorithm checks groups of 4 vertices at a time, which will take $\mathcal{O}(N^4)$ steps. So the overall complexity of Algorithm 1 is $\mathcal{O}(N^4)$. □

Figure 4.4 shows an example all the connected graphs with $N = 6$ vertices with minimum additive crosstalk sensitivity. Figure 4.5 shows the form of the total crosstalk affinity of each vertex, and the overall crosstalk of a graph of order $N = 40$ as its size increases.

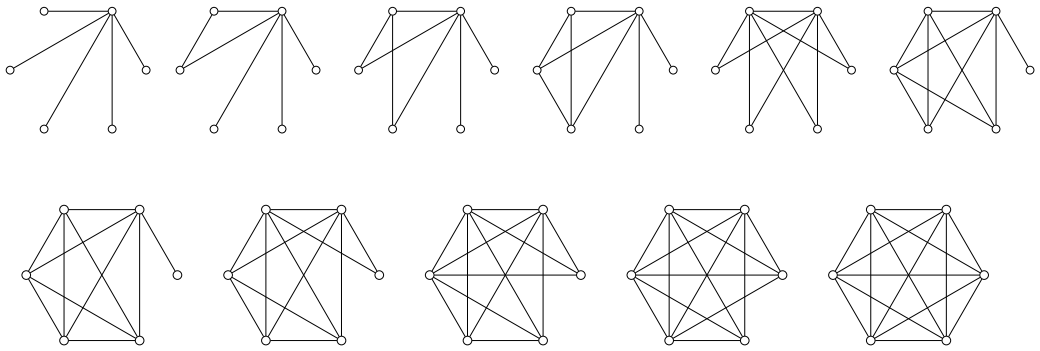


Figure 4.4: Connected graphs of order $N = 6$ and size $5 \leq m \leq 15$ which have the smallest amount of overall crosstalk sensitivity among their vertices, when the intensity of the crosstalk interaction between two vertices is equal to the sum of their individual affinities, and the affinity of each vertex is proportional to the square root of its degree $f(d) = c\sqrt{d}$. Note that we cannot find the optimal graph of size m recursively, by adding one extra edge to the optimal graph of size $m - 1$.

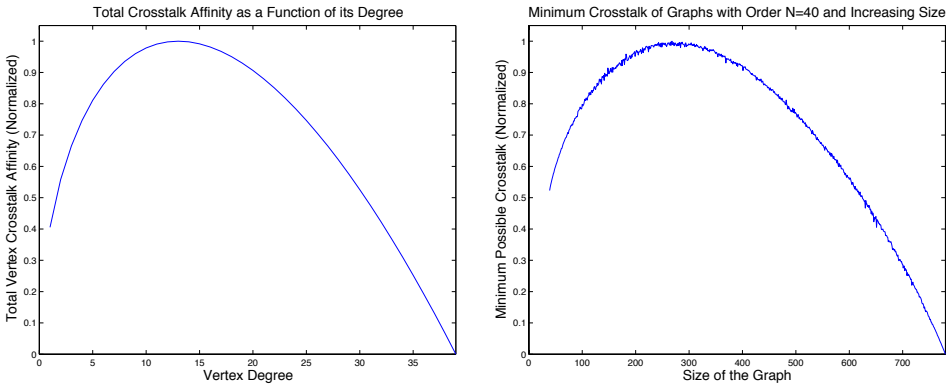


Figure 4.5: **(a)** An example of the overall vertex affinity function h. Here, $f(d) = c\sqrt{d}$, and $h(d) = (N - 1 - d)f(d)$. **(b)** The minimum possible overall crosstalk sensitivity of all graphs with $N = 40$ vertices and $39 \leq m \leq 780$ edges has the shape of the vertex crosstalk function. It is not smooth because $\mathbf{d}^*$ needs to satisfy additional constraints, most notably $\mathbf{d}^* \in \mathcal{P}$.

4.4 Pairwise Crosstalk Interactions as the Product of Individual Affinities

If the crosstalk between two vertices is proportional to the product of each node's affinity, then we can write the pairwise affinity function $g(x, y)$ as

$$g(x, y) = f(x)f(y). \tag{4.48}$$

We will refer to this type of crosstalk interaction in the network as *geometric crosstalk* or *multiplicative crosstalk* . If the affinity of each node is a function of its degree, the overall crosstalk sensitivity will be equal to

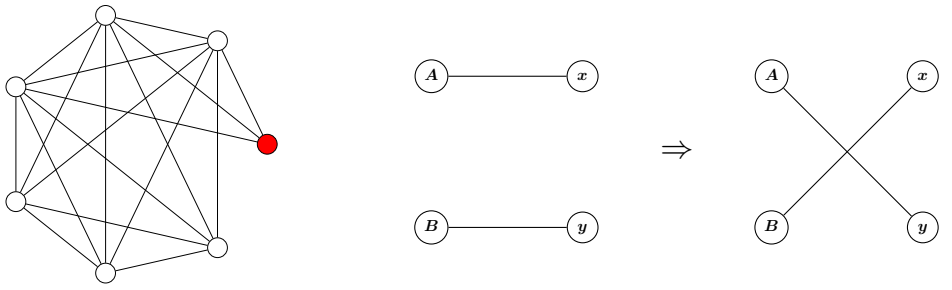
$$K(\mathcal{G}) = \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} g(d_u, d_v) = \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} f(d_u)f(d_v). \tag{4.49}$$

Once again, we consider the vertex affinity function f to be a positive, strictly increasing, and concave function of the degree of each vertex. We will first find the networks with the minimum crosstalk sensitivity without any additional constraints. We will then describe the connectivity of the networks with a fixed degree sequence which have the lowest sensitivity to multiplicative crosstalk.

4.4.1 Structure of Networks with Minimum Sensitivity to Geometric Crosstalk

We first prove some lemmas that will be needed in order to find the form of these graphs. In this chapter, the only type of almost complete graph that we will need is of type I, and for simplicity will just call it almost complete.

Lemma 27. *In a graph with the minimum sensitivity to multiplicative crosstalk, if*



(a) (b)

Figure 4.6: **(a)** The type I almost complete graph (first defined in Chapter 3) consists of a clique of order $N-1$, and one peripheral vertex (shown in red). **(b)** If $d_A < d_B$ and $d_x > d_y$ then rewiring the edges $A-x$ and $B-y$ as shown will decrease the overall crosstalk sensitivity of the graph without changing its degree sequence.

$A, B, x, y \in \mathcal{V}(\mathcal{G})$, *then*

$$d_A \leq d_B \quad \Longrightarrow \quad d_x \leq d_y \qquad \forall x \in \mathcal{N}_A \cap \mathcal{N}_B^c \quad \textit{and} \quad y \in \mathcal{N}_B. \tag{4.50}$$

Proof. If $\mathcal{N}_\mathcal{A} \subseteq \mathcal{N}_B$ then the result holds trivially, since $\mathcal{N}_A \cap \mathcal{N}_B^c = \emptyset$. Otherwise, there is at least one vertex x that is connected to A and not connected to B. If we rewire the edges (A, x) and (B, y) to (A, y) and (B, x) as shown in Figure 4.6(b), the degree distribution stays the same for every vertex in the graph. The difference in the overall sensitivity is

$$\begin{aligned} \Delta K =& f(d_A)f(d_x) + f(d_B)f(d_y) - f(d_A)f(d_y) - f(d_B)f(d_x) \\ =& (f(d_B) - f(d_A))(f(d_y) - f(d_x)). \end{aligned} \tag{4.51}$$

If the graph was optimal before the rewiring, this difference should be nonnegative, and since $d_A \leq d_B$, the this can only be true if $d_x \leq d_y$. □

Lemma 28. *Assume that a graph $\mathcal{G}_{N,m}$ with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$ has the minimum sensitivity to multiplicative crosstalk and that the vertex affinity $f : \mathbb{R}^+ \to$*

$\mathbb{R}^+$ *is a strictly increasing concave function. Then*

$$(a,b) \in \mathcal{E} \implies (a,x) \in \mathcal{E} \quad \forall a \in \mathcal{V} \quad \textit{and} \quad \forall x \in \mathcal{V}, \textit{such that} \quad d_x > d_b. \tag{4.52}$$

Proof. Suppose that this is not the case, and there exists a vertex a and two vertices b and c such that

$$(a,b) \in \mathcal{E}, \quad (a,c) \notin \mathcal{E} \quad \text{with} \quad d_b < d_c. \tag{4.53}$$

Then, by rewiring the edge between a and b so that it now connects vertices a and c, the difference in the crosstalk sensitivity will be

$$\begin{aligned} \Delta K =& K_{new} - K_{old} \\ =& f(d_a)(f(d_b) - f(d_c)) \\ &+ (f(d_b - 1) - f(d_b)) \sum_{u \in \mathcal{N}^c_{b-a}} f(d_u) \\ &+ (f(d_c + 1) - f(d_c)) \sum_{v \in \mathcal{N}^c_{c-a}} f(d_v). \end{aligned} \tag{4.54}$$

Since f is strictly increasing,

$$f(d_b) < f(d_c), \quad f(d_b - 1) - f(d_b) < 0 \quad \text{and} \quad f(d_c + 1) - f(d_c) > 0 \tag{4.55}$$

and from the concavity of f we see that

$$|f(d_b - 1) - f(d_b)| > |f(d_c + 1) - f(d_c)|. \tag{4.56}$$

In addition, from Lemma 27,

$$\sum_{u \in \mathcal{N}_{b-a}} f(d_u) < \sum_{v \in \mathcal{N}_{c-a}} f(d_v). \tag{4.57}$$

Taking into account that the degrees of all other vertices in $\mathcal{G}$ remain constant after the transformation,

$$\sum_{\substack{w\in\mathcal{V}(\mathcal{G})\\ w\neq b, w\neq c}} f(d_w) = \text{constant} \tag{4.58}$$

we find that

$$\sum_{u\in\mathcal{N}_{b-a}^{c}} f(d_u) > \sum_{v\in\mathcal{N}_{c-a}^{c}} f(d_v). \tag{4.59}$$

Combining the above equations,

$$\Delta K < 0. \tag{4.60}$$

This means that the overall crosstalk sensitivity of the graph $\mathcal{G}$ has decreased after applying this transformation, which is a contradiction. □

The last lemma shows a necessary condition for the optimality of a network with respect to crosstalk: Every vertex should be connected with vertices of the largest degree possible.

Corollary 27. *In the network with the minimum sensitivity to multiplicative crosstalk, the neighborhood set of every vertex u with degree d_u is*

$$\mathcal{N}_u = \{N - d_u + 1, N - d_u + 2, \ldots, N\}. \tag{4.61}$$

Corollary 28. *In a network with the minimum sensitivity to multiplicative crosstalk, there is at least one vertex with full degree.*

Proof. From Corollary 27, u will connect to the d_u vertices with the largest degrees. In addition, every vertex u has degree $d_u \geq 1$, since we assumed that the graph is connected. As a result, the vertex with the largest degree in the network is connected to all other $N-1$ vertices in the graph. □

Now we are ready to prove the main theorem of this subsection.

Theorem 17. *If the crosstalk affinity of each vertex is a positive, increasing, and concave function of its degree, then a graph with minimum overall multiplicative crosstalk sensitivity consists of a complete or almost complete subgraph, and the vertices that do not belong to it have degree equal to* 1*. All vertices of degree* 1 *have the same neighbor vertex, which is a vertex with full degree.*

Proof. We will start from any graph $\mathcal{G}$ that satisfies equations (4.50) and (4.52) but does not have the aformentioned form. Then, with transformations that only reduce the sensitivity to crosstalk, we will arrive at the graph with this structure. Suppose that the initial graph $\mathcal{G}$ has a degree distribution

$$\mathbf{d} = [d_1, d_2, \ldots, d_N] \tag{4.62}$$

where the elements of the vertex set $\mathcal{V}(\mathcal{G}) = \{1, 2, \ldots N\}$ are ordered in increasing degree, such that

$$1 \leq d_1 \leq d_2 \leq \cdots \leq d_N \leq N - 1. \tag{4.63}$$

The overall sensitivity to crosstalk can be written as

$$\begin{aligned} K(\mathcal{G}) &= \sum_{(u,v)\notin\mathcal{E}(\mathcal{G})} f(d_u) f(d_v) \\ &= \sum_{u\in\mathcal{V}(\mathcal{G})} f(d_u) \left[\sum_{\substack{v\in\mathcal{V}(\mathcal{G}) \\ v>u, v\notin\mathcal{N}_u}} f(d_v) \right]. \end{aligned} \tag{4.64}$$

Applying the above equation to the enumerated set of vertices,

$$K(\mathcal{G}) = \sum_{u=1}^{N} f(d_u) \left[\sum_{\substack{v=u+1 \\ v\notin\mathcal{N}_u}}^{N} f(d_v) \right]. \tag{4.65}$$

Taking into account Corollary 27 (equation (4.61)), we can simplify the expression of

the overall crosstalk sensitivity as

$$K(\mathcal{G}) = \sum_{u=1}^{N} f(d_u) \left[\sum_{v=u+1}^{N-d_u} f(d_v) \right]. \tag{4.66}$$

Now, suppose that $\mathbf{d}$ is not a degree sequence that corresponds to the form in the statement of the theorem. We pick the vertex a with the smallest possible degree such that $d_a > 1$. We also pick vertex $b = a + 1$, with degree $d_b \geq d_a$. From equation (4.61), we get

$$\mathcal{N}_{a-b} \subseteq \mathcal{N}_{b-a}. \tag{4.67}$$

More precisely,

$$\mathcal{N}_a = \{N - d_a + 1, N - d_a + 2, \ldots, N\} \quad \text{and} \quad \mathcal{N}_b = \{N - d_b + 1, N - d_b + 2, \ldots, N\}. \tag{4.68}$$

By successively rewiring edges in $\mathcal{G}$, we will change the overall sensitivity each time by ΔK, and when the graph is transformed to $\mathcal{G}^*$ with

$$K(\mathcal{G}^*) \leq K(\mathcal{G}), \tag{4.69}$$

no additional transformations will be possible. If $\mathcal{G}$ does not have the form in the statement of the theorem, there is at least one edge in the graph that can be rewired from vertex a to vertex b. We delete the edge $(a, N + 1 - d_a)$, and add an edge to connect vertices b and $N - d_b$. According to Corollary 28, any such transformation will keep the graph connected, because the vertex with full degree is connected to every other vertex in the network as long as we do not remove any edge from a vertex with degree 1. The size of the graph remains the same, and the new neighborhoods of vertices a and b are

$$\mathcal{N}'_a = \{N - d_a + 2, N - d_a + 3 \ldots N\} \quad \text{and} \quad \mathcal{N}'_b = \{N - d_b, N - d_b + 1, \ldots N\}. \tag{4.70}$$

We denote the vertices with the smallest degrees that are neighbors to b and a as $C = N + 1 - d_b$ and $D = N + 1 - d_a$, respectively. According to inequality (4.63),

$$C \leq D \implies d_C \leq d_D. \tag{4.71}$$

The difference in crosstalk after applying the transformation is

$$\begin{aligned}\Delta K =& f(d_a - 1) \sum_{u=a+1}^{N+1-d_a} f(d_u) + f(d_b + 1) \sum_{v=b+1}^{N-1-d_b} f(d_v) \\ &- f(d_a) \sum_{u=a+1}^{N-d_a} f(d_u) + f(d_b) \sum_{v=b+1}^{N-d_b} f(d_v) \\ &+ (f(d_C + 1) + f(d_D - 1)) \sum_{z=1}^{a-1} f(1) \\ &- (f(d_C) + f(d_D)) \sum_{z=1}^{a-1} f(1).\end{aligned} \tag{4.72}$$

Rearranging the terms, we find that

$$\begin{aligned}\Delta K =& [f(d_a - 1) - f(d_a)] \left[\sum_{u=a+1}^{N-d_a} f(d_u) \right] \\ &+ [f(d_b + 1) - f(d_b)] \left[\sum_{v=b+1}^{N-d_b} f(d_v) \right] \\ &+ [f(d_a - 1) f(d_D) - f(d_b + 1) f(d_C)] \\ &+ [f(d_D - 1) - f(d_D) + f(d_C + 1) - f(d_C)] \left[\sum_{z=1}^{a-1} f(1) \right].\end{aligned} \tag{4.73}$$

The sum above is clearly negative, so the transformation reduces the overall sensitivity to crosstalk. We may repeat the process described here, reducing it at each step, until we transform the initial graph to the graph described in the theorem statement. □

Figure 4.7 shows all the graphs of order $N = 6$ and size $5 \leq m \leq 15$ with the minimum multiplicative crosstalk sensitivity.

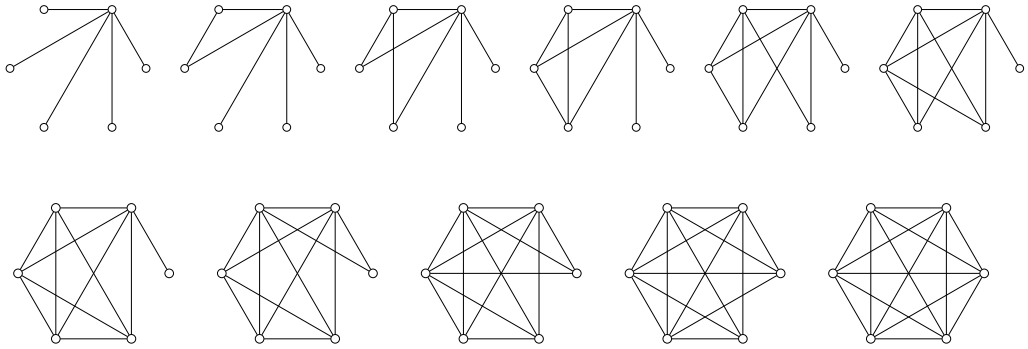


Figure 4.7: Connected graphs of order $N = 6$ and size $5 \leq m \leq 15$ which have the smallest overall crosstalk sensitivity, when the pairwise crosstalk intensity between two vertices is equal to the product of the individual affinities, and the affinity of each vertex is a positive, increasing and concave function of its degree.

Theorem 18. *If the crosstalk affinity of each vertex is a positive, increasing and concave function f of its degree, then the minimum multiplicative crosstalk sensitivity of a graph is equal to*

$$\begin{aligned} K =& (d-1-\alpha)f(\alpha)f(d-2) + (N-d)f(\alpha)f(1) \\ &+ (N-d)(\alpha-1)f(d-1)f(1) \\ &+ (N-d)(d-1-\alpha)f(d-2)f(1) \\ &+ \binom{N-d}{2} f^2(1) \end{aligned} \tag{4.74}$$

where

$$d = \left\lceil \frac{1}{2}\left(3 + \sqrt{9+8m-8N}\right) \right\rceil \tag{4.75}$$

is the number of vertices in the (almost) complete subgraph and

$$\alpha = m - \binom{d-1}{2} - (N-d) \tag{4.76}$$

is the degree of the vertex with the smallest number of neighbors in it.

Proof. The overall sensitivity will be calculated by computing intensity of the crosstalk interactions among the different vertex groups in the minimum crosstalk sensitivity graphs. Suppose that the almost complete (or complete) subgraph consists of d vertices, $d-1$ of which form a complete subgraph and the peripheral vertex of the subgraph has degree has degree α. The degree of the peripheral vertex is allowed to be equal to $d-1$, in which case, we have a complete subgraph. In addition, there are $N-d$ vertices with degree equal to 1. Counting all edges of the graph, we find that

$$m = \binom{d-1}{2} + \alpha + (N-d). \tag{4.77}$$

This equation needs to be solved for the variables d and α, under the conditions

$$\alpha \in \mathbb{N}^*, d \in \mathbb{N}^* \tag{4.78}$$

and

$$2 \leq \alpha \leq d-1. \tag{4.79}$$

As mentioned above, when $\alpha = d-1$, we have a complete subgraph, otherwise we have an almost complete subgraph. In order to find the order of the almost complete graph, we assume that the optimal graph consists of a complete subgraph of order d_0, and that $\alpha = d_0 - 1$. Then the equation that needs to be solved has only one variable d_0

$$\binom{d_0}{2} + (N-d_0) = m \tag{4.80}$$

which after rearranging the terms becomes a second-order polynomial with respect to d_0:

$$d_0^2 - 3d_0 + (2N - 2m) = 0 \tag{4.81}$$

with roots

$$d_0 = \frac{1}{2}\left(3 \pm \sqrt{9 + 8m - 8N}\right). \tag{4.82}$$

One of the solutions will be zero or negative ($m \geq N$ for connected graphs that are not trees), so we need to pick the positive solution. In addition, d needs to be an integer. Since we have artificially increased α in the above solution, $d \geq d_0$ and d is the smallest integer that is larger than d_0, so we need to pick the ceiling of the positive real number from the above equation,

$$d = \left\lceil \frac{1}{2}\left(3 + \sqrt{9 + 8m - 8N}\right) \right\rceil. \tag{4.83}$$

The value of α can now be found from equation (4.77). The structure of the optimal network is now specified, and we need to add up all the crosstalk contributions from all the vertices of the graph:

- Crosstalk between the peripheral vertex and the $d - 1 - \alpha$ vertices of degree $d - 2$ of the almost complete graph that it is not connected to:

$$K_1 = (d - 1 - \alpha) f(\alpha) f(d - 2). \tag{4.84}$$

- Crosstalk between the peripheral vertex and the $N - d$ single-edge vertices:

$$K_2 = (N - d) f(\alpha) f(1). \tag{4.85}$$

- Crosstalk among the single-edge vertices:

$$K_3 = \binom{N - d}{2} f^2(1). \tag{4.86}$$

- Crosstalk among the vertices that are connected to the peripheral vertex and the single-edge vertices:

$$K_4 = (\alpha - 1)(N - d) f(d - 1) f(1). \tag{4.87}$$

In the equation above, the first term is $\alpha - 1$ instead of α because the vertices of degree 1 are all connected to one of these α vertices of the (almost) complete graph, and the crosstalk of this vertex is zero.

- Crosstalk among the vertices of the almost complete graph that are not connected to the peripheral vertex, and the single-edge vertices:

$$K_5 = (d - 1 - \alpha)(N - d)f(d - 2)f(1). \tag{4.88}$$

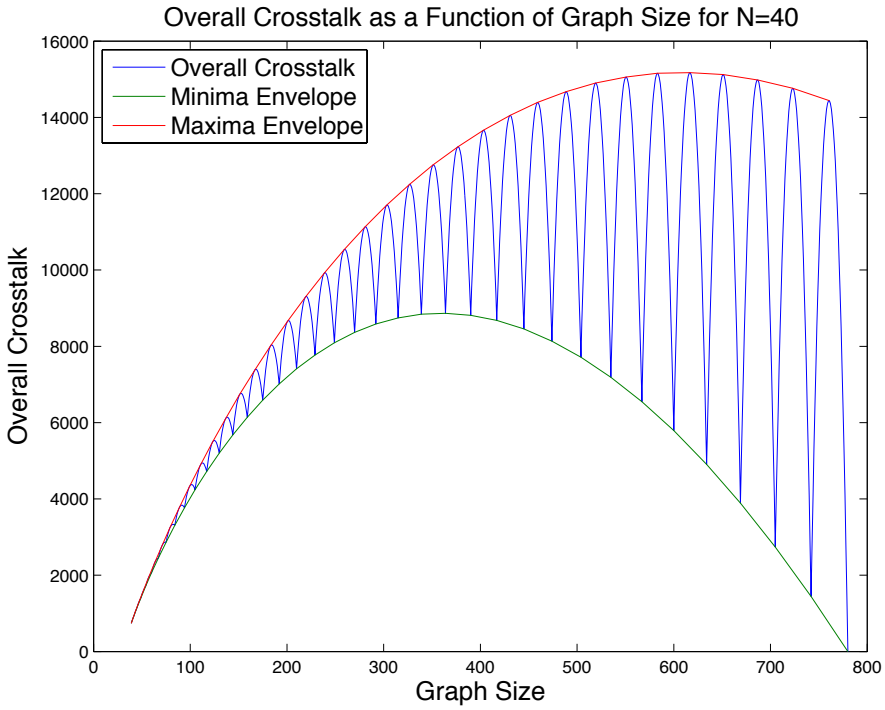


Figure 4.8: The minimum possible overall crosstalk sensitivity of a graph with 40 vertices and a size of $39 \leq m \leq 780$ when the vertex crosstalk affinity is $f(d) = d$. The green and red lines are the envelopes for the local minima and local maxima, respectively.

Adding up the above terms, we get the total amount of crosstalk interactions in the network:

$$K = K_1 + K_2 + K_3 + K_4 + K_5. \tag{4.89}$$

$$
\begin{aligned}
K = & \underbrace{f(\alpha)\left[(d-1-\alpha)f(d-2)+(N-d)f(1)\right]}_{\text{Total crosstalk of the peripheral vertex}} \\
& + \underbrace{(N-d)f(1)\left[(\alpha-1)f(d-1)+(d-1-\alpha)f(d-2)\right]}_{\text{Crosstalk between the almost complete graph and the single-edge vertices}} \\
& + \underbrace{\binom{N-d}{2}f^2(1)}_{\text{Crosstalk among the single-edge vertices}}.
\end{aligned} \tag{4.90}
$$

□

An example of the overall crosstalk sensitivity as we increase the size of the graph of given order is shown in Figure 4.8. The overall crosstalk sensitivity has "ripples". This happens because as we add more edges, the average degree of the vertices increases, and the average crosstalk intensity among vertices also becomes larger. After adding one edge between two nodes, the average decrease in crosstalk also increases, because there is no spurious interaction between them any more. Consequently, the variance of the overall crosstalk sensitivity increases with number of edges in the network. The local minima are achieved when the graph consists of a complete subgraph and the rest of the vertices have degree equal to one.

4.4.2 Minimization of Crosstalk Sensitivity For Networks with Fixed Degree Sequence

Suppose that we have a network with a given degree distribution

$$\mathbf{d} = [d_1, d_2, \ldots, d_N] \tag{4.91}$$

where the degree of each vertex $1 \leq k \leq N$ is fixed and equal to d_k. If we are free to choose the structure of the network, in order to minimize the overall crosstalk sensitivity of the network, we need to connect vertices with similar degrees together, as the next theorem shows.

Theorem 19. *Suppose that we have a network with a given degree distribution* **d**. *Then, the structure that minimizes the sensitivity to multiplicative crosstalk is assortative in the vertex degrees.*

Proof. Suppose we have a network, where vertices A and B are connected, vertices C and D are also connected, and there are no more edges among them. One way to change the structure without changing the degree sequence of the graph is shown in Figure 4.9. This method (first described in [46]) keeps the degrees of each vertex constant.

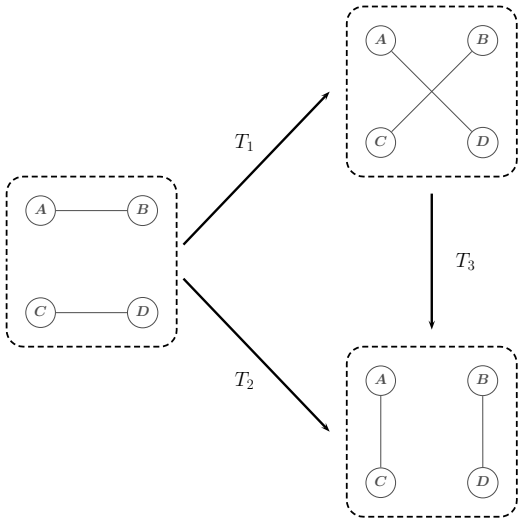


Figure 4.9: A rewiring method that keeps the degree of each vertex constant. Assume that vertices A and B are connected, as are C and D. There are no other edges in this induced subgraph. We can rearrange the edges by connecting A with D and B with C, or by connecting A with C and B with D. Each transformation can be reversed, and we can go from any subgraph to another in one step.

Initially, the overall crosstalk sensitivity among the four vertices A, B, C, and D is

$$M_0 = f(d_A)f(d_C) + f(d_A)f(d_D) + f(d_B)f(d_C) + f(d_B)f(d_D). \tag{4.92}$$

Referring to Figure 4.9, after transformation T_1, the overall crosstalk sensitivity

among the same vertices is

$$M_1 = f(d_A)f(d_B) + f(d_A)f(d_C) + f(d_B)f(d_D) + f(d_C)f(d_D) \tag{4.93}$$

and similarly, if we apply the second transformation T_2

$$M_2 = f(d_A)f(d_B) + f(d_A)f(d_D) + f(d_B)f(d_C) + f(d_C)f(d_D). \tag{4.94}$$

The difference of the overall crosstalk sensitivity of the network after applying T_1 is

$$\Delta K_{T_1} = f(d_A)f(d_B) - f(d_A)f(d_D) - f(d_B)f(d_C) + f(d_C)f(d_D) \tag{4.95}$$

$$\Delta K_{T_1} = \left(f(d_C) - f(d_A)\right)\left(f(d_D) - f(d_B)\right). \tag{4.96}$$

With transformation T_2, we get

$$\Delta K_{T_2} = f(d_A)f(d_B) - f(d_A)f(d_C) - f(d_B)f(d_D) + f(d_C)f(d_D) \tag{4.97}$$

$$\Delta K_{T_2} = \left(f(d_C) - f(d_B)\right)\left(f(d_D) - f(d_A)\right). \tag{4.98}$$

We order the degrees of the four vertices (there are $4! = 24$ possible orderings), and without loss of generality, we require that

$$d_A \leq d_B \quad \text{and} \quad d_C \leq d_D \tag{4.99}$$

which reduces the number of possible orderings to 6, shown in Table 4.2. If both transformations result in an increase in the overall sensitivity, this means that the current arrangement of edges ((A, B) and (C, D)) is optimal. This happens when one of the following inequalities holds:

$$d_A \leq d_B \leq d_C \leq d_D \quad \text{or} \quad d_C \leq d_D \leq d_A \leq d_B. \tag{4.100}$$

Degree Order	ΔK_{T_1}	ΔK_{T_2}
$d_A \leq d_B \leq d_C \leq d_D$	≥ 0	≥ 0
$d_A \leq d_C \leq d_B \leq d_D$	≤ 0	≤ 0
$d_A \leq d_C \leq d_D \leq d_B$	≤ 0	≤ 0
$d_C \leq d_A \leq d_B \leq d_D$	≤ 0	≤ 0
$d_C \leq d_A \leq d_D \leq d_B$	≤ 0	≥ 0
$d_C \leq d_D \leq d_A \leq d_B$	≥ 0	≥ 0

Table 4.2: All possible orderings of the degrees of vertices A, B, C, and D, with $d_A \leq d_B$ and $d_C \leq d_D$, and the resulting difference in the overall crosstalk sensitivity among them for transformations T_1 and T_2 (see also Figure 4.9).

Transformation T_2 always yields a network with smaller sensitivity than T_1:

$$\begin{aligned}\Delta K_{T_3} =& M_2 - M_1 \\ =& f(d_A)f(d_D) + f(d_B)f(d_C) - f(d_A)f(d_C) - f(d_B)f(d_D) \\ =& (f(d_A) - f(d_B))(f(d_D) - f(d_C)) \\ \leq& 0\end{aligned} \tag{4.101}$$

by the assumption of the relative degrees of the pairs (A, B) and (C, D). More importantly,

$$\begin{aligned}M_2 =& M_0 + \Delta K_{T_2} \\ =& M_0 + \Delta K_{T_1} + \Delta K_{T_3} \\ =& M_1 + \Delta K_{T_3}.\end{aligned} \tag{4.102}$$

The last equation shows that the overall sensitivity of a graph does not depend on the order in which we apply any transformations. The difference in the crosstalk sensitivity is equal to the sum of individual differences, and it only depends on the

structure of the network before and after the performed changes. In order to minimize the overall sensitivity to crosstalk, we need to connect vertices with large degrees to other vertices with large degrees and similarly for vertices with small degrees. □

Lemma 29. *A degree sequence* $\mathbf{d} = [d_1, d_2, \dots d_N]$ *is a graphic sequence if and only if the degree sequence* $\mathbf{d}' = [d_2 - 1, d_3 - 1, \dots, d_{d_1+1} - 1, d_{d_1+2}, \dots, d_{N-1}, d_N]$ *is a graphic sequence.*

Proof. See [33]. □

Algorithm 2 Find the Connected Graph $\mathcal{G}$ with Minimum Sensitivity to Multiplicative Crosstalk

---

Require: A graphic sequence $\mathbf{d} = [d_1, \dots d_N]$ with $d_1 \leq d_2 \leq \dots \leq d_N$.
Ensure: The graph with the lowest sensitivity to crosstalk.

$\mathcal{G} \leftarrow$ Empty Graph with $\mathcal{V}(\mathcal{G})$=1,2,... N
$k \leftarrow 0$
while $\mathbf{d} \neq [0]$ **do**
  $k \leftarrow k + 1$
  $r \leftarrow$ First element of $\mathbf{d}$
  Connect vertex k with vertices $k+1, k+2, \dots, k+r$
  $\mathbf{d} \leftarrow [d_2 - 1, d_3 - 1, \dots, d_{r+1} - 1, d_{r+2}, \dots, d_{N-1}, d_N]$
end while

while $\mathcal{G}$ is not connected **do**
  Find the two components $\mathcal{F}, \mathcal{S}$ with the smallest average degrees , $\bar{d}_{\mathcal{F}} \leq \bar{d}_{\mathcal{S}}$.
  Pick vertices A, B with the largest degree in $\mathcal{F}$ with $d_A \leq d_B$.
  Pick vertices C, D with the largest degree in $\mathcal{S}$ with $d_C \leq d_D$.
  Add edges (A, C) and (B, D); Remove edges (A, B) and (C, D)
end while

Return $\mathcal{G}$

Theorem 20. *Algorithm 2 returns a connected network with the lowest possible overall sensitivity to multiplicative crosstalk, provided that a connected network with the given degree distribution exists.*

Proof. By construction, the first part of the algorithm produces the network with the most assortative structure possible, and therefore the network with the smallest crosstalk sensitivity. If this network is not connected, we need to carry out a rewiring that will have minimum impact to the overall sensitivity. It suffices to prove that the algorithm works when the graph has two components, since this procedure can easily be generalized for an arbitrary number of components. Suppose that the two components $\mathcal{F}$ and $\mathcal{S}$ have degree sequence $d_{\mathcal{F}} = [d_1, d_2, \ldots, d_c]$ and $d_{\mathcal{S}} = [d_{c+1}, d_{c+2}, \ldots, d_N]$, respectively. In order to make the graph connected, we need to apply the transformation T_2 of Figure 4.9, to rewire one edge in $\mathcal{F}$ and one edge in $\mathcal{S}$ so that they connect vertices in the different components. The difference of the overall crosstalk sensitivity is given in equation (4.98), where the four vertices have degrees

$$d_{c-1} \leq d_c \leq d_{c+1} \leq d_{c+2}. \tag{4.103}$$

The difference will be

$$\Delta_c = (f(d_{c+1}) - f(d_c))(f(d_{c+2}) - f(d_{c-1})), \tag{4.104}$$

which is the smallest positive difference while connecting the two components. Repeating this procedure for all the components of the graph, we end up with a connected graph, which has the minimum additional overall sensitivity to crosstalk. □

4.5 Conclusions

In this chapter, we focused on how network structure affects crosstalk in systems where there may be spurious connections among units that are not designed to interact with each other. We focused on two types of crosstalk interactions: additive and geometric. In additive crosstalk interactions, the crosstalk strength between any pair of vertices that are not connected in the network is the sum of their individual

affinities. We showed that in this case, crosstalk sensitivity only depends on the vertex degree distribution, regardless of the exact interconnection topology of the network.

The sensitivity to geometric crosstalk depends on the network topology, with the assortative network structure being the one that minimizes crosstalk, where vertices with large degrees form cliques with other vertices of large degrees. Finally, we have presented an algorithm which constructs a connected network that minimizes crosstalk when there are constraints regarding its degree distribution. The methods and algorithms described here can easily be generalized to take into account the specific details of the network function, by assigning weights to each crosstalk interaction, depending on how much it affects the system's normal operation.

Chapter 5

Noise Propagation in Biological and Chemical Reaction Networks

In this chapter, we describe how noise propagates through a network. Using stochastic calculus and dynamical systems theory, we study the network topologies that accentuate or alleviate the effect of random variance in the network for both directed and undirected graphs. Given a linear tree network, the variance in the output is a convex function of the poles of the individual nodes. Cycles create correlations which in turn increase the variance in the output. Feedforward and feedback have a limited effect on noise propagation when the respective cycles are sufficiently long. Crosstalk between the elements of different pathways helps reduce the output noise, but makes the network slower. Next, we study the differences between disturbances in the inputs and disturbances in the network parameters, and how they propagate to the outputs. Separating internal from external dynamics is important in order to understand fluctuations in various systems [4]. We find that these fluctuations have very different behaviors, and as a result, it is easy to analyze their origin in networked systems. Finally, we show how noise can affect the steady state of the system in chemical reaction networks with reactions of two or more reactants, each of which may be affected by independent or correlated sources.

5.1 Introduction and Overview

Noise is ubiquitous in nature, and virtually all signals carry some amount of random noise. In addition, even the simplest systems can be represented as a set of smaller entities interconnected with each other. There have been numerous studies on how noise affects specific functions (e.g. [54, 59] and references therein), but few of them have looked at how noise propagates in general networks, or the impact of network structure on the robustness of each system to noise. Although there is evidence that noise may degrade system performance, it is sometimes necessary for specific functions [23]. Robustness to noise and disturbances in general is something that biological and engineered systems depend upon for their proper function [39, 60]. It becomes even more important if we take into account that molecule concentrations in the cell might be carrying more than one signal through multiplexing [21, 61]. In this case, noise may have the potential of disrupting the network function in more than one way.

We present a new method to quantify the noise propagation in a system, and the vulnerability of each of its subsystems. We use results from graph theory and control systems theory to quantify noise propagation in networks, and use them to evaluate various network structures in terms of how well they filter out noise. We study how crosstalk can help suppress noise, when the noise sources are independent or correlated. We show that perturbations that depend on the state of the system (for example, feedback loops that are prone to noise or noisy degradation rates) have a fundamentally different effect on the system output, compared to noise in the inputs. Finally, we study noise propagation in chemical reaction networks where all reactants may introduce noise, and analytically find that noise correlations may affect the expected behavior of such systems.

5.2 White Noise Input

In the state space, when all parameters are deterministic and the input consists of a deterministic and a random component (white noise), then the system (2.7) is defined by the following equations:

$$\mathcal{S}: \begin{cases} dx = Axdt + B(u_t dt + \Sigma_t dW_t) \\ \\ y = Cx, \end{cases} \tag{5.1}$$

where $dW_t = W_{t+dt} - W_t$ is the standard vector Wiener process in the time interval $[t, t + dt)$ and u_t is a deterministic input. We will denote the value of a function f at time t as $f(t)$ or f_t interchangeably. The matrix Σ_t consists of nonnegative entries, possibly time-varying, each of which is proportional to the strength of the corresponding disturbance input. The only difference with the system (2.7) is that now the infinitesimal state difference dx depends not only on the current state and the deterministic input, but also on a random term $dW_t \sim \mathcal{N}(0, dt)$.

It should be noted that the fraction dW_t/dt does not exist as $dt \to 0$, so dividing both sides of equation (5.1) by dt would not make sense mathematically. This notation helps us to intuitively understand the effect of randomness in the system, when we know how its state is affected by the input noise, and enables us to easily generalize these results when the random term is a product of many noise sources. The different Wiener processes in (5.1) may be correlated with each other but since each input may consist of a weighted sum of all of the different processes through multiplication by matrix Σ_t, without loss of generality, we may assume that they are independent.

The output of the system is the superposition of the deterministic output, and

the response to the random input:

$$\begin{aligned} y(t) &= \int_{-\infty}^{t} h(t-s)(u(s)ds + \Sigma_s dW_s) \\ &= \int_{-\infty}^{t} h(t-s)u(s)ds + \int_{-\infty}^{t} h(t-s)\Sigma_s dW_s. \end{aligned} \tag{5.2}$$

The expected value for the output, according to equation (2.12) will be

$$\begin{aligned} \mathbb{E}[y(t)] &= \int_{-\infty}^{t} h(t-s)\mathbb{E}\left[u(s)ds + \Sigma_s dW_s\right] \\ &= \int_{-\infty}^{t} h(t-s)u(s)ds, \end{aligned} \tag{5.3}$$

since Brownian motion is a martingale [44]. Applying equation (2.13) when the input is white noise, the covariance matrix can be written as

$$\mathbb{V}[y] = \int_{-\infty}^{t}\int_{-\infty}^{t} h(t-r)\mathbb{E}\left[dW_r \Sigma_r \Sigma_s^T dW_s^T\right] h^T(t-s). \tag{5.4}$$

The inputs are assumed to be white noise processes, so the covariance among all of them is nonzero only if they take place during the same interval, and in that case, the covariance is proportional to its length.

$$\begin{aligned} \mathbb{V}[y] &= \int_{-\infty}^{t}\int_{-\infty}^{t} h(t-r)\left(\Sigma_r\sqrt{dr}\delta(r-s)\sqrt{ds}\Sigma_s^T\right) h^T(t-s) \\ &= \int_{-\infty}^{t} h(t-s)\cdot V(s)\cdot h^T(t-s)ds, \end{aligned} \tag{5.5}$$

where $V(s) = \Sigma_s\Sigma_s^T$ is the covariance matrix of the input random vector. For the linear time invariant system (2.7) and white noise inputs of constant variance $V(s)$ is a constant matrix, and we can write

$$\mathbb{V}[y] = \int_{-\infty}^{t} (Ce^{A(t-s)}B)\cdot V\cdot (Ce^{A(t-s)}B)^T ds = C\left(\int_0^{+\infty} e^{Ax}BVB^T e^{A^T x}dx\right)C^T. \tag{5.6}$$

The mean and the variance of the output signal in the steady state can be written as a function of the Fourier transforms of the input signal and the network transfer function. From equation (2.12)

$$\mathbb{E}\left[y(t)\right] = \mathbb{E}\left[\int_{-\infty}^{t} h(t-s)u(s)\right] ds = h(t) * \mathbb{E}\left[u(t)\right] \tag{5.7}$$

where $f(t) * g(t)$ denotes the convolution of two functions $f(t)$ and $g(t)$ given that it exists. When the input is constant with time, the expected value of the input is constant as well ($\mathbb{E}[u(t)] = \mu_x$) and the last expression can be simplified to

$$\mathbb{E}[y] = \mu_x \int_0^{+\infty} h(u)du = \mu_x H(0). \tag{5.8}$$

If the input itself is not known, but its frequency content can be estimated, we can find the variance of the output using Parseval's theorem:

$$\begin{aligned}
\mathbb{V}[y] = \mathbb{E}[y \cdot y^T] &= \lim_{t\to\infty} \int_{-\infty}^{t} y(t)y^T(t)dt \\
&= \int_{-\infty}^{+\infty} |Y(f)|^2 df = \int_{-\infty}^{+\infty} Y(f) \cdot Y^*(f)df \\
&= \int_{-\infty}^{+\infty} H(f)X(f)X^*(f)H^*(f)df.
\end{aligned} \tag{5.9}$$

More generally, if we know the autocorrelation function of the random processes in the input, we may find the expected autocorrelation in the output, and then estimate the output variance:

$$\begin{aligned}
R_y(\tau) &= \int_{-\infty}^{+\infty} S_y(f)\cos(2\pi f\tau)df \\
&= \int_{-\infty}^{+\infty} |H(f)|^2 S_x(f)\cos(2\pi f\tau)df \\
&= \int_{-\infty}^{+\infty} |H(f)|^2 \left(\int_{-\infty}^{+\infty} R_x(u)\cos(2\pi fu)du\right)\cos(2\pi f\tau)df.
\end{aligned} \tag{5.10}$$

We will be focusing on Wiener processes exclusively, because this is the most general approach for sums of random disturbances. According to the Central Limit Theorem, the sum of a large number of independent identically distributed random variables with finite mean and variance always approaches the normal distribution (see also equation (2.4)). The only assumption in the case of additive disturbances is that the inputs at every time are sums of independent random variables from an arbitrary distribution with finite standard deviation. This is a reasonable assumption in most settings. For example, in biology the Poisson distribution is frequently used to model random disturbances [54]. The Poisson distribution can be well approximated by a Gaussian when the event rate is greater than 10, and the same can be said for small sums of Poisson random variables [62]. When the input disturbance at each time is correlated with the disturbances during earlier times, the correlation structure can be emulated by passing white noise through a filter that produces it. Lastly, if noise cannot be expected to have equal frequency content for all frequencies up to infinity, we can still use white noise as an input, which is then passed through a filter with zero response for all the frequencies outside the desired range.

5.3 Tree Networks

Tree networks are a special case of networks where there is a unique path among every pair of vertices. In other words, there are no cycles, which makes the analysis easier. Many natural networks have been found to be locally tree-like [34]. When analyzing the behavior of a network around an equilibrium point, or if the network is linear, then the analysis can be significantly simplified. Since there is a unique path from any vertex to another, it suffices to analyze path networks, which consist of all their vertices connected in series. For each output, the total response of the system is the superposition of the signals caused for all the individual inputs. First, we will show that the order of the nodes in the network does not matter in the case of

linear pathways. Then, we will find the variance of a linear path graph assuming that every node is a first-order filter. The result can easily be generalized for arbitrary tree graphs. Finally, we are going to find the optimal placement of poles to minimize the output variance.

5.3.1 Output Variance of Linear Pathways

Lemma 30. *The noise response of a linear pathway is independent of the relative position of its nodes.*

Proof. Without loss of generality, we can assume that the linear pathway has one input and one output. Otherwise, since the system is linear, we can repeat the process each time considering only the respective subtree. Under the last assumption, the output is the state of the last node, and all inputs affect only the first node. From equation (5.9):

$$\begin{aligned}
\mathbb{V}[y] &= \int_{-\infty}^{+\infty} H(f)X(f)X^*(f)H^*(f)df \\
&= \int_{-\infty}^{+\infty} H(f)(X_1(f)+\ldots+X_n(f))(X_1^*(f)+\ldots+X_n^*(f))H^*(f)df \\
&= \sum_{k=1}^{n}\sum_{m=1}^{n}\int_{-\infty}^{+\infty} X_k(f)X_m^*(f)H(f)H^*(f)df \\
&= \sum_{k=1}^{n}\sum_{m=1}^{n}\int_{-\infty}^{+\infty} X_k(f)X_m^*(f)(h_1(f)\cdot\ldots\cdot h_N(f))(h_n^*(f)\cdot\ldots\cdot h_1^*(f))df \\
&= \sum_{k=1}^{M}\sum_{m=1}^{M}\int_{-\infty}^{+\infty} X_k(f)X_m^*(f)\prod_{n=1}^{N}|h_n(f)|^2df.
\end{aligned} \tag{5.11}$$

It is evident that we can interchange the transfer functions inside the product in the integral, without changing its value. □

Assume that we have a linear pathway such that the system is described by equa-

tion (2.7) and the dynamical and input matrices are

$$A = \begin{bmatrix} -d_1 & 0 & 0 & \dots & 0 \\ f_2 & -d_2 & 0 & \dots & 0 \\ 0 & f_3 & -d_3 & \dots & 0 \\ 0 & 0 & \dots & f_N & -d_N \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad C^T = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

For simplicity, we assume that there is only one noise source and only one output, but since there are no cycles, there is a unique path from each node to every other, which means we can use the result for a linear pathway repeatedly, in order to find the total variance. The variance is independent of the deterministic input that is applied to the pathway. Using equation (5.6), and after performing all calculations, the variance at the output will be

$$\mathbb{V}_{out} = \left(\prod_{u=1}^{N-1} f_u \right) \left(\sum_{k=1}^{N} \sum_{m=1}^{N} \frac{1}{(d_k + d_m) \prod\limits_{a=1, a\neq k}^{N} (d_k - d_a) \prod\limits_{b=1, b\neq m}^{N} (d_m - d_b)} \right). \quad (5.12)$$

The expression above holds even if there exist two vertices a and b such that their reaction rates are equal, according to the next lemma.

Lemma 31. *The output variance of a linear pathway is always finite, even if some of the reaction rates are equal.*

Proof. We pick two rates d_x and d_y and show that $\mathbb{V}_{out}$ does not depend on their

difference. If we denote

$$T_{k,m} = \frac{1}{(d_k + d_m) \prod_{a=1, a\neq k}^{N} (d_k - d_a) \prod_{b=1, b\neq m}^{N} (d_m - d_b)}, \tag{5.13}$$

the difference $d_x - d_y$ appears only in the terms $T_{x,x}, T_{x,y}, T_{y,x}$, and $T_{y,y}$. Their sum T_{x-y} is equal to

$$\begin{aligned} T_{x-y} &= T_{x,x} + T_{x,y} + T_{y,x} + T_{y,y} \\ &= \frac{1}{2d_x(d_x - d_y)^2 \prod_{s\neq x,y} (d_x - d_s)} + \frac{1}{2d_y(d_y - d_x)^2 \prod_{s\neq x,y} (d_y - d_s)} \\ &\quad - \frac{2}{(d_x + d_y)(d_y - d_x)^2 \prod_{s\neq x,y} (d_y - d_s)}. \end{aligned} \tag{5.14}$$

We set

$$P_x = \prod_{s\neq x,y} (d_x - d_s) \quad \text{and} \quad P_y = \prod_{s\neq x,y} (d_y - d_s) \tag{5.15}$$

so that sum above can be written as

$$T_{x-y} = \frac{d_x(d_x + d_y)P_x^2 + d_y(d_x + d_y)P_y^2 - 4d_x d_y P_x P_y}{2d_x d_y(d_x + d_y)(d_y - d_x)^2 P_x^2 P_y^2}. \tag{5.16}$$

Expanding the nominator of T_{x-y} and grouping the relevant terms together:

$$\begin{aligned} T_{x-y} &= \frac{d_x^2 P_x^2 + d_x d_y P_x^2 + d_x d_y P_y^2 + d_y^2 P_y^2 - 4d_x d_y P_x P_y}{2d_x d_y(d_x + d_y)(d_y - d_x)^2 P_x^2 P_y^2} \\ &= \frac{\left(d_x^2 P_x^2 - 2d_x d_y P_x P_y + d_y^2 P_y^2\right) + d_x d_y \left(P_x^2 - 2P_x P_y + P_y^2\right)}{2d_x d_y(d_x + d_y)(d_y - d_x)^2 P_x^2 P_y^2} \\ &= \frac{(d_x P_x - d_y P_y)^2 + d_x d_y (P_x - P_y)^2}{2d_x d_y(d_x + d_y)(d_y - d_x)^2 P_x^2 P_y^2}. \end{aligned} \tag{5.17}$$

It is easy to see that both terms in the nominator of the last fraction have a factor

of order $(d_y - d_x)^2$, and the lemma is proved. □

5.3.2 Optimization of Linear Pathways

In this subsection, we will try to find the parameters of a linear pathway such that the output variance is minimized.

Lemma 32. *Assume that the same noise source is applied to two different pathways with impulse responses $h_1(t)$ and $h_2(t)$, respectively. The covariance of the signals in their output will be equal to*

$$C(\tau) = \lim_{t\to\infty} \mathbb{E}\left[y_1(t)y_2(t+\tau)\right] = \int_0^\infty h_1(r)h_2(r+\tau)dr. \tag{5.18}$$

Proof. The two outputs $y_1(t)$ and $y_2(t)$ are

$$y_1(t) = \int_{-\infty}^t h_1(t-x)dW_x \quad \text{and} \quad y_2(t) = \int_{-\infty}^t h_2(t-y)dW_y \tag{5.19}$$

where W is the Wiener process that drives both systems simultaneously. The expected value of the product of the first and a delayed version of the second is

$$\begin{aligned} C(\tau) &= \mathbb{E}\left[\int_{-\infty}^t h_1(t-x)dW_x \cdot \int_{-\infty}^t h_2(t+\tau-y)dW_y\right] \\ &= \int_{-\infty}^t \int_{-\infty}^{t+\tau} h_1(t-x)h_2(t+\tau-y)\mathbb{E}\left[dW_x dW_y\right] \\ &= \int_{-\infty}^t h_1(t-s)h_2(t+\tau-s)ds \\ &= \int_0^\infty h_1(r)h_2(r+\tau)dr. \end{aligned} \tag{5.20}$$

□

Corollary 29. *Assume that noise from a single noise source with standard deviation σ enters a network, and propagates through N independent pathways to reach the output. If the impulse response of each of the independent pathways is $h_1(t), h_2(t), \ldots, h_N(t)$,*

respectively, the mean of the output y will be zero, and its variance equal to

$$V_{out} = \sigma^2 \int_0^\infty \left(\sum_{k=1}^{N} a_k h_k(x) \right)^2 dx. \tag{5.21}$$

Proof. The output vertex will receive a weighted sum of the outputs of the two independent pathways

$$z(t) = \sum_{k=1}^{N} a_k y_k(t). \tag{5.22}$$

Its expected value is equal to zero at all times:

$$\begin{aligned} \mathbb{E}[z(t)] &= \mathbb{E}\left[\sum_{k=1}^{N} a_k y_k(t) \right] = \sum_{k=1}^{N} \mathbb{E}\left[a_k y_k(t) \right] \\ &= \sum_{k=1}^{N} a_k \int_{-\infty}^{t} h_k(t-x) \sigma \mathbb{E}\left[dW_x \right] \\ &= 0. \end{aligned} \tag{5.23}$$

The variance is equal to:

$$\begin{aligned} \mathbb{V}_y &= \mathbb{E}[z^2(t)] \\ &= \mathbb{E}\left[\left(\int_{-\infty}^{t} \sum_{k=1}^{N} a_k h_k(t-x) dW_x \right) \cdot \left(\int_{-\infty}^{t} \sum_{k=1}^{N} a_k h_k(t-y) dW_y \right) \right] \\ &= \int_{-\infty}^{t} \int_{-\infty}^{t} \left(\sum_{k=1}^{N} a_k h_k(t-x) \right) \cdot \left(\sum_{k=1}^{N} a_k h_k(t-y) \right) \mathbb{E}\left[dW_x dW_y \right] \\ &= \int_{-\infty}^{t} \sigma^2 \left(\sum_{k=1}^{N} a_k h_k(t-s) \right)^2 ds \\ &= \sigma^2 \int_0^\infty \left(\sum_{k=1}^{N} a_k h_k(x) \right)^2 dx. \end{aligned} \tag{5.24}$$

□

Suppose we have a linear pathway with each element representing a single-pole

linear filter, and we need to pick the position of the poles such that the variance in the output is minimized. The next lemma shows an easy way to find them if all the vertices are identical and subject to symmetric constraints.

Definition 8. *A symmetric multivariable function $f : \mathbb{R}^n \to \mathbb{R}$ is a function for which $f(x) = f(\pi(x))$ where $\pi(x)$ is an arbitrary permutation of the input vector x.*

Lemma 33. *Assume that a symmetric multivariable function $f : \mathbb{R}^n \to \mathbb{R}$ is nowhere constant and has a sign definite Hessian matrix. Then it has a unique extremum under symmetric constraints, such that all the inputs are equal.*

Proof. Since the Hessian has the same sign everywhere, the function f is convex or concave. Without loss of generality, we analyze the case where f is convex. Suppose that the extremum of the function f is equal to f^*, and the argument that achieves this is x^*. Further assume that $\min(x^*) = m$ and $\max(x^*) = M$ are the minimum and maximum elements of the input x^*, respectively. Since f is symmetric,

$$f(m, M, x_3^*, \ldots, x_n^*) = f(M, m, x_3^*, \ldots, x_n^*) = f^* \tag{5.25}$$

where the arguments still satisfy the symmetric constraints. But since f is strictly convex, every convex combination of these values will be

$$\begin{aligned} f(a, b, x_3, \ldots x_n) &\leq tf(m, M, x_3, \ldots, x_n) + (1-t)f(M, m, x_3, \ldots, x_n) \\ &= tf^* + (1-t)f^* \\ &= f^*. \end{aligned} \tag{5.26}$$

Generalizing the last argument, it is straightforward to see that

$$f(x_1, x_2, \ldots x_n) = f^* \quad \text{for every} \quad m \leq x_1, x_2, \ldots x_n \leq M. \tag{5.27}$$

Therefore, $f(x)$ needs to be constant in that area, which contradicts the assumption that the function has sign definite Hessian. □

When the constraints are convex but not necessarily symmetric, then we can use the Lagrangian to find the optimal parameters. Considering a linear pathway, and assuming that the input is white noise, if the poles of the different nodes are placed at $a_1, a_2, \ldots, a_N$, the total variance in the output is equal to (see equation (5.9)):

$$\begin{aligned}\mathbb{V}_{out}(a_1, a_2, \ldots a_N) &= \frac{1}{2\pi}\int_{-\infty}^{+\infty}\left|\frac{1}{j\omega + a_1}\right|^2 \cdot \left|\frac{1}{j\omega + a_2}\right|^2 \cdots \left|\frac{1}{j\omega + a_N}\right|^2 d\omega \\ &= \frac{1}{2\pi}\int_{-\infty}^{+\infty}\frac{1}{\omega^2 + a_1^2}\cdot\frac{1}{\omega^2 + a_2^2}\cdots\frac{1}{\omega^2 + a_N^2}d\omega.\end{aligned} \tag{5.28}$$

The function $\mathbb{V}_{out}$ is convex with respect to all its arguments $a_1, a_2 \ldots a_N$, as an (infinite) sum of products of convex functions. Consequently, it has a unique minimum under convex constraints. The Lagrangian of the function for $\mathbb{V}_{out}$ is

$$\mathcal{L}(a_1, a_2, \ldots a_N) = \frac{1}{2\pi}\int_{-\infty}^{+\infty}\frac{1}{\omega^2 + a_1^2}\cdot\frac{1}{\omega^2 + a_2^2}\cdots\frac{1}{\omega^2 + a_N^2}d\omega - \lambda g(a_1, a_2, \ldots, a_N). \tag{5.29}$$

Differentiating with respect to a_k, under the Leibnitz integral rule:

$$\frac{\partial\mathcal{L}}{\partial a_k} = \frac{1}{2\pi}\int_{-\infty}^{+\infty}\frac{1}{\omega^2 + a_1^2}\cdots\frac{-2a_k}{(\omega^2 + a_k^2)^2}\cdots\frac{1}{\omega^2 + a_N^2}d\omega = \lambda\frac{\partial g(a_1, \ldots, a_N)}{\partial a_k} \tag{5.30}$$

for every k. Differentiating with respect to all the parameters will give us N equations, and we have one more equation by requiring $g(a_1, \ldots, a_N) = 0$. So we can solve the system of $N + 1$ equations and $N + 1$ unknowns $\lambda, a_1, \ldots a_N$, which is guaranteed to have a unique solution as all functions are convex.

In conclusion, we can find the unique minimum of the variance of a linear pathway, when each node is a linear filter with a real negative pole. Given that a linear tree network with independent noise inputs can be decomposed to many linear pathways, this method can be applied to any arbitrary tree network.

5.4 Feedforward and Feedback Cycles

In a serial pathway where each vertex acts as a filter, the output of each node has a different frequency content as the noise propagates through the network, being filtered at each step. The variance decreases when we move further from the noise source (Figure 5.1). As the serial pathway becomes longer, the input and the output

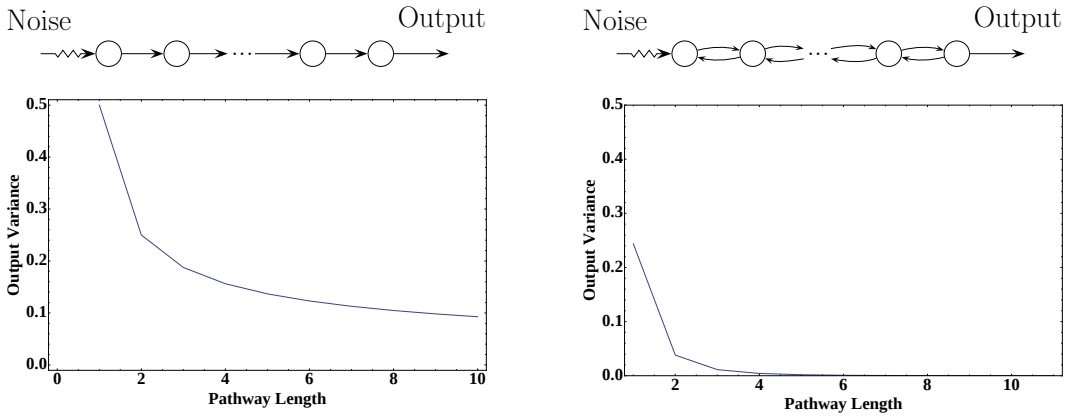
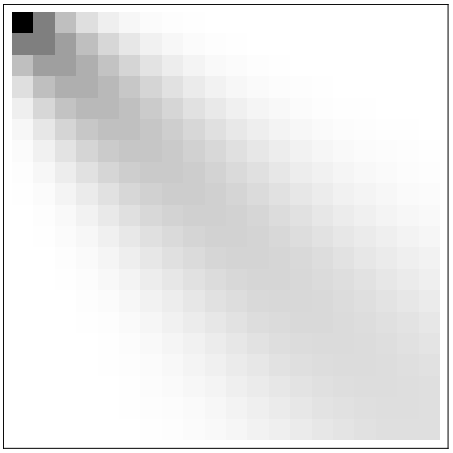


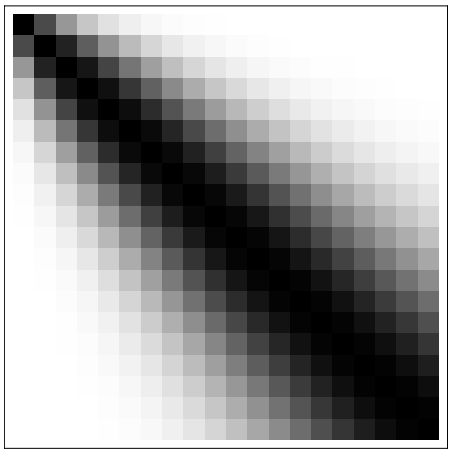
Figure 5.1: Variance of the output of a unidirectional and a bidirectional serial pathway as a function of the pathway length. All nodes are assumed to be identical single-pole filters. In the unidirectional pathway, each node is affected only by the node immediately preceding it, whereas in the bidirectional pathway each intermediate vertex is receiving input from the node preceding and the node succeeding it. The bidirectional pathway is much more efficient in filtering out noise. The variance for both pathways decreases with the pathway length. The bidirectional pathway has variance very close to zero even when it is relatively short.

become less correlated since their distance increases. In addition, every node changes the phase of its inputs, which also contributes to the decreased correlation. Therefore, applying negative feedback or setting up a feedforward cycle can only have a measurable effect if the cycle length is relatively small. Figure 5.2 shows the covariances and correlations among the vertices of two simple linear pathways, one unidirectional and one bidirectional, as they are depicted in Figure 5.1.

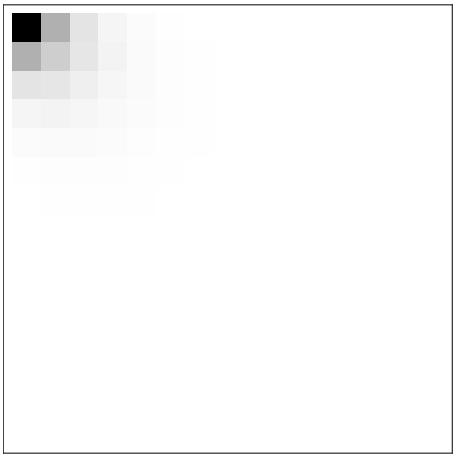
Feedforward cycles can significantly increase the effect of noise in the system. There are two reasons for this: First, the noise can now reach more vertices since



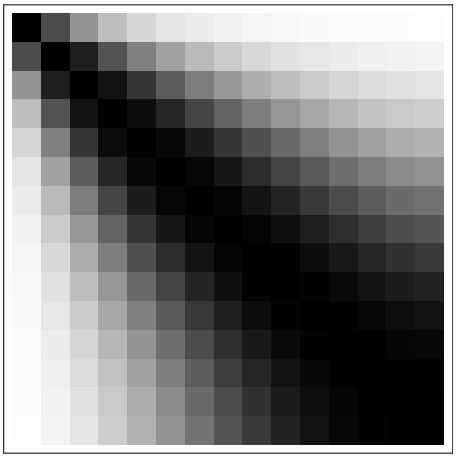
(a) Unidirectional Pathway Covariance Matrix



(b) Unidirectional Pathway Correlation Matrix



(c) Bidirectional Pathway Covariance Matrix



(d) Bidirectional Pathway Correlation Matrix

Figure 5.2: Covariance and correlation among all pairs of nodes in a linear pathway. Every square (x, y) in the matrices above corresponds to the value of their correlation $R_{x,y}(\tau = 0)$ of nodes of distance x and y from the origin, $0 \leq x, y \leq N - 1$. The larger the correlation, the darker the respective square. As the distance $|x - y|$ among the nodes increases, their covariance and correlation decreases. The covariance among nodes of the same distance in the unidirectional pathway decreases, and the correlation among them increases towards the end. The covariance of the nodes in the bidirectional pathway is essentially zero within a small distance, and the correlation is larger even when the distance is relatively large.

the average distance among nodes decreases, and second, every node now receives the same disturbance from at least two different paths, and the two signals are correlated, contributing to larger variance. An example is shown in Figure 5.3, where we compare the average variance of two systems whose only difference is the connection between the first and the last node. Both networks receive the same inputs, but in the cycle network, the variance is much larger. The variance is even larger when there is correlation among the noise inputs to different nodes.

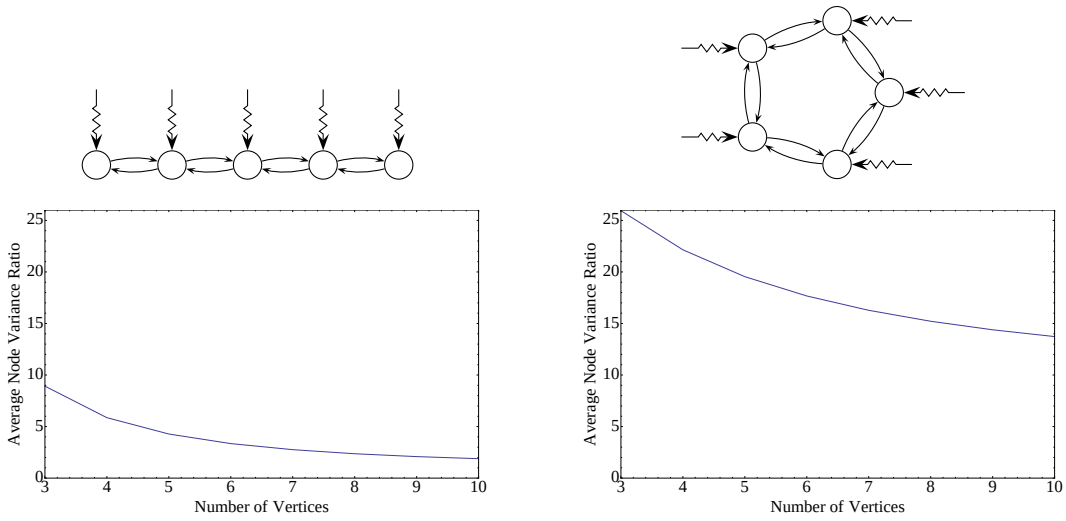


Figure 5.3: Average variance of all nodes in a network in a cycle as compared to an identical network without the feedback loop. Every node has a noise input which is then spread through the network. The average variance of all the nodes for both the cycle is normalized by the variance of the respective serial pathway. The variance cycle is always much larger than the variance of the simple serial pathway when the noise inputs for each node are uncorrelated (bottom left). The ratio becomes even larger when the inputs are correlated (bottom right).

The effect of cycles on the output noise can be reduced if we make sure that each independent pathway also changes the phase of its input by different amounts. Different phases in the output (for at least a relatively large frequency spectrum) will ensure that the various frequencies partially cancel each other out, reducing the output variation. When a pathway significantly reduces the frequency content, or has

small gain for most frequencies, then correlations do not play a significant role. This behavior is clearly shown in Figure 5.4 for a unidirectional cycle and in Figure 5.5 for a bidirectional cycle. Phase shifts in a pathway are equivalent to time delays, as we will see in the next section.

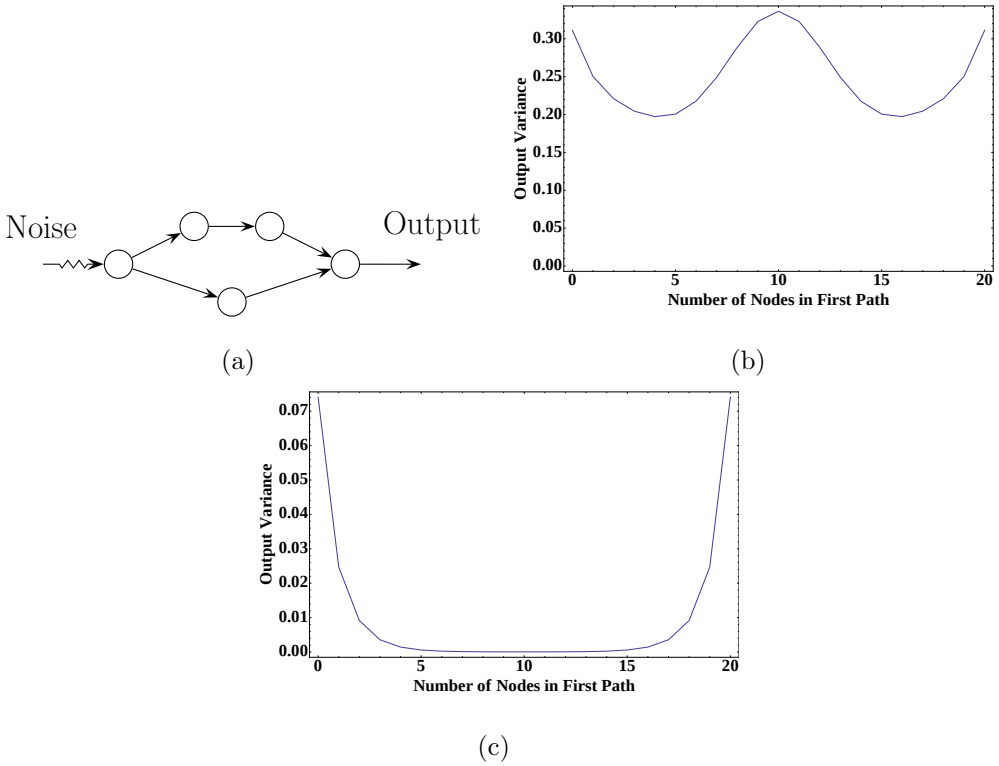


Figure 5.4: A network consisting of a feedforward cycle and the corresponding noise strength in its output. If the nodes of the network have poles with relatively small absolute values, then the output variance may be larger than the variance in the intermediate nodes. A fixed number of identical nodes is divided into two pathways, whose output is combined in the output node. If the number of nodes is similar in both pathways, then their outputs are highly correlated, and when combined produce large random swings. This does not happen when the poles of each node have a large negative real part (right). In the first case, the poles are placed at $a = -1$, whereas in the second the poles are placed at $a = -1.5$.

Similarly, negative feedback carefully applied to a network contributes to better disturbance rejection [6]. When the disturbance is white noise, the effect of feedback is smaller when the feedback cycle is long. As we move towards the end of the pathway, the covariance of nodes of a given distance decreases but the correlation of nodes of

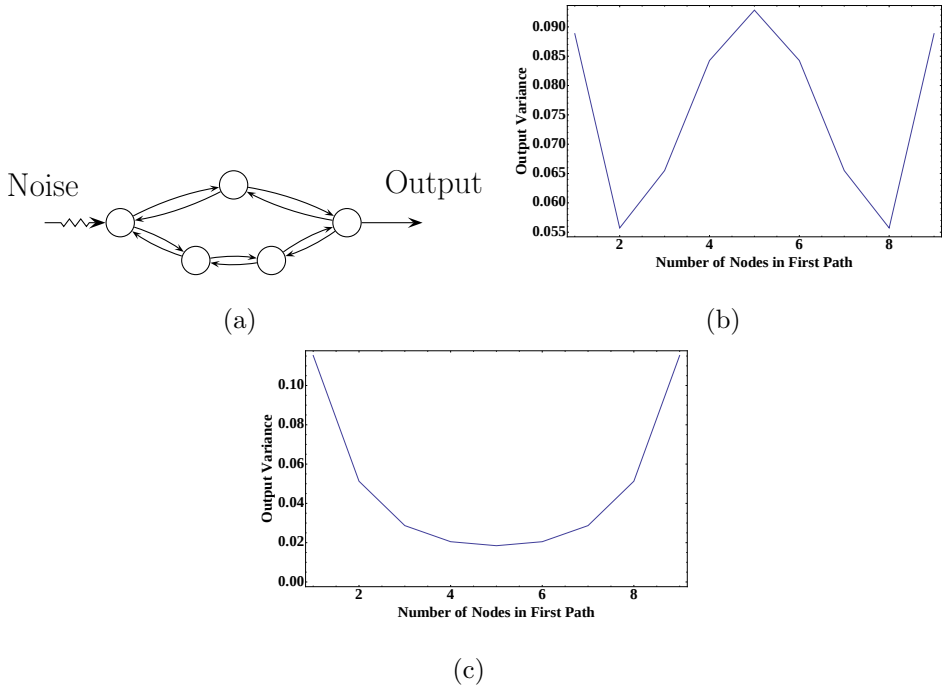
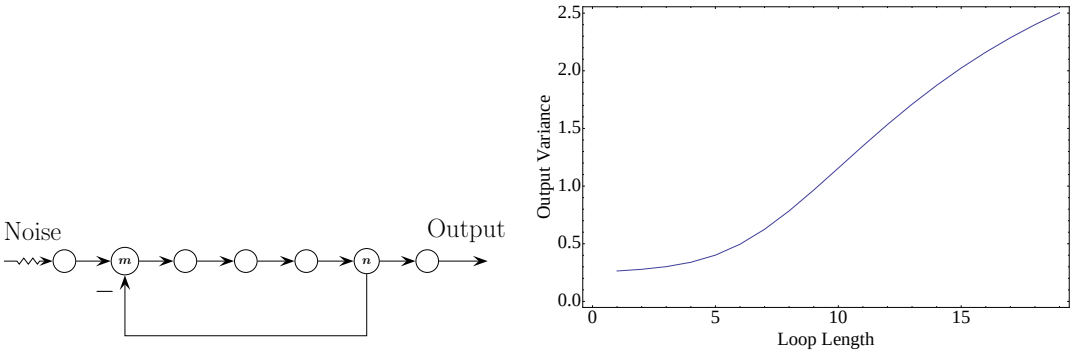


Figure 5.5: Correlations increase the variance in bidirectional networks. If the outputs of two pathways that are correlated are combined, then the output has relatively large variance. Here, a single output receives input from two pathways of different lengths, which consist of identical nodes. Bidirectional pathways filter noise very effectively as shown before, and the output variance is still small.

a given distance increases. The last observation is easily justified taking into account that each new node acts as a filter, and the output of each node in a pathway will tend to have very similar frequency content the more filters it has gone through.

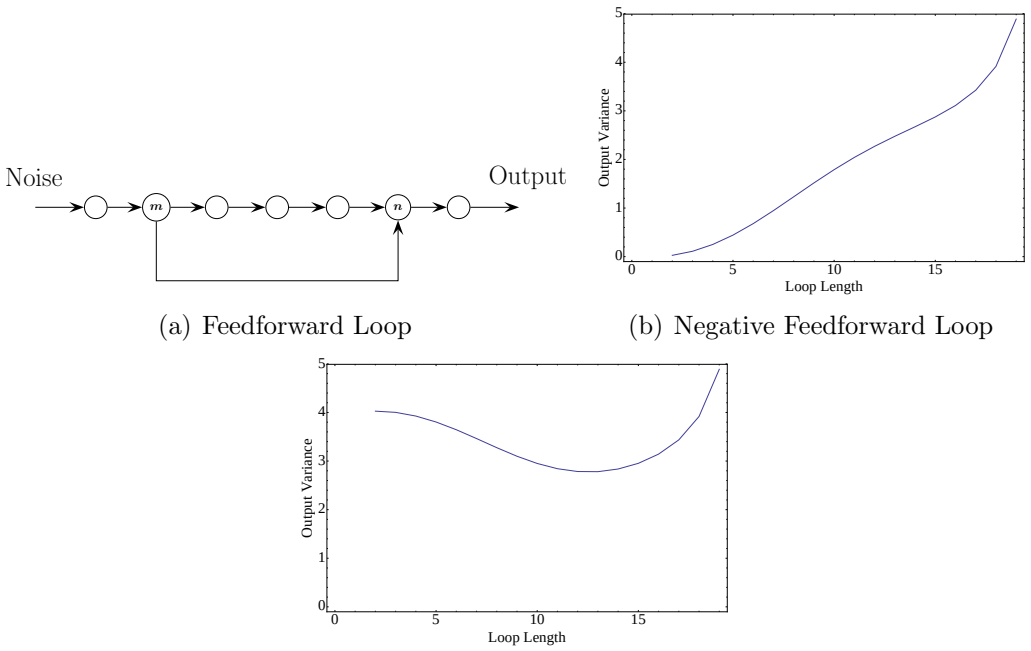
This hints to the fact that feedback cycles have limited utility when applied to long pathways. Figure 5.6 shows the variance of the output after we apply negative feedback to a linear pathway. The darkness of each element (m, n) in the upper triangular matrix shows the standard deviation of the pathway output when we apply feedback from node n to node m. As one would expect, the effect of feedback is directly proportional to the correlation between the source and target vertices. The same holds for feedforward loops, both positive and negative.

In the case of negative feedforward loop, the variance in the output increases as the loop length increases. When the feedforward interaction is positive, the variance



(a) Feedback Topology Figure (b) Output Variance

Figure 5.6: A serial pathway with a unit feedback loop. The output variance of a pathway depends on the length of the loop, all else being equal. The variance decreases as the length of the feedback loop becomes smaller, and vice versa.



(a) Feedforward Loop (b) Negative Feedforward Loop

(c) Positive Feedforward Loop

Figure 5.7: Output variance of a linear pathway when the input is white noise, and we add a negative (left) or positive (right) feedforward loop starting from the first vertex. For the positive loop, the variance is largest when we connect nearby vertices (large correlation) or we connect an early vertex to the end of the pathway, since it has a large variance that is transmitted directly to the output without being further filtered.

decreases at first, since the correlation among the different states also decreases, but then goes up, partly because when it affects a node towards the end of the pathway, it does not pass through successive filters, so the variance remains high (Figure 5.7).

5.4.1 Delayed Feedforward and Feedback Cycles

Adding delay to the interactions among any nodes in a network driven by noise decreases their correlation. Consequently feedforward or feedback cycles will have a smaller effect. The covariance of a white noise process with a delayed version of the same signal can be computed the same way as in equation (5.5):

$$\begin{aligned}
\mathbb{V}_\tau[y] &= \lim_{t\to\infty} \mathbb{E}[y(t)y(t+\tau)] \\
&= \lim_{t\to\infty} \mathbb{E}\left[\left(\int_{-\infty}^{t} h(t-r)\Sigma_r dW_r\right)\left(\int_{-\infty}^{t+\tau} h(t+\tau-s)\Sigma_s dW_s\right)^T\right] \\
&= \lim_{t\to\infty} \int_{-\infty}^{t}\int_{-\infty}^{t+\tau} h(t-s)\Sigma_r \mathbb{E}\left[dW_r dW_s^T\right]\Sigma_s^T h^T(t+\tau-s) \\
&= \lim_{t\to\infty} \int_{-\infty}^{t}\int_{-\infty}^{t+\tau} h(t-r)\Sigma_r\sqrt{dr}\,\delta(s-r)\sqrt{ds}\,\Sigma_s h^T(t+\tau-s) \\
&= \lim_{t\to\infty} \int_{-\infty}^{t} h(t-s)V_s h^T(t+\tau-s)ds \\
&= \lim_{t\to\infty} \int_{0}^{t} h(t-s)V_s h^T(t+\tau-s)ds.
\end{aligned} \tag{5.31}$$

If the system is causal, linear, and time invariant, and the disturbance is white noise of constant strength added to the input,

$$\mathbb{V}_\tau[y] = \int_0^\infty h(s)Vh^T(s+\tau)ds. \tag{5.32}$$

As a specific example, if the impulse response is $h(t) = Ce^{At}B$ and the covariance matrix is constant:

$$\begin{aligned}\mathbb{V}_\tau[y] &= \int_0^\infty Ce^{As}BVB^Te^{(s+\tau)A^T}C^Tds \\ &= C\left(\int_0^\infty e^{As}BVB^Te^{sA^T}ds\right)e^{\tau A^T}C^T.\end{aligned} \tag{5.33}$$

Note that the last equation is similar to equation (5.6), except for the exponential delay term in the end. We assume that the dynamical matrix A has negative eigenvalues, otherwise the system is not stable. If the delay is $\tau > 0$,

$$\begin{aligned}\|\mathbb{V}_\tau\| &= \|C\left(\int_0^\infty e^{As}BVB^Te^{A^Ts}ds\right)e^{A^T\tau}C^T\| \\ &\leq \|C\left(\int_0^\infty e^{As}BVB^Te^{A^Ts}ds\right)C^T\| \cdot \|e^{A^T\tau}\| \\ &< \|C\left(\int_0^\infty e^{As}BVB^Te^{A^Ts}ds\right)C^T\| \\ &= \|\mathbb{V}_0\|.\end{aligned} \tag{5.34}$$

The matrix norm used here is the first-order elementwise norm, since we are usually interested in the average variance of all parts of the network:

$$\|M\| = \sum_{i=1}^{N}\sum_{j=1}^{N}|m_{i,j}|. \tag{5.35}$$

If we only know the autocorrelation function of the disturbance, we can compute the output variance by moving to the frequency domain:

$$\begin{aligned}R_y(\tau) &= \int_{-\infty}^{+\infty} S_y(f)\cos(2\pi f\tau)df \\ &= \int_{-\infty}^{+\infty} |H(f)|^2 S_x(f)\cos(2\pi f\tau)df \\ &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} |C(j\omega I - A)^{-1}B|^2\left(\int_{-\infty}^{+\infty} R_x(u)\cos(\omega u)du\right)\cos(\omega\tau)d\omega.\end{aligned} \tag{5.36}$$

The shape of the autocorrelation function is a good indicator of how a feedback or feedforward loop will affect the output variation. A correlation function that quickly goes to zero as τ increases shows that the feedback cycle will not decrease the variance of the output by a lot. Conversely, a random signal with a long correlation structure can be easily filtered out by applying an appropriate feedback mechanism.

5.4.2 Minimization of the Average Vertex Variance

In a general network, signals are propagated from a node to its neighbors. Every vertex receives a filtered version of the noise signal, since each node acts as a single pole filter. We will assume that the poles are real, and proportional to the degree of each vertex, given that each node receives input proportional to the differences of concentrations among its neighbors and itself, and that nodes that interact with many others have proportionally large degradation rates. In this case, we can model the dynamics of a first-order linear network through its Laplacian matrix. In such a network, the state of each node x_k follows the differential equation

$$\frac{dx_k}{dt} = \sum_{m \in \mathcal{N}_k} a_{km}(x_k - x_m), \tag{5.37}$$

where $a_{km} > 0$ for every $k, m \in \mathcal{V}(\mathcal{G})$. The Laplacian of a matrix has been used to model a wide range of systems, including formation stabilization for groups of agents, collision avoidance of swarms, and synchronization of coupled oscillators [51]. It can also be used in biological and chemical reaction networks, if the degradation rate of each species is equal to the sum of the rates with which it is produced.

In this section, we will model the dynamics of each network with the Laplacian matrix, where each node is affected by a noise source which is independent of all other nodes, but has the same standard deviation. Given that each vertex contributes equally to the overall noise measure of the graph, and since the noise entering each node propagates towards all its neighbors, we can use Lemma 33 to see that the

degrees of the network vertices have to be as similar as possible (see also [11] and [33]). In addition, Figure 5.3 shows that the cycles need to be as long as possible in order to avoid any correlations of signals through two different paths. For longer cycles, the noise inputs go through more filters before they are combined. Lastly, the phase shift is larger for all their frequencies, which reduces their correlation.

On the other hand, there are bounds on how long a cycle can be given the network's order and size, as shown in Chapter 3. Networks with long cycles tend to have a large radius and larger average distance, which makes noise harder to propagate, having to pass through many filters. By the same token, networks with a large clustering coefficient will tend to be less immune to noise, since they tend to create cliques or densely connected subnetworks, which facilitate noise propagation, especially if the noise sources are correlated, as shown in previous sections. A method to find these graphs is to first determine their degree sequence, and then determine which one has the largest average cycle length. This procedure can be simplified by working recursively, building networks with progressively larger order and size.

Lemma 34. *There is always a connected graph of order N and size m in which there are k vertices with degree $d+1$ and $N-k$ vertices with degree d where*

$$d = \left\lfloor \frac{2m}{N} \right\rfloor \quad \textit{and} \quad k = 2m - Nd. \tag{5.38}$$

Proof. We will prove the existence of such a graph by starting with its degree distribution and, by successive transformations, convert it to a graph that is known to exist. Specifically, at each step we will remove one vertex along with its edges, repeating the process until we end up having a cycle graph. Assume that the degree sequence of the graph $\mathcal{G}_0$ is as above, and we arrange the degrees of the vertices in a decreasing order:

$$s_0 = \{\underbrace{d+1, d+1, \ldots, d+1}_{k}, \underbrace{d, d, \ldots, d}_{N-k}\}. \tag{5.39}$$

According to the Havel-Hakimi theorem [33], the above sequence is a graph sequence if and only if the graph sequence in which the largest degree vertex is connected to vertices $2, 3, ..., d+2$ is also a graph sequence. The new graph will have a degree sequence of

$$s_1 = \begin{cases} \{\underbrace{d+1, d+1, \ldots, d+1}_{k-d-2}, \underbrace{d, d, \ldots, d}_{N-k+d+1}\} & \text{if } d < k-2 \\ \{\underbrace{d, d, \ldots, d, d,}_{N+k-d-3} \underbrace{d-1, d-1, \ldots, d-1}_{d-k+2}\} & \text{if } d \geq k-2. \end{cases} \tag{5.40}$$

The key observation is that the transformation above preserves the property of degree homogeneity. In other words, in the new graph $\mathcal{G}_1 = \mathcal{G}_1(N-1, m-d+1)$, the minimum and maximum vertex degrees are

$$d_{min} = \left\lfloor \frac{m-d+1}{N-1} \right\rfloor \tag{5.41}$$

and

$$d_{min} \leq d_{max} \leq d_{min} + 1. \tag{5.42}$$

Repeating the process, there will be a graph $\mathcal{G}_r$ with at least one vertex of degree $d_{min} = 1$. The graph $\mathcal{G}_r$ will include either one or two vertices of degree $d_{min} = 1$. If it has two vertices with degree one, it is the path graph. If it had only one vertex with degree one, the sum of all the degrees of $\mathcal{G}_r$ would be an odd number, which is not possible, since at every transformation we removed $2d_{max}$ from the sum of degrees. The graph $\mathcal{G}_r$ is a connected graph, and implementing the inverse transformations, we connect new vertices to an already connected network, which guarantees that $\mathcal{G}_0$ is connected. □

For small graph orders, we can find all networks with the desired degree sequence, and among them, exhaustively search for the ones with the largest average cycle

length that have the smallest average variance. For $N = 6$ nodes, all connected networks (with $5 \leq m \leq 15$ edges) with most homogeneous degree distribution and longest average cycles are shown in Figure 5.8.

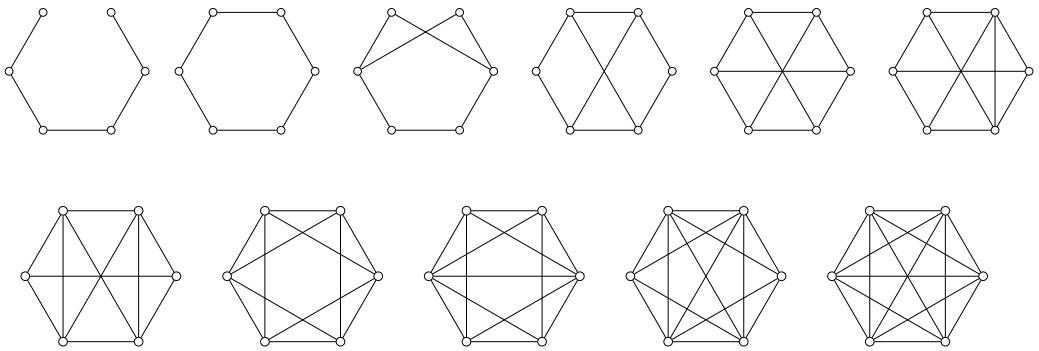


Figure 5.8: All connected networks of order $N = 6$ and size $5 \leq m \leq 15$ and with minimum output variance. We assume that every vertex is affected by an independent noise source. In addition, each vertex acts as a single pole filter. The total noise of the network is measured as the average of the variances of all nodes.

To summarize this subsection, positive correlations increase the output variance, and cycles create correlations that make the system more prone to random inputs. The longer the cycles, the smaller their effect. The immunity to noise is increased when pathways with the same output introduce different phase shifts, so that the different noise contributions cancel each other at least partially. This result holds both for feedforward and feedback loops. When we have some convex constraint on the strength of the various filters, placing the poles, we can find the optimal placement such that the output noise is reduced. Specifically, for a linear network where all nodes act as single pole filters and the dynamics of the network are described by its Laplacian matrix, there is a systematic way to find the network with the smallest average variance. The optimal networks have homogeneous degree distribution, and cycles that are as long as possible.

5.5 Crosstalk Reduces Noise in Pathway Outputs

In this section, we will explore the relation between crosstalk and noise. We will show that crosstalk may decrease the noise in the output of a system, if there are different and uncorrelated noise sources in the inputs.

5.5.1 Motivating Example

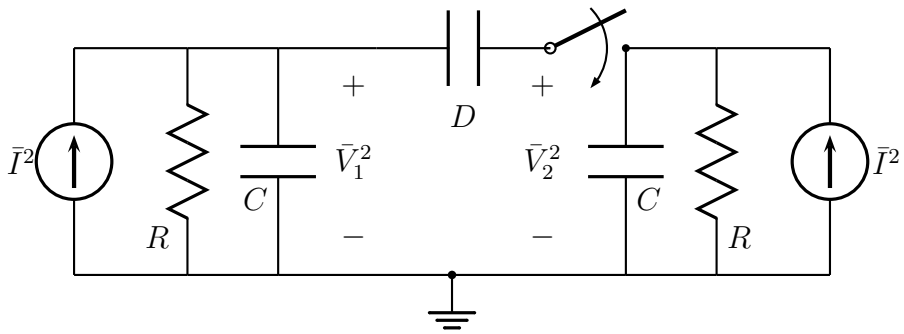


Figure 5.9: A simple circuit with two noise sources. The two resistors generate thermal noise, which is modeled as current sources in parallel to them. When the switch is open, the two circuits are independent. When the switch is closed, the noise in both outputs has smaller variance than before.

Assume that we have a resistor without any external voltage source. If we measure the voltage between its endpoints, we will find that in any infinitesimal frequency interval df there is thermal noise V_t with

$$\mathbb{E}[V_t] = 0 \qquad \text{and} \qquad \mathbb{E}\left[V_t^2\right] = 4kTRdf, \tag{5.43}$$

where R is the resistance. The above equation shows that the noise increases as temperature and resistance increase. We connect a capacitor in parallel to the resistor, and measure the voltage between its endpoints. We are interested in the total amount of variance of the voltage in the output of the parallel combination of the resistor

and the capacitor. When the switch is open, each of the two subcircuits operate independently, and the output variance for both of them is

$$\begin{aligned}\bar{V}_1^2 = \bar{V}_2^2 &= \int_0^{+\infty} \frac{4kT}{R} \left| \frac{R}{1 + j2\pi fRC} \right|^2 df \\ &= \int_0^{+\infty} \frac{4kT}{R} \frac{R^2}{1 + (RC)^2(2\pi f)^2} df \\ &= \frac{4kTR}{2\pi RC} \int_0^{+\infty} \frac{du}{1 + u^2} \\ &= \frac{kT}{C}.\end{aligned} \tag{5.44}$$

If we close the switch, the output variance becomes

$$\begin{aligned}\bar{V}_1^2 = \bar{V}_2^2 &= \frac{4kT}{R} \int_0^{+\infty} \left| \frac{R(1 + j2\pi fR(C + D))}{(j2\pi fR(C + 2D) + 1)(1 + j2\pi fRC)} \right|^2 df \\ &\quad + \frac{4kT}{R} \int_0^{+\infty} \left| \frac{j2\pi fR^2 D}{(1 + j2\pi fRC)(1 + j2\pi fR(C + 2D)} \right|^2 df \\ &= \frac{kTD^2}{2C(C + D)(C + 2D)} + \frac{kT(2C^2 + 4CD + D^2)}{2C(C + D)(C + 2D)} \\ &= \frac{kT}{C} \cdot \frac{C + D}{C + 2D}.\end{aligned} \tag{5.45}$$

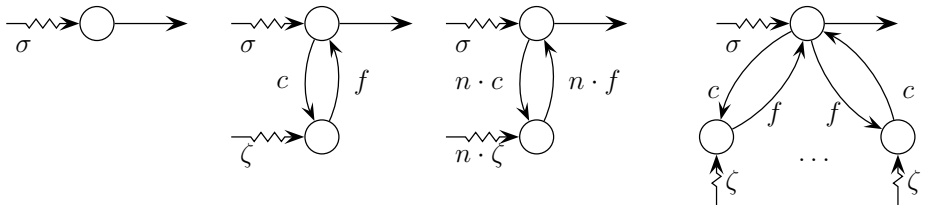
The capacitor that connects the two subcircuits has capacitance $D > 0$, so if the two noise sources are uncorrelated, then both outputs have smaller variance.

In biology, there are countless sources of noise, and the noise is often stronger than the signal itself. It is possible that the cell needs to employ the same technique for reducing noise, distributing it among many different and unrelated components. Crosstalk between different elements couples the behavior of different parts of the network, introducing more poles in the network dynamics. This is equivalent to introducing capacitances between random parts of an electrical network. The new system filters out noise more effectively, but it may be slower to react to various inputs. Consequently, there seems to be a trade-off between how fast a network can respond to changes in the input and how well it filters out noise. The next section

studies the effect of crosstalk on the behavior of small networks.

5.5.2 Crosstalk on Single Nodes

In this subsection, we will analyze the four simple subgraphs of Figure 5.10.



(a) (b) (c) (d)

Figure 5.10: Crosstalk topologies involving one network node. **(a)** A node without crosstalk interactions with white noise input having standard deviation equal to σ. **(b)** A node with crosstalk interaction with one other node in the network, which also is affected by noise with standard deviation ζ. **(c)** Same as before, but we assume that both the crosstalk and the noise are increased by a factor n. **(d)** Crosstalk interactions with many other nodes, each of which has an independent noise input of the same strength. See text for quantitative analysis of these subsystems.

We will disregard any deterministic inputs, since we assume these are linear systems. Any deterministic inputs only affect the output mean, but not the variance. System (a) obeys a simple stochastic differential equation, with one noise input, and it has no other interactions with any other parts of the network.

$$dX = -aXdt + \sigma dW_t. \tag{5.46}$$

The output variance is

$$V_a = \frac{\sigma^2}{2a}. \tag{5.47}$$

This is the trivial case without any crosstalk, and will be used for comparison to the performance of the other subnetworks.

Subsystem (b) consists of one vertex that interacts with another node which may also be prone to other noise sources. Crosstalk is modeled through a new vertex in the network, with which the studied node exchanges flows. In chemical reaction networks for example, the species of interest X may be forming a complex Y with species I. The concentration of I is assumed to be constant.

$$X + I \underset{f}{\overset{c}{\rightleftharpoons}} Y. \tag{5.48}$$

We also expect X to have a constant degradation rate a. The equations for the concentrations of X and Y are

$$\begin{aligned} dX &= -(a+c)X dt + fY dt + \sigma dW_t \\ dY &= cX dt - fY dt + \zeta dU_t \end{aligned} \tag{5.49}$$

and the output variance is

$$V_b = \frac{a+f}{2a(a+c+f)}\sigma^2 + \frac{f}{2a(a+c+f)}\zeta^2. \tag{5.50}$$

The next step is to see what happens if we increase the crosstalk intensity by a factor n. We can distinguish two cases. The first is when there is crosstalk with one other node (Figure 5.10(b)). In the chemical reaction network analogy,

$$X + A \underset{n\cdot f}{\overset{n\cdot c}{\rightleftharpoons}} Y. \tag{5.51}$$

It is straightforward to find the new differential equations, and the variance in the output.

$$\begin{aligned} dX &= -(a+nc)X dt + nfY dt + \sigma dW_t \\ dY &= ncX dt - nfY dt + n\zeta dU_t \end{aligned} \tag{5.52}$$

$$V_c = \frac{a+nf}{2a(a+n(c+f))}\sigma^2 + \frac{n^3 f}{2a(a+n(c+f))}\zeta^2. \tag{5.53}$$

Finally, we consider the case where one node has crosstalk interactions with many different nodes, each of which is affected by a different noise process (Figure 5.10(d)).

$$\begin{aligned} dX &= -(a+nc)Xdt + nfYdt + \sigma dW_t \\ dY_k &= cXdt - fY_k dt + \zeta dU_t^k \qquad 1 \leq k \leq n \end{aligned} \tag{5.54}$$

and the output variance is

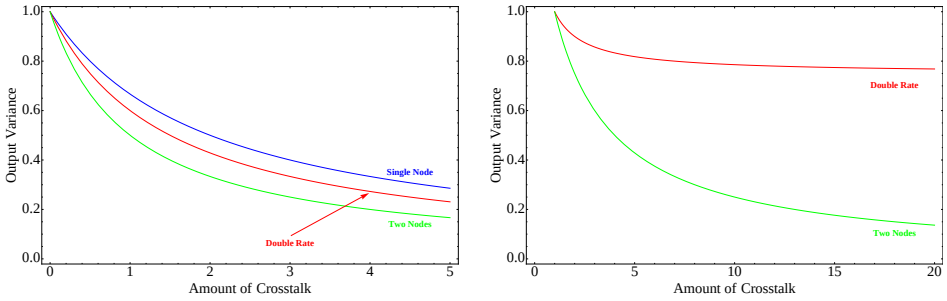
$$V_d = \frac{a+f}{2a(a+nc+f)}\sigma^2 + \frac{nf}{2a(a+nc+f)}\zeta^2. \tag{5.55}$$

When no noise is introduced from the crosstalk nodes ($\zeta = 0$), crosstalk reduces the output variance. Figure 5.11 compares the last three cases, as the strength of crosstalk interactions among the nodes increases. The crosstalk strength in this case is quantified by the ratio

$$r_x = \frac{c}{c+f} \tag{5.56}$$

which is equal to the concentration of the crosstalk product Y in equation (5.49) in the absence of degradation rates and noise inputs. It is shown that distributing the crosstalk among many nodes (equation (5.55)) decreases the effect of noise noticeably more, compared to the single node case. This is even more pronounced when we normalize by the variance in the base case (equation (5.50)). When crosstalk introduces additional noise, it may increase the variance in the output of any given node if crosstalk is not strong enough to make up for the introduced noise (Figure 5.12).

It should be noted that although crosstalk affects the output variance, it leaves the output mean unaffected. Crosstalk is modeled as additional differential equations that couple the existing states together, and when they receive no external deterministic inputs, they do not alter the mean of the steady state response of the system. Consequently, the coefficient of variation of the output changes proportionally to its standard deviation, since the mean is unchanged.



(a) (b)

Figure 5.11: Output variance as a result of noise input for a single vertex in the network given the existence of crosstalk interactions with other vertices. **(a)** Output variance as a function of the amount of crosstalk (concentration of crosstalk complex), when no additional noise is introduced. Crosstalk clearly mitigates the output variance. Having crosstalk with two independent nodes reduces the variance even more, compared to having a single crosstalk node. **(b)** Normalized output variance as a fraction of the variance when only one crosstalk node is present. Having many small sources of crosstalk is clearly better than having one strong crosstalk interaction. For the same amount of total crosstalk, dividing it among many nodes drives the output noise variance to zero as the number of nodes increases.

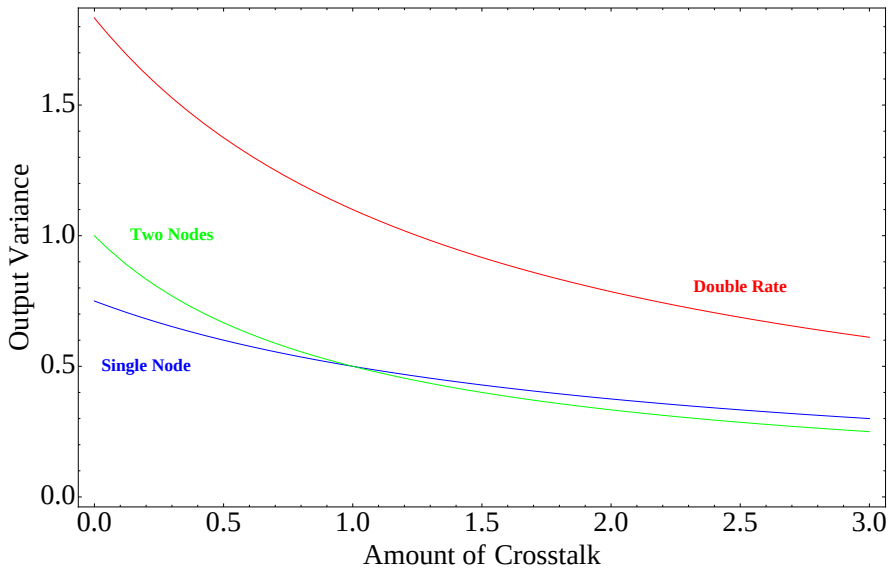


Figure 5.12: Normalized variance of the output when the crosstalk introduces additional noise. Having strong crosstalk interactions with one single node increases the variance because noise propagates easily. When crosstalk is distributed among many nodes, the variance may be smaller or larger than before, depending on the strength of the interactions. This is because having crosstalk interactions with many other vertices introduces a proportional amount of noise.

5.5.3 Parallel Pathways

We consider two pathways with crosstalk among more than one of their nodes. We distinguish two cases, when the two pathways have the same or different outputs. In the first case, since the two outputs are independent, it is easier to reduce the noise variance in both of them, by "exchanging" their noise through each node, assuming that the different noise sources are independent. When the output is the same, there is little reduction in the output variance, since every disturbance eventually reaches the output, and is combined with other correlated versions of the same signal (Figure 5.13). The variance reduction is then caused by the increase of the effective pathway lengths, since they follow on average a longer path towards the output.

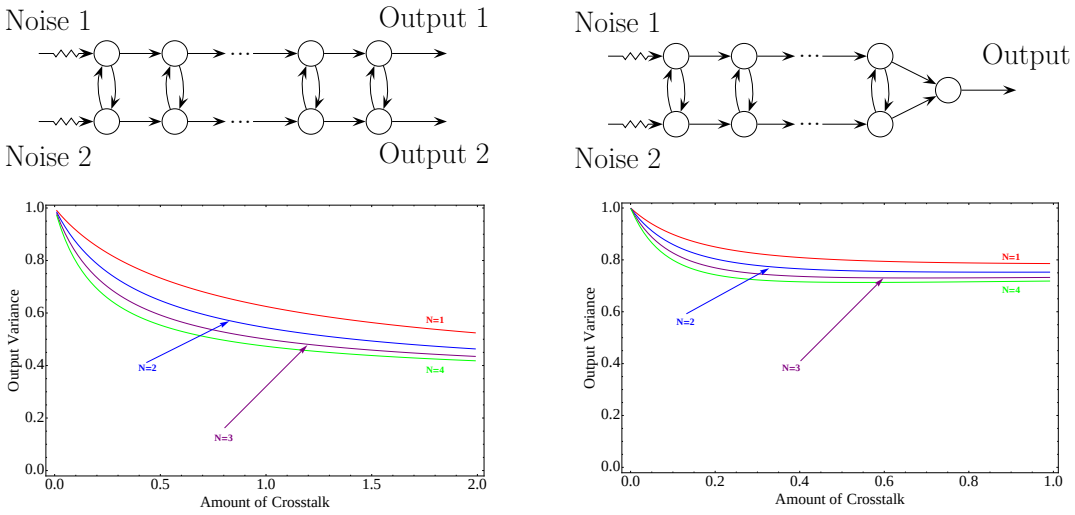


Figure 5.13: Output variance when crosstalk is present among all stages of two different pathways for various pathway lengths, when their output is different (left) or the same (right). The output variances are normalized by the variance of a pathway without crosstalk. We assume that every stage of the pathway has some noise input. A small amount of crosstalk can help reduce the effect of noise in the output, but more crosstalk does not help filtering out the noise of the system. Crosstalk has a much smaller effect when the two pathways have the same output. Although it reduces the variance of the intermediate nodes, it creates correlations among them that in turn increase the variance in the output.

5.5.4 Crosstalk Modeling: Direct Conversion and Intermediate Nodes

Suppose we have a simple decomposed system:

$$\begin{aligned} dY_1 &= -aY_1dt + \sigma dU_t \\ dY_2 &= -aY_2dt + \sigma dW_t. \end{aligned} \tag{5.57}$$

The two outputs of the system are completely independent, since they do not interact in any way, and are therefore uncorrelated. The variance of each output is:

$$\mathbb{V}[Y_1] = \mathbb{V}[Y_2] = \frac{\sigma^2}{2a}. \tag{5.58}$$

If there is crosstalk, then the different states of the system are correlated. If we model crosstalk as a positive conversion rate from one state to another, with the conversion rates being equal among every pair of states, the $2-$state system above becomes:

$$\begin{aligned} dY_1 &= -(a+c)Y_1dt + cY_2dt + \sigma dU_t \\ dY_2 &= -(a+c)Y_2dt + cY_1dt + \sigma dW_t. \end{aligned} \tag{5.59}$$

The variance of each of the outputs now becomes:

$$\begin{aligned} \mathbb{V}[Y_1] &= \int_{-\infty}^{+\infty} (|h_{11}(f)|^2 + |h_{21}(f)|^2)df \\ &= \frac{\sigma^2}{2\pi} \int_{-\infty}^{+\infty} \left(\left| \frac{a+c+j\omega}{(a+j\omega)(a+2c+j\omega)} \right|^2 + \left| \frac{c}{(a+j\omega)(a+2c+j\omega)} \right|^2 \right) d\omega \\ &= \frac{\sigma^2}{2a} \cdot \frac{a+c}{a+2c}, \end{aligned} \tag{5.60}$$

where h_{11} and h_{21} are the impulse responses of the first node when the input is an impulse response to the first and second node, respectively. The symmetry is preserved, so $\mathbb{V}[Y_1] = \psi_y^2 = \mathbb{V}[Y_2]$. The variance when crosstalk is present ($c > 0$) is always smaller than the initial variance of the outputs. Generalizing the equations

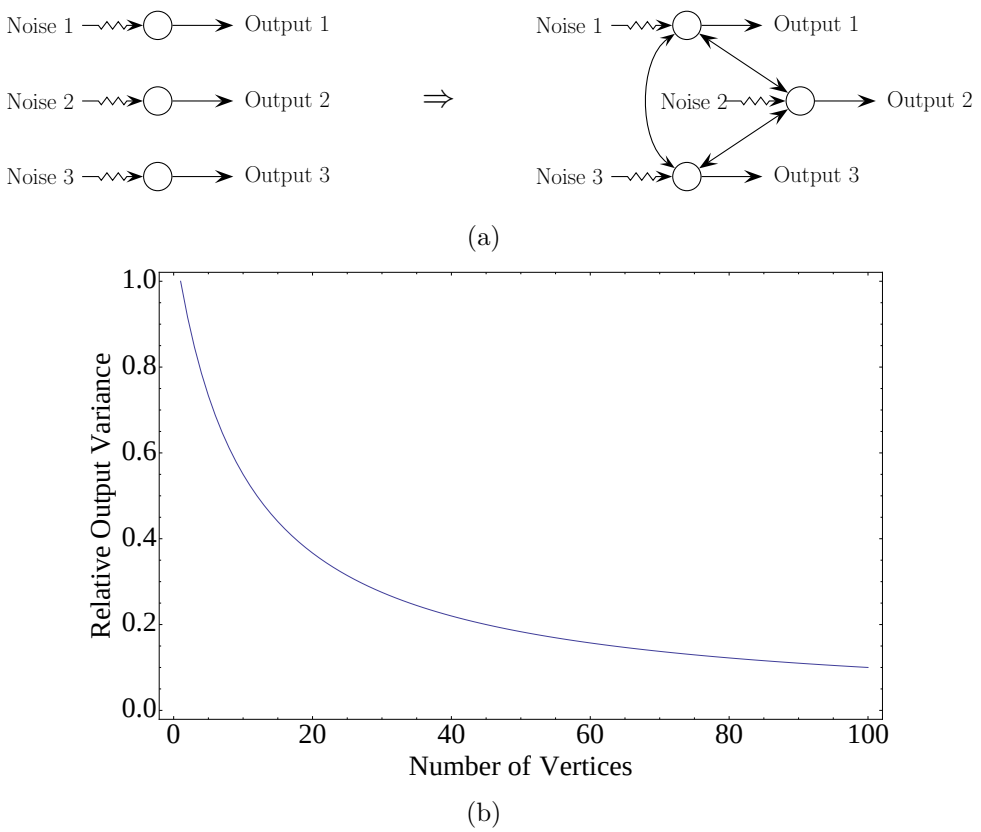


Figure 5.14: Output variation for each node in a system of N nodes, when there are crosstalk interactions among every pair of nodes. The variance has been normalized by the corresponding variance without crosstalk. Each node is identical, and receives an independent noise input of the same intensity. When the number of vertices increases, the noise is distributed among all the nodes, thus the output variance is reduced.

above for N nodes (see Figure 5.14), we find that

$$\sigma_y^2 = \frac{\sigma^2}{2a} \quad \text{and} \quad \psi_N^2 = \frac{a+c}{(a+Nc)}\sigma_y^2 \tag{5.61}$$

and as a result,

$$\frac{\psi_N^2}{\sigma_y^2} = \frac{a+c}{a+Nc} \tag{5.62}$$

which tends to zero as N becomes large.

Alternatively, we can model crosstalk interactions as two species being converted

to an intermediate complex, as has been done in the previous sections. A very simple example of a chemical reaction network which demonstrates this type of behavior is

$$\begin{aligned} Y_1 &\to A \\ Y_2 &\to B \\ Y_1 + Y_2 &\rightleftharpoons Z. \end{aligned} \tag{5.63}$$

Crosstalk is defined by the presence of the last reaction. We are interested in the variance of the concentration of the output products A and B, which are directly affected by the variance of Y_1 and Y_2. The two pathways will interact through an intermediate vertex. The system can be written as

$$\begin{aligned} dY_1 &= -aY_1dt - cY_1Y_2dt + fZdt + \sigma dU_t \\ dY_2 &= -aY_2dt - cY_1Y_2dt + fZdt + \sigma dW_t \\ dZ &= cY_1Y_2dt - fZdt. \end{aligned} \tag{5.64}$$

We assume that there is a new "crosstalk vertex" Z among each pair of original vertices. After linearizing around the steady state $(\bar{Y}_1, \bar{Y}_2, \bar{Z})$, the three equations above become

$$\begin{aligned} dY_1 &= -(a + c\bar{Y}_2)Y_1dt - c\bar{Y}_1Y_2dt + fZdt + \sigma dU_t \\ dY_2 &= -(a + c\bar{Y}_1)Y_2dt - c\bar{Y}_2Y_1dt + fZdt + \sigma dW_t \\ dZ &= c\bar{Y}_2Y_1dt + c\bar{Y}_1Y_2dt - fZdt. \end{aligned} \tag{5.65}$$

The network is now more capable of reducing the output variance (Figure 5.15).

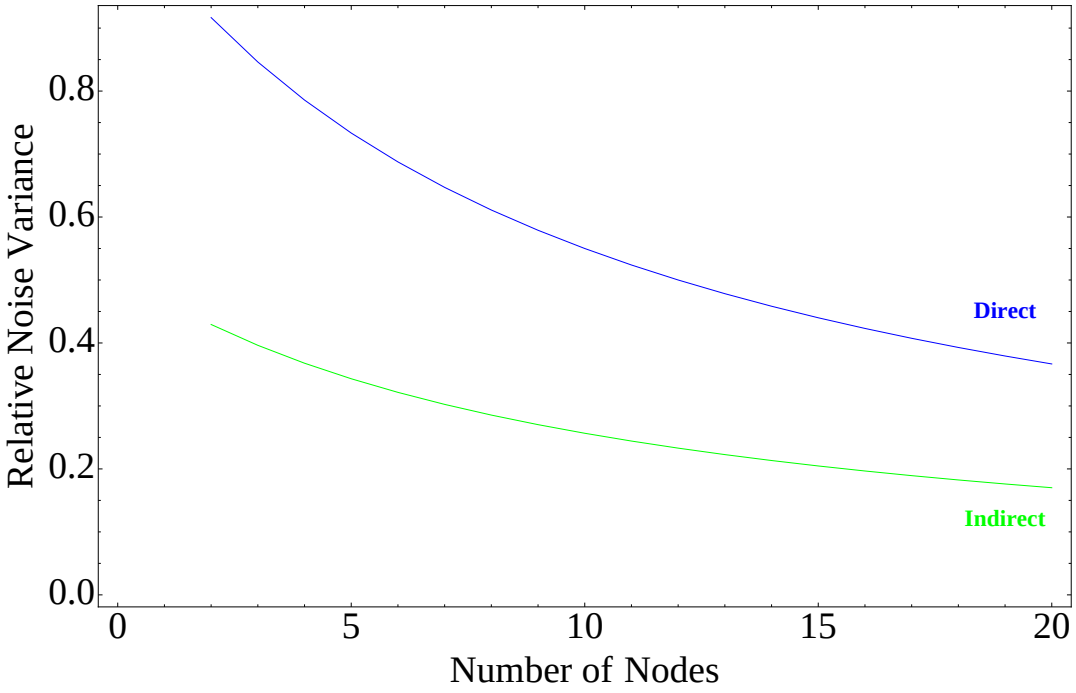


Figure 5.15: Comparison of the noise in the output of a simple network with two different implementations of crosstalk, direct conversion or forming a new complex, as described by equations (5.59) and (5.64).

5.6 Multiplicative (Geometric) Noise

There are cases where the noise intensity is proportional to a state of the system. In biological networks for example, the degradation of various proteins depends on specific enzymes, whose concentration may be subject to random fluctuations. This makes the degradation of a protein prone to noise whose source is independent of the protein concentration, but makes the rate at which it degrades proportional to it. The noise intensity is also proportional to the state of the system when a state is autoregulated, either with positive or negative feedback, if the rate at which the concentration of that particular state changes is subject to random noise. We will call this type of noise multiplicative or geometric, because it is multiplied by the state of the system. As a specific example, consider a gene that is regulated by a single

regulator [2]. The transcription interaction can be written as

$$P \rightarrow X. \tag{5.66}$$

When P is in its active form, gene X starts being transcribed and the mRNA is translated, resulting in accumulation of protein X at a rate b_t. The production of X is balanced by protein degradation (by other specialized proteins) and cell dilution during growth at rate a_t. The differential equation that describes this simplified system is:

$$\frac{dX}{dt} = b_t - a_t X. \tag{5.67}$$

If there is noise in the concentration of the degradation proteins, or the cell growth rate is not constant, a_t consists of a deterministic component and a random component. We will now show that noise in the production rate b_t has a fundamentally different effect on system behavior compared to the effect of noise in the degradation rate a_t, because the latter is multiplied by the concentration of the protein itself. This analysis focuses on the random and constant fluctuation of network parameters. When the parameters are unknown but constant, it has been shown that there are adaptation mechanisms that ensure the robustness of a system regardless of the exact value of each parameter [9]. We will first study the homogenous version of the differential equation (5.67), and then we will add the constant production term. Ignoring the production term b_t, and multiplying by dt, equation (5.67) becomes

$$dX = -(a_t dt)X. \tag{5.68}$$

After adding a random component in the degradation rate, the last equation becomes

$$dX = (-a_t dt + \sigma_t dW_t)X, \tag{5.69}$$

where W_t is the regular Wiener process and dW_t represents the noise term. Note that the degradation rate and the noise intensity are allowed to be time-dependent. We will first find the differential of the logarithm of X using Itô's lemma [26]. We will require that all the input functions are continuous and non-pathological, so that we can always change the order of taking the limit and the expectation operator. All integrals are assumed to be finite, so that we can also change the order of integration. The technical details mentioned above are covered in more detail in [26] and [44]. We denote by $f(X,t)$ the logarithm of the state variable X:

$$f(X,t) = \log(X_t). \tag{5.70}$$

We apply Itô 's lemma on the last function:

$$\begin{aligned} d\log(X_t) &= df(X,t) \\ &= \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial X}dX + \frac{1}{2}\frac{\partial^2 f}{\partial X^2}dX^2 \\ &= 0 + \frac{dX}{X} - \frac{1}{2}\frac{1}{X^2}\left(a_t^2X^2dt^2 - 2a_t\sigma_tX^2dtdW_t + \sigma_t^2X^2dW_t^2\right) \\ &= \left(-a_tdt + \sigma_tdW_t - \frac{1}{2}\sigma^2dW_t^2\right) + X^2\left(a_t^2dt^2 - 2a_t\sigma_tdtdW_t\right). \end{aligned} \tag{5.71}$$

The last two terms can be neglected, since $dt^2 = \mathrm{O}(dt)$ and $dt \cdot dW_t = \mathrm{O}(dt)$ as $dt \to 0$. On the other hand, as dt becomes small,

$$\lim_{dt\to 0} dW_t^2 = \mathbb{E}[dW_t^2] = dt. \tag{5.72}$$

Applying the rules above to equation (5.71),

$$\log\frac{X(t)}{X_0} = -\int_0^t \left(a_s + \frac{1}{2}\sigma_s^2\right)ds + \sigma_tW_t. \tag{5.73}$$

We can now solve for $X(t)$:

$$X_t = X_0 e^{-\int_0^t \left(a_s + \frac{1}{2}\sigma_s^2\right) ds} \cdot e^{\sigma_t W_t}. \tag{5.74}$$

The above derivation is valid only when the equilibrium state (concentration) is equal to zero and we start from a state $X_0 \neq 0$. If the rate a_t and the noise strength σ_t are constant, it simplifies to

$$X_t = X_0 e^{-\left(a + \frac{1}{2}\sigma^2\right)t} \cdot e^{\sigma W_t}. \tag{5.75}$$

When the equilibrium is positive (which is the case for most systems), the following differential equation with a deterministic input is more relevant:

$$dY = b_t dt + (-a_t dt + \sigma_t dW_t)Y. \tag{5.76}$$

One way to view the terms on the right hand side of equation (5.76) is that the concentration of species X depends on a deterministic input, and is regulated by a negative feedback mechanism which is subject to random disturbances. It has been shown in [42] that when feedback is also noisy, there are fundamental limits on how much the noise in the output can be reduced, because there are bounds on how well we can estimate the state of the system. The authors focus on discrete random events (birth-death processes) as the source of noise, and the result is that feedback noise makes it harder to control the noise in the output.

We will also study how multiplicative noise affects the output of a system compared to additive noise of equal strength, and how it propagates in a cascade of linear filters. Using Itô's lemma and the solution to the homogeneous equation, the solution of the inhomogeneous stochastic differential equation is (see [26] for details):

$$\begin{aligned} Y(t) &= Y_0 X(t) + bX(t) \int_0^t X^{-1}(s) ds \\ &= Y_0 e^{-\int_0^t \left(a_x + \frac{1}{2}\sigma_x^2\right) dx} \cdot e^{\sigma_t W_t} + b \int_0^t e^{-\int_s^t \left(a_x + \frac{1}{2}\sigma_x^2\right) dx} \cdot e^{\sigma_t W_t - \sigma_s W_s} ds \end{aligned} \tag{5.77}$$

where $X(t)$ is the solution of the homogeneous equation (5.74) with initial condition $X(t=0)=1$. If the initial state is equal to zero (or when t is large), and all the parameters are constant, then we can simplify the last expression:

$$Y(t) = b\int_0^t e^{-\left(a+\frac{1}{2}\sigma^2\right)s}\cdot e^{\sigma W_s}ds. \tag{5.78}$$

The form of the last equation is fundamentally different from the response of linear systems to input noise, because the Wiener process input depends on the same time variable as the kernel of the integral, and the output is not a convolution of the impulse response of the system with the input. In order to see how noise propagates through the network, it is helpful to find the autocorrelation of two versions of this stochastic process. We cannot use equation (5.9), because the random input is now multiplied by the system state. As a first step, we will compute the correlation of the exponential of Brownian motion. The expected value is

$$\begin{aligned}\mathbb{E}\left[Z_t\right] &= \mathbb{E}\left[e^{\sigma W_t}\right] \\ &= \int_{-\infty}^{+\infty} e^{\sigma\sqrt{t}x}\frac{1}{\sqrt{2\pi t}}e^{-\frac{x^2}{2t}}dx \\ &= e^{\frac{1}{2}\sigma^2 t}.\end{aligned} \tag{5.79}$$

The expected value of the square of the exponential Wiener process is

$$\mathbb{E}\left[Z_t^2\right] = \mathbb{E}\left[e^{2\sigma W_t}\right] = e^{2\sigma^2 t}. \tag{5.80}$$

Combining the last two equations:

$$\begin{aligned}\sigma_{Z_t}^2 = \mathrm{Var}\left[Z_t\right] &= \mathbb{E}\left[Z_t^2\right] - \left(\mathbb{E}\left[Z_t\right]\right)^2 \\ &= e^{2\sigma^2 t} - e^{\sigma^2 t} \\ &= e^{\sigma^2 t}\left(e^{\sigma^2 t} - 1\right).\end{aligned} \tag{5.81}$$

The expected value of $Y(t)$ in equation (5.78) can now be computed:

$$\begin{aligned}\mathbb{E}\left[Y(t)\right] &= b\int_0^t e^{-\left(a+\frac{1}{2}\sigma^2\right)u}\cdot\mathbb{E}\left[e^{\sigma W_u}\right]du \\ &= b\int_0^t e^{-\left(a+\frac{1}{2}\sigma^2\right)x}\cdot e^{\frac{1}{2}\sigma^2 x}dx \\ &= \frac{b}{a}(1-e^{-at})\end{aligned} \tag{5.82}$$

which means that

$$\bar{Y} = \lim_{t\to\infty}\mathbb{E}[Y(t)] = \frac{b}{a}. \tag{5.83}$$

As one would expect, it the same as when the system is completely deterministic. Next, we compute the covariance of two samples of the random process Z_t:

$$\begin{aligned}\mathrm{Cov}\left[Z_s, Z_t\right] &= \mathbb{E}\left[Z_s\cdot Z_t\right] - \mathbb{E}\left[Z_s\right]\cdot\mathbb{E}\left[Z_t\right] \\ &= \mathbb{E}\left[e^{\sigma W_s}e^{\sigma W_t}\right] - \mathbb{E}\left[e^{\sigma W_s}\right]\cdot\mathbb{E}\left[e^{\sigma W_t}\right] \\ &= \mathbb{E}\left[e^{\sigma W_{s\wedge t}}e^{\sigma W_{s\vee t}}\right] - e^{\frac{1}{2}\sigma^2(s+t)} \\ &= \mathbb{E}\left[e^{2\sigma W_{s\wedge t}}e^{\sigma(W_{s\vee t}-W_{s\wedge t})}\right] - e^{\frac{1}{2}\sigma^2(s+t)} \\ &= \mathbb{E}\left[e^{2\sigma W_{s\wedge t}}\right]\cdot\mathbb{E}\left[e^{\sigma(W_{s\vee t}-W_{s\wedge t})}\right] - e^{\frac{1}{2}\sigma^2(s+t)} \\ &= e^{2\sigma^2 s\wedge t}\cdot e^{\frac{1}{2}\sigma^2(s\vee t - s\wedge t)} - e^{\frac{1}{2}\sigma^2(s+t)}\end{aligned} \tag{5.84}$$

where we follow the standard notation $s\wedge t = \min(s,t)$ and $s\vee t = \max(s,t)$. Combining these equations, we find the autocorrelation for the geometric Brownian motion:

$$\begin{aligned}R(s,t) &= \mathrm{Corr}\left[Z_s, Z_t\right] \\ &= \frac{\mathrm{Cov}\left[Z_s, Z_t\right]}{\sigma_{Z_s}\cdot\sigma_{Z_t}} \\ &= \frac{e^{2\sigma^2 s\wedge t}\cdot e^{\frac{1}{2}\sigma^2(s\vee t - s\wedge t)} - e^{\frac{1}{2}\sigma^2(s+t)}}{\sqrt{e^{\sigma^2 s}\left(e^{\sigma^2 s}-1\right)}\sqrt{e^{\sigma^2 t}\left(e^{\sigma^2 t}-1\right)}} \\ &= \sqrt{\frac{e^{\sigma^2 s\wedge t}-1}{e^{\sigma^2 s\vee t}-1}}.\end{aligned} \tag{5.85}$$

The general autocorrelation formula of geometric noise in the steady state is:

$$\begin{aligned} R(\tau) &= \lim_{t\to\infty} R(t, t+\tau) \\ &= \lim_{t\to\infty} \sqrt{\frac{e^{\sigma^2 t}-1}{e^{\sigma^2(t+\tau)}-1}} \\ &= \sqrt{\frac{e^{\sigma^2 t}}{e^{\sigma^2(t+\tau)}}} \\ &= e^{-\frac{1}{2}\sigma^2\tau}. \end{aligned} \tag{5.86}$$

The correlation is exponentially decreasing as a function of the time lag. We can now follow the same procedure in order to find the correlation of the stochastic process defined by equation (5.78). Its second moment is equal to

$$\begin{aligned} \mathbb{E}[Y^2(t)] &= b^2 \int_0^t \int_0^t e^{-\left(a+\frac{\sigma^2}{2}\right)(x+y)} \cdot \mathbb{E}\left[e^{\sigma W_x} e^{\sigma W_y}\right] dxdy \\ &= b^2 \int_0^t \int_0^t e^{-\left(a+\frac{\sigma^2}{2}\right)(x+y)} e^{2\sigma^2 x\wedge y} e^{\frac{1}{2}\sigma^2(x\vee y - x\wedge y)} dxdy \\ &= b^2 \int_0^t \int_0^x e^{-\left(a+\frac{\sigma^2}{2}\right)(x+y)} e^{2\sigma^2 y} e^{\frac{1}{2}\sigma^2(x-y)} dxdy \\ &\qquad + b^2 \int_0^t \int_x^t e^{-\left(a+\frac{\sigma^2}{2}\right)(x+y)} e^{2\sigma^2 x} e^{\frac{1}{2}\sigma^2(y-x)} dxdy \\ &= \frac{2\left(a\left(1-2e^{-at}+e^{t\left(-2a+\sigma^2\right)}\right)+\left(-1+e^{-at}\right)\sigma^2\right)}{a\left(2a^2-3a\sigma^2+\sigma^4\right)}. \end{aligned} \tag{5.87}$$

All integrals are assumed to be finite, meaning that a has to be greater than $\frac{\sigma^2}{2}$. As t goes to infinity, we may ignore all the decaying exponentials.

$$\lim_{t\to\infty} \mathbb{E}[Y^2(t)] = \begin{cases} \infty & \text{if } a \leq \frac{\sigma^2}{2} \\ \\ \frac{b^2}{a(a-\frac{\sigma^2}{2})} & \text{if } a > \frac{\sigma^2}{2}. \end{cases} \tag{5.88}$$

In what follows, we will only be interested in the behavior of the system when

$a > \frac{\sigma^2}{2}$, when the standard deviation is finite. Based on equation (5.88), the standard deviation (when it is defined) is equal to

$$\sigma_Y^2 = \frac{b^2\sigma^2}{a^2\left(2a-\sigma^2\right)} = \frac{\sigma^2}{2a-\sigma^2}\bar{Y}^2. \tag{5.89}$$

The standard deviation is proportional to the square of the average value of Y. The output variance when the parameters are themselves noisy have a different form, compared to the variance when the inputs are noisy.

A similar formalism exists in the control literature for both continuous time systems [43] and discrete time systems [15]. In control theory, the focus is on state estimation and on finding feedback mechanisms that can accurately estimate the state of a system, or minimize its variance. In the next subsection, we will analytically compute the mean and the variance of the output of a system when geometric noise is supplied as input to a series of linear filters.

5.6.1 Geometric Noise Through a Low-Pass Filter

Assume that a system consists of two systems connected in series. The first one is affected by geometric noise, and it is used as an input to the second node. We first analyze a system where each state has one real pole, and later on we will generalize it for an arbitrary number of poles. The equations of the system are

$$\begin{aligned} dX &= cdt + (-fdt + \sigma dW_t)X \\ dY &= bXdt - aYdt, \end{aligned} \tag{5.90}$$

where all parameters are positive real numbers. Combining the forms for the multiplicative noise and the output of a single pole filter,

$$Y(t) = bce^{-at}\int_0^t e^{as}\left(\int_0^s e^{-\left(f+\frac{\sigma^2}{2}\right)u}e^{\sigma W_u}du\right)ds. \tag{5.91}$$

The goal is to use the previous methods to explore the propagation of noise in a network of first order filters when the input noise is geometric. The output mean is:

$$\begin{aligned}\mathbb{E}[Y(t)] &= bce^{-at}\int_0^t e^{as}\left(\int_0^s e^{-\left(f+\frac{\sigma^2}{2}\right)u}\mathbb{E}\left[e^{\sigma W_u}\right]du\right)ds \\ &= bce^{-at}\int_0^t e^{as}\left(\int_0^s e^{-fu}du\right)ds \\ &= \frac{bc\left(a-ae^{-ft}+\left(-1+e^{-at}\right)f\right)}{a(a-f)f}.\end{aligned} \tag{5.92}$$

The last equation holds even if $a = f$, and we can find the expected value by finding the limit as $f \to a$. Letting the time t go to infinity,

$$\mathbb{E}[Y] = \lim_{t\to\infty}\mathbb{E}[Y(t)] = \frac{bc}{af} \tag{5.93}$$

which is the same as an equivalent system without any noise. The second moment is

$$\mathbb{E}[Y^2] = b^2c^2e^{-2at}\int_0^t e^{ar}dr\int_0^t e^{as}ds\int_0^r\int_0^s e^{-(f+\frac{\sigma^2}{2})(x+y)}\mathbb{E}\left[e^{\sigma(W_x+W_y)}\right]dxdy. \tag{5.94}$$

We break the integral above in five parts, in order to compute the expected value inside it:

$$\begin{aligned}\frac{e^{2at}}{b^2c^2}\mathbb{E}[Y^2(t)] = &\int_0^t e^{ar}dr\int_r^t e^{as}ds\int_0^r\left(\int_0^x e^{-fx}e^{-fy}e^{\sigma^2 y}\,dy\right)dx \\ &+\int_0^t e^{ar}dr\int_r^t e^{as}ds\int_0^r\left(\int_x^s e^{-fx}e^{-fy}e^{\sigma^2 x}dy\right)dx \\ &+\int_0^t e^{ar}dr\int_0^r e^{as}ds\int_0^s\left(\int_0^x e^{-fx}e^{-fy}e^{\sigma^2 y}dy\right)dx \\ &+\int_0^t e^{ar}dr\int_0^r e^{as}ds\int_0^s\left(\int_x^s e^{-fx}e^{-fy}e^{\sigma^2 x}dy\right)dx \\ &+\int_0^t e^{ar}dr\int_0^r e^{as}ds\int_s^r\left(\int_0^s e^{-fx}e^{-fy}e^{\sigma^2 y}dy\right)dx.\end{aligned} \tag{5.95}$$

The above sum of integrals is finite provided that $f > \frac{\sigma^2}{2}$. After performing all the algebraic calculations,

$$E[Y^2] = \lim_{t\to\infty} \mathbb{E}[Y^2(t)] = \frac{b^2c^2}{a^2 f(f - \frac{\sigma^2}{2})}. \tag{5.96}$$

The variance is

$$\mathbb{V}[Y] = b^2c^2 \frac{\sigma^2}{a^2 f^2(2f - \sigma^2)}. \tag{5.97}$$

We can write the above equation as a constant times the variance of the first state:

$$\begin{aligned} \mathbb{V}[Y] &= \left(\frac{b}{a}\right)^2 \frac{c^2\sigma^2}{f^2(2f - \sigma^2)} \\ &= \left(\frac{b}{a}\right)^2 \mathbb{V}[X]. \end{aligned} \tag{5.98}$$

The variance of Y is, also in this case, fundamentally different from the variance of a system with white noise added directly to the input. The latter would be equal to

$$V_0[Y] = \frac{b^2}{2a}\sigma_{in}^2. \tag{5.99}$$

The time evolution of the variance is shown in Figure 5.16. When the noise is geometric, it takes longer for the variance to settle to its steady state value, which is also an indication that the output variance consists of lower frequencies than in the case of additive noise.

More generally, if we pass geometric noise through an arbitrary linear filter with impulse response $h(t)$, then the output is defined as the convolution of the impulse response and the input:

$$Y(t) = c\int_0^t h(t-s)\left(\int_0^s e^{-\left(f+\frac{\sigma^2}{2}\right)u} e^{\sigma W_u} du\right) ds. \tag{5.100}$$

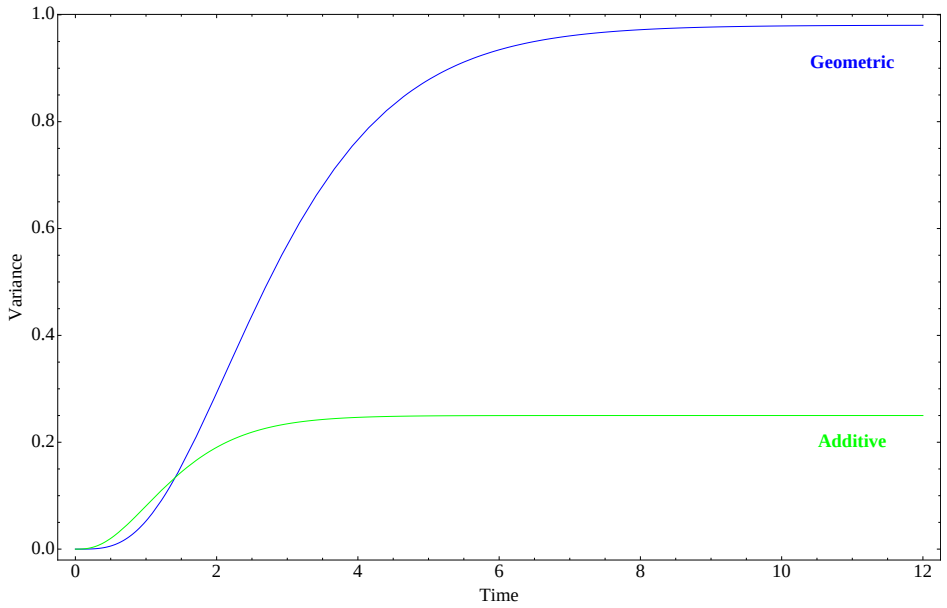


Figure 5.16: Evolution of the output variance of a single pole filter when the input is affected by additive and multiplicative noise, respectively. The system with additive noise has less variance in the output compared to the one with multiplicative noise. Also, in the case of geometric noise, the variance takes more time to settle to its equilibrium value.

The mean is

$$\begin{aligned}\mathbb{E}[Y(t)] &= c\int_0^t h(t-s)\left(\int_0^s e^{-fu}du\right)ds \\ &= \frac{c}{f}\left(\int_0^t (1-e^{-fs})h(t-s)ds\right).\end{aligned} \tag{5.101}$$

The variance is equal to

$$\begin{aligned}\mathbb{V}[Y(t)] &= \mathbb{E}[Y^2(t)] - \left(\mathbb{E}[Y(t)]\right)^2 \\ &= c\int_0^t h(t-r)dr\int_0^r h(t-s)ds\int_0^s\int_0^y e^{-f(x+y)}e^{\sigma^2 x}dxdy \\ &\quad + c\int_0^t h(t-r)dr\int_0^r h(t-s)ds\int_0^s\int_y^r e^{-f(x+y)}e^{\sigma^2 y}dxdy \\ &\quad + c\int_0^t h(t-r)dr\int_r^t h(t-s)ds\int_0^r\int_0^x e^{-f(x+y)}e^{\sigma^2 y}dydx \\ &\quad + c\int_0^t h(t-r)dr\int_r^t h(t-s)ds\int_0^r\int_x^s e^{-f(x+y)}e^{\sigma^2 x}dydx \\ &\quad - \frac{c^2}{f^2}\left(\int_0^t (1-e^{-fs})h(t-s)ds\right)^2.\end{aligned} \tag{5.102}$$

For example, if the filter has one pole at $-a$ with $a > 0$, then $h(t,s) = e^{-a(t-s)}$, we can verify that the mean and the variance are equal to the ones found in equations (5.93) and (5.96). If we have n identical single-pole filters in series, with the same pole at $-a$, with $a \in \mathbb{R}^+$, and their input is multiplied by b, then the mean is

$$\begin{aligned}\mathbb{E}[Y] &= \lim_{t\to\infty} \frac{b^n c}{f} \int_0^t (1 - e^{-fs}) \frac{(t-s)^{n-1}}{(n-1)!} e^{-as} ds \\ &= \left(\frac{b}{a}\right)^n \cdot \frac{c}{f}\end{aligned} \tag{5.103}$$

and the variance is

$$\mathbb{V}[Y] = \left(\frac{b}{a}\right)^{2n} \left(\frac{c}{f}\right)^2 \frac{\sigma^2}{(2f - \sigma^2)}. \tag{5.104}$$

These results show how variation that enters the system through noisy degradation rates affects the output of a pathway. For example, in the two-step cascade

$$\begin{aligned} X &\to Y \\ Y &\to Z \end{aligned} \tag{5.105}$$

described by (5.90), species Y is affected by geometric noise, and then is used as an input to the next reaction that produces Z. The second reaction acts as a first-order linear filter, and the noise propagates to the pathway output Z. The analysis can be used for any system that can be described by linear differential equations. If a linear time invariant system is described by equation (2.7) then, if there is noise in the input u or its input matrix B, then we can consider noise a new additional input as in equation (5.1), and solve it accordingly. The same holds for the off-diagonal elements of the dynamical matrix A. But noise in the diagonal elements of A is multiplicative noise, and needs to be considered separately from all other noise sources, and it leads to qualitatively different behavior.

5.7 Noise Propagation in Chemical Reaction Networks

In this section, we will examine how noise propagates in general linear chemical reaction networks. Noise in chemical reaction networks that do not involve bimolecular or higher-order reactions has been studied extensively (see for example [68]) and chemical reactions have also been analyzed as analog signal processing systems [63]. In this section, we will study reactions where two or more reactants are noisy, and their disturbances may be correlated with each other.

5.7.1 Motivating Example

Consider the following reaction:

$$X + Y \to Z. \tag{5.106}$$

Further assume that the concentration of X and Y is subject to random white noise fluctuations around a deterministic mean value:

$$\begin{aligned} X_t &= X_0 + \sigma_X dU_t \\ Y_t &= Y_0 + \sigma_Y dW_t \end{aligned} \tag{5.107}$$

and Z degrades with a rate proportional to its concentration. The corresponding stochastic differential equation is

$$\begin{aligned} dZ &= (X_0 Y_0 - aZ_t)dt + d(X_t Y_t) \\ &= (X_0 Y_0 - aZ_t)dt + X_0 \sigma_Y dW_t + Y_0 \sigma_X dU_t + \sigma_X \sigma_Y d[U_t, W_t], \end{aligned} \tag{5.108}$$

where U_t and W_t are standard Wiener processes. and $dU_t dW_t = d[U_t, W_t]$ is the differential of the quadratic covariation of U_t and W_t. Equation (5.108) is a natural generalization of the case where we have only one or more noise terms that are added to the deterministic differential equation. In all stochastic differential equations so far, we multiply the deterministic factors that contribute to the infinitesimal change in the state of the system by dt, and then we add the noise terms. In equation (5.108) the deterministic part is $X_0 Y_0$ and the rest is noise. If the two processes have correlation ρ, then

$$d[U_t, W_t] = \rho dt. \tag{5.109}$$

Simplifying the last expression for dZ,

$$dZ = (X_0 Y_0 + \rho \sigma_X \sigma_Y - aZ_t)dt + X_0 \sigma_Y dW_t + Y_0 \sigma_X dU_t \tag{5.110}$$

which is the familiar Ornstein−Uhlenbeck process with two noise sources. The final expression for the concentration of Z is

$$Z(t) = \frac{1}{a}(X_0 Y_0 + \rho \sigma_X \sigma_Y)(1 - e^{-at}) + \sigma_X Y_0 \int_0^t e^{a(t-s)} dU_s + \sigma_Y X_0 \int_0^t e^{a(t-s)} dW_s. \tag{5.111}$$

As the effect of the initial conditions diminishes, the mean is

$$\bar{Z} = \lim_{t \to \infty} \mathbb{E}[Z(t)] = \frac{1}{a}(X_0 Y_0 + \rho \sigma_X \sigma_Y) \tag{5.112}$$

and the variance is equal to

$$\mathbb{V}[Z] = \lim_{t \to \infty} \mathbb{V}[Z(t)] = \frac{Y_0^2 \sigma_X^2 + X_0^2 \sigma_Y^2 + 2 X_0 Y_0 \rho \sigma_X \sigma_Y}{2a}. \tag{5.113}$$

An important consequence of correlations in the input noise ($\rho \neq 0$) is that the mean is different from the case where there is no noise, even if both noise terms in (5.107) have a mean of zero. In addition, the variance is larger when there are positive correlations

in the two input noise terms, as expected. When the correlation is negative, the two noise processes partially cancel each other, resulting in lower variance.

5.7.2 General Reactions

We can generalize the above results to general reactions of the form

$$a_1 X_1 + \cdots + a_N X_N \rightarrow b_1 Y_1 + \cdots + b_M Y_M \tag{5.114}$$

where each of the elements on the left-hand side is assumed to be a random variable that consists of a deterministic mean $\bar{X}_k$ and a standard white noise process $dW_t^{(k)}$ multiplied by the standard deviation of its concentration.

$$X_k(t) = \bar{X}_k + \sigma_k dW_t^{(k)} \qquad 1 \leq k \leq N. \tag{5.115}$$

The concentration of the product Y_j is described by a stochastic differential equation:

$$\begin{aligned} dY_j = &\left(b_j \prod_{u=1}^{N} \bar{X}_u - f_j Y_j \right) dt + b_j \sum_{k=1}^{N} \sigma_k \left(\prod_{\substack{u=1 \\ u \neq k}}^{N} \bar{X}_u \right) dW_t^{(k)} \\ &+ b_j \sum_{k=1}^{N} \sum_{m=1}^{N} \sigma_k \sigma_m \left(\prod_{\substack{u=1 \\ u \neq k,m}}^{N} \bar{X}_u \right) \rho_{k,m} dt + \mathrm{O}\left(dt\right). \end{aligned} \tag{5.116}$$

The last equation is derived by using Itô's box rule [26], and the fact that higher order products of Wiener processes have variances that tend to zero faster than dt as $dt \rightarrow 0$. As in the bimolecular case, we multiply the noiseless input by dt, and then we add all the noise terms, and their products. Disregarding the initial conditions,

we compute the mean:

$$\mathbb{E}[Y_j] = \frac{b_j}{f_j}\left(\prod_{u=1}^{N}\bar{X}_u + \sum_{k=1}^{N}\sum_{m=1}^{N}\sigma_k\sigma_m\left(\prod_{\substack{u=1\\u\neq k,m}}^{N}\bar{X}_u\right)\rho_{k,m}\right) \tag{5.117}$$

which is different from the case when there is no noise, if there are correlations among the noise terms. The last equation clearly shows that noisy inputs can have an effect in the average of the concentration of the output, even if their mean is zero. The amount by which they shift the mean depends on their own variances, their correlations, and the product of concentrations of all *other* reactants. The variance is equal to

$$\mathbb{V}[Y_j] = \frac{b_j^2}{2f_j}\left(\sum_{k=1}^{N}\sigma_k^2\prod_{\substack{u=1\\u\neq k}}^{N}\bar{X}_u^2 + \sum_{k<m}2\rho_{km}\sigma_k\sigma_m\sigma_k^2\prod_{\substack{u=1\\u\neq k,m}}^{N}\bar{X}_u^2\right). \tag{5.118}$$

As before, positive correlations increase variance, negative correlations reduce it, and the extent by which the correlations affect it depends on the concentrations of the other species in the reaction.

5.7.3 Reactions with Filtered Noise

Suppose we have the following simple reaction:

$$X_1 + X_2 + \cdots + X_n \to Y \tag{5.119}$$

where $X_1 \ldots X_N$ fluctuate around an average value, but the noise has already passed through a linear filter. The differential equation that Y satisfies is:

$$dY = -aYdt + \prod_{u=1}^{N}\left(\bar{X}_k + \sigma_k\int_0^t h_k(t-s)dW_s^k\right)dt. \tag{5.120}$$

The terms dW_t^k are standard Wiener processes, each corresponding to the respective species k. We expand the last equation:

$$\begin{aligned} dY = &\left(\prod_{u=1}^{N} \bar{X}_u - aY\right) dt + \sum_{k=1}^{N} \sigma_k \prod_{\substack{u=1 \\ u\neq k}}^{N} \bar{X}_u dt \int_0^t h_k(t-s) dW_s^k \\ &+ \sum_{k=1}^{N}\sum_{m=1}^{N} \sigma_k \sigma_m \prod_{\substack{u=1 \\ u\neq k,m}}^{N} \bar{X}_u dt \int_0^t \int_0^t h_k(t-x) h_m(t-y) dW_x^k dW_y^m \\ &+ \mathrm{O}\left(dt\right). \end{aligned} \tag{5.121}$$

We have omitted all the terms whose order is larger than dt as $dt \to 0$, gathering them under the term $\mathrm{O}\left(dt\right)$. By using Itô's box rule again, we can replace the products of Wiener processes by their correlation times the infinitesimal time interval dt.

$$\begin{aligned} dY = &\left(\prod_{u=1}^{N} \bar{X}_u - aY\right) dt + \sum_{k=1}^{N} \sigma_k \prod_{\substack{u=1 \\ u\neq k}}^{N} \bar{X}_u dt \int_0^t h_k(t-s) dW_s^k \\ &+ \sum_{k=1}^{N}\sum_{m=1}^{N} \sigma_k \sigma_m \prod_{\substack{u=1 \\ u\neq k,m}}^{N} \bar{X}_u dt \int_0^t \rho_{km} h_k(t-x) h_m(t-x) dx \\ &+ \mathrm{O}\left(dt\right). \end{aligned} \tag{5.122}$$

Note that the second sum of integrals is deterministic and does not depend on any Wiener process. Setting

$$\begin{aligned} f(t) &= \sum_{k=1}^{N}\sum_{m=1}^{N} \sigma_k \sigma_m \prod_{\substack{u=1 \\ u\neq k,m}}^{N} \bar{X}_u dt \int_0^t \rho_{km} h_k(t-x) h_m(t-x) dx \\ c_N &= \prod_{u=1}^{N} \bar{X}_u \quad , \quad \hat{\sigma}_k = \sigma_k \prod_{\substack{u=1 \\ u\neq k}}^{N} \bar{X}_u \qquad \text{and} \\ q_k(t) &= \hat{\sigma}_k \int_0^t h_k(t-s) dW_s^k, \end{aligned} \tag{5.123}$$

the solution to the last differential equation (with zero initial conditions) is

$$Y(t) = c_N(1 - e^{-at}) + \int_0^t e^{-a(t-u)} f(u)du + \sum_{k=1}^{N} \int_0^t e^{-a(t-u)} q_k(u)du. \tag{5.124}$$

More generally, if the differential equation for the output has impulse response $g(t)$, and initial condition Y_0,

$$Y(t) = Y_0 g(t) + c_N \int_0^t g(t-u)du + \int_0^t g(t-u)f(u)du + \sum_{k=1}^{N} \int_0^t g(t-u)q_k(u)du, \tag{5.125}$$

where all terms except for the last sum are deterministic. The last equation nicely decomposes the factors that drive the output $Y(t)$. The first term is the effect of the initial conditions, the second term denotes the effect of the mean value of the inputs, the third term results from the noise correlations of the inputs, and the last term corresponds to the sum of the random fluctuations of all input sources.

If the output of reaction (5.126) receives inputs that are affected by both filtered and unfiltered disturbances, then we can use the same methods to find the mean and standard deviation of the output. We will analyze the case where we have two inputs, one of each type. The generalization to an arbitrary number of inputs is straightforward. Suppose that the chemical species Y depends on species X_1 and X_2

$$X_1 + X_2 \to Y \tag{5.126}$$

where the inputs X_1 and X_2 are defined by the following differential equations:

$$X_1(t) = \bar{X}_1 + \sigma_1 \int_0^t h(t-s)dU_s \tag{5.127}$$

and

$$X_2(t) = \bar{X}_2 + \sigma_2 dW_t, \tag{5.128}$$

where U_t and W_t are standard Wiener processes. The stochastic differential equation for Y is

$$\begin{aligned} dY &= (\bar{X}_1\bar{X}_2 - aY)dt + \sigma_2\bar{X}_1 dW_t + \sigma_1\bar{X}_2 dt \int_0^t h(t-s)dU_s \\ &\quad + \sigma_1\sigma_2 \int_0^t h(t-s)dU_s dW_t \\ &= (\bar{X}_1\bar{X}_2 + \rho h_0\sigma_1\sigma_2 - aY)dt + \sigma_2\bar{X}_1 dW_t + \sigma_1\bar{X}_2 \int_0^t h(t-s)dU_s \end{aligned} \tag{5.129}$$

since

$$dU_s dW_t = \begin{cases} \rho dt & \text{if } s = t \\ \\ 0 & \text{otherwise.} \end{cases} \tag{5.130}$$

The output is equal to

$$\begin{aligned} Y(t) = Y_0 e^{-at} + (\bar{X}_1\bar{X}_2 + \rho h_0\sigma_1\sigma_2)(1 - e^{-at}) + \sigma_2\bar{X}_1 \int_0^t e^{-a(t-s)} dW_s \\ + \sigma_1\bar{X}_2 \int_0^t \int_0^s e^{a(t-s)} h(s-x)dU_x. \end{aligned} \tag{5.131}$$

The mean is

$$\mathbb{E}[Y] = \frac{1}{a}\left(\bar{X}_1\bar{X}_2 + \rho h_0\sigma_1\sigma_2\right), \tag{5.132}$$

which differs from the noiseless case by the last term, which is proportional to the correlation and the standard deviation of the noise inputs. Similarly, the variance is found to be equal to

$$\mathbb{V}[Y(t)] = V_1(t) + V_2(t) + V_{12}(t), \tag{5.133}$$

where

$$V_1(t) = e^{-2at}\sigma_1^2\bar{X}_2^2 \int_0^t \int_0^t e^{a(r+s)} \left(\int_0^{r\wedge s} h(s-u)h(r-u)du \right) drds, \tag{5.134}$$

$$V_2(t) = \frac{\sigma_2^2 \bar{X}_1^2}{2a}(1 - e^{-2at}), \tag{5.135}$$

and

$$V_{12}(t) = \rho\sigma_1\sigma_1\bar{X}_1\bar{X}_2 \int_0^t e^{-a(t-y)} \left(\int_0^{t\wedge y} e^{-a(t-x)} h(y-x)dx \right) dy. \tag{5.136}$$

The first component $V_1(t)$ is the variance because of the noise in the first input dU_t, $V_2(t)$ the variance because of noise in the second input, and the last term $V_{12}(t)$ is the variance emanating from their correlation.

When the inputs X_1 and X_2 in (5.126) both have a filtered multiplicative noise component, then the differential equation becomes

$$dY = -aYdt + \left(\bar{X}_1\lambda_1 \int_0^t e^{-(\lambda_1 + \frac{\sigma_1^2}{2})x} e^{\sigma_1 U_x} dx \right) \left(\bar{X}_2\lambda_2 \int_0^t e^{-(\lambda_2 + \frac{\sigma_1^2}{2})y} e^{\sigma_1 W_y} dy \right) dt. \tag{5.137}$$

In order to account for the possibly nonzero correlation between processes U_t and W_t, we write each of them as a sum of two uncorrelated standard processes:

$$\begin{aligned} U_t &= aA_t + \sqrt{1-a^2}B_t \\ W_t &= bA_t + \sqrt{1-b^2}C_t. \end{aligned} \tag{5.138}$$

The processes A_t, B_t and C_t have correlation zero, and $\rho = ab$ is the correlation between U_t and W_t :

$$-1 \le a \le 1, \quad -1 \le b \le 1 \quad \text{and} \quad -1 \le \rho \le 1. \tag{5.139}$$

We are interested in finding the mean and variance of Y. First, we compute the

expected value of the product of the two exponential Wiener processes U_t and W_t.

$$\begin{aligned}\mathbb{E}\left[e^{\sigma_1 U_x} e^{\sigma_2 W_y}\right] &= \mathbb{E}\left[e^{\sigma_1\left(aA_x+\sqrt{1-a^2}B_x\right)} e^{\sigma_2\left(bA_y+\sqrt{1-b^2}C_y\right)}\right] \\ &= \mathbb{E}\left[e^{a\sigma_1 A_x + b\sigma_2 A_y}\right] \cdot \mathbb{E}\left[e^{\sigma_1\sqrt{1-a^2}B_x}\right] \cdot \mathbb{E}\left[e^{\sigma_2\sqrt{1-b^2}C_y}\right] \\ &= e^{\frac{1}{2}(a\sigma_1+b\sigma_2)^2 x\wedge y} e^{\frac{1}{2}\left((a\sigma_1)^2\delta_x+(b\sigma_2)^2\delta_y\right)(x\vee y - x\wedge y)} e^{\frac{1}{2}\sigma_1^2\left(1-a^2\right)x} e^{\frac{1}{2}\sigma_2^2\left(1-b^2\right)y}\end{aligned} \tag{5.140}$$

where δ denotes the Kronecker delta with $\delta_x = \delta(x \geq y)$ and $\delta_y = \delta(y \geq x)$.

The expected value of the input of the differential equation is

$$\begin{aligned}\mathbb{E}[u(t)] &= \mathbb{E}\left[\left(\bar{X}_1 \int_0^t e^{-(\lambda_1+\frac{\sigma_1^2}{2})x} e^{\sigma_1 U_x} dx\right)\left(\bar{X}_2 \int_0^t e^{-(\lambda_2+\frac{\sigma_2^2}{2})y} e^{\sigma_2 W_y} dy\right)\right] \\ &= \lambda_1\lambda_2\bar{X}_1\bar{X}_2 \int_0^t e^{-(\lambda_1+\frac{\sigma_1^2}{2})x}\left(\int_0^t e^{-(\lambda_2+\frac{\sigma_2^2}{2})y}\mathbb{E}\left[e^{\sigma_1 U_x} e^{\sigma_2 W_y}\right] dy\right) dx \\ &= \lambda_1\lambda_2\bar{X}_1\bar{X}_2 \int_0^t e^{-\lambda_1 x}\left(\int_0^x e^{-(\lambda_2-\rho\sigma_1\sigma_2)y} dy\right) dx \\ &\quad + \lambda_1\lambda_2\bar{X}_1\bar{X}_2 \int_0^t e^{-(\lambda_1+\rho\sigma_1\sigma_2)x}\left(\int_x^t e^{-\lambda_2 y} dy\right) dx \\ &= \lambda_2\bar{X}_1\bar{X}_2 \frac{e^{-t\lambda_1}\left(\left(1-e^{\rho t\sigma_2\sigma_1 - t\lambda_2}\right)\lambda_1 + \left(-1+e^{t\lambda_1}\right)\left(\rho\sigma_2\sigma_1 - \lambda_2\right)\right)}{\left(\rho\sigma_2\sigma_1-\lambda_2\right)\left(-\rho\sigma_2\sigma_1+\lambda_1+\lambda_2\right)} \\ &\quad + \lambda_1\bar{X}_1\bar{X}_2 \frac{e^{-t\lambda_2}\left(\left(1-e^{t\lambda_2}\right)\rho\sigma_2\sigma_1 - \left(1-e^{t\lambda_2}\right)\lambda_1 - \left(1-e^{\rho t\sigma_2\sigma_1 - t\lambda_1}\right)\lambda_2\right)}{\left(\rho\sigma_2\sigma_1-\lambda_1\right)\left(\rho\sigma_2\sigma_1-\lambda_1-\lambda_2\right)}\end{aligned} \tag{5.141}$$

where we assume that

$$\lambda_1 > \frac{\sigma_1^2}{2} \quad , \quad \lambda_2 > \frac{\sigma_2^2}{2} \implies \lambda_1+\lambda_2 > \rho\sigma_1\sigma_2. \tag{5.142}$$

The inequalities above guarantee that the inputs have finite variances, as shown in equation (5.88). In the equilibrium state,

$$\lim_{t\to\infty} \mathbb{E}\left[u(t)\right] = \bar{X}_1\bar{X}_2 \frac{\lambda_1+\lambda_2}{\left(\lambda_1+\lambda_2-\rho\sigma_1\sigma_2\right)}. \tag{5.143}$$

The output average is then equal to

$$\mathbb{E}[Y] = \lim_{t\to\infty} \mathbb{E}\left[Y(t)\right] = \frac{\bar{X}_1\bar{X}_2}{a} \cdot \frac{\lambda_1 + \lambda_2}{(\lambda_1 + \lambda_2 - \rho\sigma_1\sigma_2)}. \tag{5.144}$$

The last equation clearly shows that if the input noise sources are correlated ($\rho \neq 0$), the average value of the output will be different from the value when there is no correlation ($\rho = 0$). As shown in the other types of noise, positive correlations increase the mean, and negative correlations reduce it. The variance can be computed using the same methods. First, we will calculate the expected value of a product of different instances of a standard Wiener process.

Lemma 35. *If $t_1, t_2, \ldots t_n \in \mathbb{R}^+$ is an ordered set of times such that $t_1 \leq t_2 \leq \ldots \leq t_n$ and $\sigma_1, \sigma_2 \ldots \sigma_n \in \mathbb{R}^+$ are arbitrary positive numbers denoting standard deviations, then*

$$\mathbb{E}\left[\prod_{k=1}^{n} e^{\sigma_k W_{t_k}}\right] = \exp\left[\frac{1}{2}\sum_{k=1}^{n}\left(\sum_{m=k}^{n}\sigma_m\right)^2 (t_k - t_{k-1})\right] \tag{5.145}$$

where W_t is the standard Wiener process.

Proof. For each t_k, we decompose the Wiener process W_{t_k} as a sum of independent processes:

$$W_{t_k} = \sum_{m=1}^{k} \left(W_{t_m} - W_{t_{m-1}}\right). \tag{5.146}$$

Based on this sum, we can write

$$\begin{aligned}
\prod_{k=1}^{n} e^{\sigma_k W_{t_k}} &= \exp\left[\sum_{k=1}^{n}\sigma_k W_{t_k}\right] \\
&= \exp\left[\sum_{k=1}^{n}\sigma_k \sum_{m=1}^{k}\left(W_{t_m} - W_{t_{m-1}}\right)\right] \\
&= \exp\left[\sum_{k=1}^{n}\left(W_{t_k} - W_{t_{k-1}}\right)\sum_{m=k}^{n}\sigma_k\right] \\
&= \prod_{k=1}^{n}\exp\left[\left(W_{t_k} - W_{t_{k-1}}\right)\sum_{m=k}^{n}\sigma_k\right]
\end{aligned} \tag{5.147}$$

where in the last equation, we changed the order of summation making use of the triangle rule. All terms in the last product are independent:

$$\begin{aligned}
\mathbb{E}\left[\prod_{k=1}^{n} e^{\sigma_k W_{t_k}}\right] &= \mathbb{E}\left[\prod_{k=1}^{n} \exp\left[\left(W_{t_k} - W_{t_{k-1}}\right)\sum_{m=k}^{n}\sigma_k\right]\right] \\
&= \prod_{k=1}^{n} \mathbb{E}\left[\exp\left[\left(W_{t_k} - W_{t_{k-1}}\right)\sum_{m=k}^{n}\sigma_k\right]\right] \\
&= \prod_{k=1}^{n} \exp\left[\frac{1}{2}\left(t_k - t_{k-1}\right)\left(\sum_{m=k}^{n}\sigma_k\right)^2\right] \\
&= \exp\left[\frac{1}{2}\sum_{k=1}^{n}\left(\sum_{m=k}^{n}\sigma_k\right)^2\left(t_k - t_{k-1}\right)\right].
\end{aligned} \tag{5.148}$$

□

When one of the inputs is affected by multiplicative noise, and the other by additive noise, the mean value of the output is not affected, even if the driving noise is the same in both cases. The differential equation of the chemical reaction (5.126) becomes

$$\frac{dY}{dt} = -aY + \left(\lambda_1 \bar{X}_1 \int_0^t e^{-(\lambda_1 + \frac{\sigma^2}{2})x} e^{\sigma W_x} dx\right)\left(\bar{X}_2 + \sigma \int_0^t e^{-a(t-y)} dW_y\right). \tag{5.149}$$

The input is equal to

$$u(t) = \left(\lambda_1 \bar{X}_1 \int_0^t e^{-(\lambda_1 + \frac{\sigma^2}{2})x} e^{\sigma W_x} dx\right)\left(\bar{X}_2 + \sigma \int_0^t e^{-\lambda_2(t-y)} dW_y\right) \tag{5.150}$$

and its expected value is

$$\begin{aligned}
\mathbb{E}[u(t)] = \lambda_1 \bar{X}_1 \bar{X}_2 &\int_0^t e^{-(\lambda_1 + \frac{\sigma^2}{2})x} \mathbb{E}\left[e^{\sigma W_x}\right] dx \\
&+ \sigma \lambda_1 \bar{X}_1 \int_0^t \int_0^t e^{-(\lambda_1 + \frac{\sigma^2}{2})x} e^{-\lambda_2(t-y)} \mathbb{E}\left[e^{\sigma W_x} dW_y\right] dx.
\end{aligned} \tag{5.151}$$

In order to compute the second term of the last equation, we will need the following Lemma about the expected value of the product of an exponential Wiener process with an infinitesimal difference of the same process.

Lemma 36. *If W_t is a standard Wiener process, then*

$$\mathbb{E}\left[e^{\sigma W_s} dW_t\right] = \begin{cases} 0 & \text{if } s \leq t \\ \sigma^2 e^{\frac{\sigma^2}{2}s} dt & \text{if } s > t. \end{cases} \tag{5.152}$$

Proof. If $s < t$, then W_s and $dW_t = W_{t+dt} - W_t$ are uncorrelated, so

$$\mathbb{E}\left[e^{\sigma W_s} dW_t\right] = \mathbb{E}\left[e^{\sigma W_s}\right] \mathbb{E}\left[dW_t\right] = 0. \tag{5.153}$$

If a, b are two positive real numbers such that $0 < a < b < s$, then

$$\begin{aligned} \mathbb{E}\left[e^{\sigma W_s}(W_b - W_a)\right] &= \mathbb{E}\left[e^{\sigma W_a}\right] \mathbb{E}\left[e^{\sigma(W_b - W_a)}(W_b - W_a)\right] \mathbb{E}\left[e^{\sigma(W_s - W_b)}\right] \\ &= e^{\frac{1}{2}\sigma^2 a} e^{\frac{1}{2}\sigma^2(b-a)} \sigma^2 (b-a) e^{\frac{1}{2}\sigma^2(s-b)} \\ &= \sigma^2 e^{\frac{1}{2}\sigma^2 s}(b-a). \end{aligned} \tag{5.154}$$

Setting $a = t$ and $b = t + dt$, we get the desired result. □

Recalling equation (5.151),

$$\begin{aligned} \mathbb{E}[u(t)] &= \lambda_1 \bar{X}_1 \bar{X}_2 \int_0^t e^{-\lambda_1 x} dx + \sigma^3 \lambda_1 \bar{X}_1 e^{-\lambda_2 t} \int_0^t \left(\int_y^t e^{-\lambda_1 x} e^{\lambda_2 y} dx \right) ds \\ &= \bar{X}_1 \bar{X}_2 \left(1 - e^{-\lambda_1 t}\right) + \sigma^3 \bar{X}_1 \frac{e^{-t(\lambda_1+\lambda_2)} \left(\lambda_1 \left(1 - e^{t\lambda_2}\right) - \left(1 - e^{t\lambda_1}\right) \lambda_2\right)}{\lambda_2 \left(\lambda_1 - \lambda_2\right)}. \end{aligned} \tag{5.155}$$

As time t grows large,

$$\lim_{t \to \infty} \mathbb{E}\left[u(t)\right] = \bar{X}_1 \bar{X}_2 \tag{5.156}$$

and the mean of the output is

$$\mathbb{E}[Y] = \frac{1}{a}\bar{X}_1\bar{X}_2 \tag{5.157}$$

which is exactly the same as in the case where the two noise inputs are completely uncorrelated. So, input noise correlation does not affect the average concentration of the output in this case.

In this section we have analyzed how noise propagates in an arbitrary chemical reaction network where one or more inputs include a random component, either additive or geometric noise. The different noise sources may have arbitrary correlations with each other. One of the main results is that even if all noise sources have mean equal to zero, their correlations may shift the mean of the outputs, for both types of noise. If there is positive correlation, the mean of the output increases, and when the correlation is negative, it shifts lower, and the same is true for the output variance.

5.8 Conclusions

In this chapter, we have studied how noise propagates in networks and how a network's noisy parameters can affect its output. Since many biological networks are locally tree-like, we have shown how noise propagates in the absence of feedforward or feedback cycles. Tree networks are relatively easy to quantitatively analyze, since there is only one path from each node to another. We have derived a method to compute the variance of the output of any tree network, and shown that the variance is minimized when there are no "bottlenecks" in each pathway, in other words when there is no rate limiting step. When a network is not a tree, there are cycles, which means that a signal can propagate through two or more paths towards the output. Feedback cycles typically reduce the output variance, and feedforward cycles increase it. When the noise sources are correlated, the variance in the output is larger, and small cycles have a stronger influence on the output, compared to longer cycles in both cases. Delays contribute to the decrease of the output noise when we have two or more

noise sources, since their correlation is usually reduced. Crosstalk is also shown to decrease the output variance, but the trade-off is that the output mean is lowered, or the concentration of the inputs needs to be proportionally higher in order to ensure the same output.

In biological and chemical reaction networks, the reaction rates are prone to noise, since they depend on the concentration of other species. When the degradation rates are affected by noise, the result is increased output variance, which also depends on the concentration of the respective species, and the form of the output is different from when the noise is in the inputs, in the sense that higher concentrations also correspond to larger deviations from the mean. Finally, we have extensively studied how noise propagates through chemical reaction networks where one or more of the reactants are noisy, and their disturbances may be correlated. Even when the disturbances have zero average, correlations may change the output mean and variance.

Chapter 6

Summary and Future Directions

6.1 Concluding Remarks

This thesis consists of two parts, the first one is devoted to the study of the structural properties of arbitrary networks, and the second on their applications. In the first part (Chapter 3), we found the structure of the networks that minimize or maximize a variety of important structural properties, including the graph's radius, diameter, efficiency, average clustering, average distance, betweenness centrality, and resistance. We found similarities and differences among the networks that achieve these extremes. Sometimes the relation between different properties is unexpected. For example, we have found that the betweenness centrality of a network is a function of its average distance, and consequently, networks with minimum or maximum average distance also have minimum or maximum betweenness centrality, respectively. On the other hand, networks with the maximum radius are not necessarily the same as networks with the maximum diameter. In most cases, the structures that maximize a property are unique, and the structures that minimize it are not, although there are notable exceptions, like the network's efficiency. Generally, the form of the graphs that maximize each of the aforementioned properties is very special, in the sense that most random graphs of the same order and size will tend to be very close to the theoretical minimum. Furthermore, these architectures are quite sensitive to the addition of

new edges, or rewiring existing edges, since even a few such changes can have a large impact on reducing each property towards its minimum. These structural properties can alternatively be used as a metric of how random a network is, or alternatively how many networks with similar architectures may exist.

In the second part (Chapters 4 and 5) we have studied crosstalk and noise and the way they affect the function of networks. We have considered two cases of crosstalk, as a function of the number of connections in each of the nodes. In both cases, assortative networks are more immune to crosstalk than disassortative ones. Networks that minimize crosstalk also have a large clustering coefficient. Crosstalk leads to smaller available concentrations of each reactant, and makes the system slower to respond to changes in the inputs. On the other hand, although crosstalk may degrade the performance of a system, it may help reduce noise. We explore the trade-offs between crosstalk and noise in biological and chemical reaction networks, and find that the form of the networks with the smallest average variance have cycles of the largest possible length, given their order and size. Consequently, these networks have minimum average clustering, in contrast to the networks that minimize crosstalk. Finally, we have explored the effect of parameter noise in the output variance. Noise in the parameters of a network, although it has the same sources, has a completely different effect on the output, leading to noise that is harder to control with the application of feedback.

6.2 Future Directions

There are several new directions that might be pursued based on the findings presented in this thesis. An obvious application of the theories developed here is the study of the relation between structure and function of biological or other natural and engineered networks. We can try to find the relative importance of each property of these networks based on their structure, and how close they are to the theoretical

extremes, or to the expected value of a random network. This would be useful both for the study of existing networks, as well as for the design of new ones.

The bounds of structural properties, along with the sensitivity of the respective architectures to rewiring can be the basis for quantifying the structural entropy of a graph. So far, most studies have defined the entropy of a network only as a function of its degree distribution [3, 16]. Other studies have used automorphisms to measure entropy [72]. The first definition leaves out almost all the information about the topology of a network, whereas the second makes it hard to measure it, or to compare the various architectures. Knowing the minimum, maximum and average of each structural property makes it easy to assess how much randomness there is in the topology of a network. The number of rewirings needed to transform a network to another one which statistically has the same properties as a random graph, can be a measure of the amount of randomness that needs to be introduced in order to maximize its entropy.

As it has been mentioned before, although noise is usually unwanted, and cells employ several strategies to keep it low [42, 67], sometimes it is useful, since it is the driving force behind cell differentiation, gene expression coordination, and evolution in general [23]. In terms of network structure in biological networks, it would be interesting to find structures that isolate noise in specific subsystems, and diminish it in other parts of the network, such that processes that need to be stable are actually stable, and processes that need variation are noisy. Furthermore, we have seen that crosstalk and noise are orthogonal notions as far as the function and the structure of biological networks is concerned, based on the models used in this study. A good question would be to determine the optimal trade-off between crosstalk and noise, as a function of the order and the size of a network, along with its specific properties, and then compare it to data from real biological networks. Also, we may try to study how negative correlations among noise sources are taken advantage of in order to reduce or increase noise, as shown in Chapter 5.

Engineers building communication systems have long been using elaborate coding schemes to transmit information. First, a message is encoded by the sender, then transmitted and eventually decoded by the receiver. The coding is used in order to add redundancy in the message, so that the recipient is able to decode it despite changes in its content due to random fluctuations in the communication channel. This need arises from random changes in the transmitted information while it travels through noisy channels. It is conceivable that nature might use the same mechanisms in order to add redundancy, and increase the robustness of information propagation. In [21], the authors present an interesting way in which biology can potentially encode information through multiplexing. Crosstalk may be an easy way in which nature encodes signals, since it makes the information more robust to changes both because it has the potential to reduce the amount of noise, and by introducing redundancy in the message.

There are a lot of experimental studies in biology focusing on input noise and how it affects specific systems (e.g. [23, 42, 56]). It would be interesting to experimentally study how fluctuations on the reaction rates affect biological networks, and how they translate to geometric noise in the outputs. From a theoretical point of view, we may also try to quantify the properties of the noise when we use a Poisson process, instead of white noise, as in [55]. This would be useful for systems where the number of proteins that affect the reaction rates is small.

The structural properties of natural and engineered networks are rarely the only things that matter in a network. The way that structure helps achieve the system's function is the ultimate question both for evolution and engineering. As a result, we need to find a general framework in which we can study the relations between the dynamics of a network and its structure. Specifically, we need to find quantitative measures for how each desirable or undesirable network property is affected by structure. When this is done, we can readily optimize the network at hand based on the imposed constraints and the relative desirability of various properties. A par-

ticularly interesting case would be to study how these properties affect information propagation when each node can assume a binary state. Specific examples include neuronal networks, where neurons either fire or don't fire, fashion preferences, disease, and rumor propagation. Finding general rules on these types of networks would have important consequences in many scientific fields.

Bibliography

[1] Albert, R., Jeong, H., and Barabási, A.L. *Error and attack tolerance of complex networks*, Nature **406**, 378–382 (2000)

[2] Alon, U. *An Introduction to Systems Biology: Design Principles of Biological Circuits*, Chapman and Hall/CRC (2006)

[3] Anand, K., Bianconi, G., and Severini, S. *Shannon and von Neumann entropy of random networks with heterogeneous expected degree*, Physical Review E **83**, 036109 (2011)

[4] Argollo de Menezes, M., and Barabási, A.L. *Separating internal and external dynamics of complex systems*, **93**, 68701 (2004)

[5] Ash, J. and Newth, D. *Optimizing complex networks for resilience against cascading failure*, Physica A: Statistical Mechanics and Its Applications **380**, 673–683 (2007)

[6] Åström, K.J. and Murray, R.M. *Feedback Systems: An Introduction for Scientists and Engineers*, Princeton University Press (2008)

[7] Banavar, J.R., Maritan, A., and Rinaldo, A. *Size and form in efficient transportation networks*, Physica A: Statistical Nature **399**, 130–131 (1999)

[8] Barabási, A.L. and Oltvai, Z.N. *Network biology: Understanding the cell's functional organization*, Nature Reviews Genetics **5**, 101–113 (2004)

[9] Barkai, N. and Leibler, S. *Robustness in simple biochemical networks*, Nature **387**, 913–917 (1997)

[10] Barmpoutis, D. and Murray, R.M. *Networks with the smallest average distance and the largest average clustering*, arXiv:1007.4031v1 (2010)

[11] Barmpoutis, D. and Murray, R.M. *Quantification and minimization of crosstalk sensitivity in networks*, arXiv:1012.0606v1 (2010)

[12] Barmpoutis, D. and Murray, R.M. *Extremal properties of complex networks*, arXiv:1104.5532v1 (2011)

[13] Barmpoutis, D. and Murray, R.M. *Noise propagation in biological and chemical reaction networks*, arXiv:1108.2538v1 (2011)

[14] Barthélemy, M. and Flammini, A. *Optimal traffic networks*, Journal of Statistical Mechanics: Theory and Experiment **07**, L07002 (2006)

[15] Berman, N. and Shaked, U. *Robust L_∞-Induced Filtering and Control of Stochastic Systems With State-Multiplicative Noise*, IEEE Transactions on Automatic Control **55**, 732–737 (2010)

[16] Bianconi, G. *Entropy of network ensembles*, Physical Review E **79**, 036114 (2009)

[17] Boyd, S.P. and Vandenberghe, L. *Convex Optimization*, Cambridge University Press (2004)

[18] Cohen, R., Erez, K., ben-Avraham, D., and Havlin, S. *Breakdown of the internet under intentional attack*, Physical Review Letters **86**, 3682–3685 (2001)

[19] Colizza, V., Banavar, J., Maritan, A., and Rinaldo, A. *Network structures from selection principles*, Physical Review Letters **92**, 198701 (2004)

[20] Csete, M.E. and Doyle, J.C. *Reverse engineering of biological complexity*, Science **295** 1664–1669 (2002)

[21] De Ronde, W., Tostevin, F., and ten Wolde, P.R. *Multiplexing biochemical signals*, arXiv:1008.0576 (2010)

[22] Doyle, P.G. and Snell, J.L. *Random Walks and Electric Networks*, Mathematical Association of America, Washington, DC (1984)

[23] Eldar, A. and Elowitz, M.B. *Functional roles for noise in genetic circuits*, Nature **467**, 167–173 (2010)

[24] Elowitz, M.B. and Leibler, S. *A synthetic oscillatory network of transcriptional regulators*, Nature **403**, 335–338 (2000)

[25] Franco, E. et al. *Timing molecular motion and production with a synthetic transcriptional clock*, Proceedings of the National Academy of Sciences **108**, E784–793 (2011)

[26] Gardiner, C.W. *Handbook of Stochastic Methods for Physics, Chemistry, and the Natural Sciences*, Springer (1983)

[27] Girvan, M. and Newman, M.E.J. *Community structure in social and biological networks*, Proceedings of the National Academy of Sciences **99**, 7821–7826 (2002)

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

[29] Granovetter, M.S. *The strength of weak ties*, American Journal of Sociology **78**, 1360–1380 (1973)

[30] Grindrod, P. *Range-dependent random graphs and their application to modeling large small-world proteome datasets*, Physical Review E **66**, 066702 (2002)

[31] Gross, J.L. and Yellen, J. *Graph Theory and Its Applications*, CRC Press (2006)

[32] Hagberg, A.A. et al. *Exploring network structure, dynamics, and function using NetworkX*, Proceedings of the 7th Python in Science Conference, Gael Varoquaux, Travis Vaught, and Jarrod Millman (Eds.), pp. 11–15 (2008)

[33] Hakimi, S.L. *On realizability of a set of integers as degrees of the vertices of a linear graph*, Applied Mathematics **10**, 496–506 (1962)

[34] Jeong, H. *The large-scale organization of metabolic networks*, Nature **407**, 651–654 (2000)

[35] Kashtan, N. and Alon, U. *Spontaneous evolution of modularity and network motifs*, Proceedings of the National Academy of Sciences **102**, 13773–13778 (2005)

[36] Kashtan, N., Noor, E., and Alon, U. *Varying environments can speed up evolution*, Proceedings of the National Academy of Sciences **104**, 13711–13716 (2007)

[37] Kim, B.J. *Performance of networks of artificial neurons: The role of clustering*, Physical Review E **69**, 045101 (2004)

[38] Kim, J., White, K.S., and Winfree, E. *Construction of an in vitro bistable circuit from synthetic transcriptional switches*, Molecular Systems Biology **2**, 68 (2006)

[39] Kitano, H. *Biological robustness*, Nature Reviews Genetics **5**, 11 826–837 (2004)

[40] Lan, Y. and Mezic, I. *On the architecture of cell regulation networks*, BMC Systems Biology **5**, 37 (2011)

[41] Latora, V. and Marchiori, M. *Efficient behavior of small-world networks*, Physical Review Letters **87**, 198701 (2001)

[42] Lestas, I., Vinnicombe, G., and Paulsson, J. *Fundamental limits on the suppression of molecular fluctuations*, Nature **467**, 174–178 (2010)

[43] Li, W., Todorov, E., and Skelton, R.E. *Estimation and control of systems with multiplicative noise via linear matrix inequalities*, Proceedings of the 2005 American Control Conference 1811–1816 (2005)

[44] Liptser, R.S. and Shiryaev, A.N. *Statistics of Random Processes I: General Theory*, Springer (2000)

[45] Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, D. Chklovskii, D., and Alon, U. *Network motifs: simple building blocks of complex networks*, Science **298**, 824–827 (2002)

[46] Molloy M. and Reed, B. *A critical point for random graphs with a given degree sequence*, Rand. Struct. Algorithms **6**, 161–179 (1995)

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

[48] Newman, M.E.J. *Networks: An Introduction*, Oxford University Press (2010)

[49] Newman, M.E.J. *Random graphs with clustering*, Physical Review Letters **103**, 58701 (2009)

[50] Oikonomou, P. and Cluzel, P. *Effects of topology on network evolution*, Nature Physics **2**, 532–536 (2006)

[51] Olfati-Saber, R. and Murray, R.M. *Consensus problems in networks of agents with switching topology and time delays*, IEEE Transactions on Automatic Control **49**, 1520–1533 (2004)

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

[53] Pastor-Satorras, R. and Vespignani, A. *Epidemic spreading in scale-free networks*, Physical Review Letters **87**, 3200–3203 (2001)

[54] Paulsson, J. *Summing up the noise in gene networks*, Nature **427**, 415–418 (2004)

[55] Paulsson, J. and Ehrenberg, M. *Random signal fluctuations can reduce random fluctuations in regulated components of chemical regulatory networks*, Physical Review Letters **84**, 5447–5450 (2000)

[56] Pedraza, J.M. and van Oudenaarden, A. *Noise propagation in gene networks*, Science **307**, 1965–1969 (2005)

[57] Purnick, P.E.M. and Weiss, R. *The second wave of synthetic biology: from modules to systems*, Nature Reviews Molecular Cell Biology, **10**, 410–422 (2009)

[58] Qian, L. and Winfree, E. *A simple DNA gate motif for synthesizing large-scale circuits*, Journal of The Royal Society Interface, **62**, 1281–1297 (2011)

[59] Raj, A. and van Oudenaarden, A. *Nature, nurture, or chance: Stochastic gene expression and its consequences*, Cell **135**, 216–226 (2008)

[60] Rao, C.V., Wolf, D.M., and Arkin, A.P. *Control, exploitation and tolerance of intracellular noise*, Nature **420**, 231–237 (2008)

[61] Rezaei-Ghaleh, N.A. *Signaling in biological systems: Insights from communication theory*, Journal of Theoretical Biology **224**, 411–412 (2003)

[62] Rosner, B. *Fundamentals of Biostatistics*, Duxbury Press (2005)

[63] Samoilov, M., Arkin, A., and Ross, J. *Signal processing by simple chemical systems*, Journal of Physical Chemistry **106**, 10205–10221 (2002)

[64] Shinar, G., Dekel, E., Tlusty, T., and Alon, U. *Rules for biological regulation based on error minimization*, Proceedings of the National Academy of Sciences **103**, 3999–4004 (2006)

[65] Strogatz, S.H. *Exploring complex networks*, Nature **410**, 268–276 (2001)

[66] Variano, E.A., McCoy, J.H., and Lipson, H. *Networks, dynamics, and modularity*, Physical Review Letters **92**, 188701 (2004)

[67] Vilar, J.M.G., Kueh, H.Y., Barkai, N., and Leibler, S. *Mechanisms of noise-resistance in genetic oscillators*, Proceedings of the National Academy of Sciences **99**, 5988 (2002)

[68] Warren, P.B., Tanase-Nicola, S., and ten Wolde, P.R. *Exact results for noise power spectra in linear biochemical reaction networks*, The Journal of Chemical Physics **125**, 144904 (2006)

[69] Watts, D.J. *A simple model of global cascades on random networks*, Proceedings of the National Academy of Sciences **99**, 5766–5771 (2002)

[70] Watts, D.J. and Strogatz, S.H. *Collective dynamics of "small world" networks*, Nature **393**, 440–442 (1998)

[71] Weng, G., Bhalla, U.S., and Iyengar, R. *Complexity in biological signaling systems*, Science **284**, 92–96 (1999)

[72] Xiao, Y.H., Wu, W.T., Wang, H., Xiong, M., and Wang, W. *Symmetry-based structure entropy of complex networks*, Physica A: Statistical Mechanics and its Applications **387**, 2611–2619 (2008)