UvA-DARE (Digital Academic Repository)

Information processing in complex networks

Quax, R.

Publication date
2013

Link to publication

Citation for published version (APA):
Quax, R. (2013). *Information processing in complex networks*.

Chapter 1

Introduction

1.1 Background

1.1.1 The network description

In the eighteenth century in the Prussian city of Königsberg (now Kaliningrad, Russia), seven bridges connected the two sides of the Pregel River and two islands in its midst. A popular riddle among the people became known as the Königsberg bridge problem: is it possible to find a round trip that crosses each bridge exactly once? In 1736, the famous mathematician Leonard Euler invented a so-called 'graph theory' to prove that such a path did not exist. A graph, or network, is a generic mathematical object where nodes (or points) are connected by edges (or links). Though the concept of networks may seem trivial today, Euler's proof is one of the first known examples where a network description is used to solve a real problem. Nowadays it is one of the most powerful modeling techniques to study a wide variety of complex systems, such as cellular regulatory systems, brains, human behavior in social communities, spreading of epidemics, and financial trading markets.

In the 1950s, researchers studied the topological properties of 'random graphs', i.e., networks in which the edges are placed randomly between two nodes. The focus was a purely mathematical description of such random networks, with predictions such as the lengths of connected paths (1), the existence of cyclic paths (1), the sizes of connected clusters (1, 2), how many nodes should be removed to disconnect a network (3), and the probability that each node in the network is reachable by each other node (2). Suggested applications included brain networks (neurons connected by synapses), epidemic spreading (humans infecting other humans), and

transport networks (geographic locations connected by roads or rails). Although these works were a marked step forward in 'network thinking', the idealized random-mixing topology would later turn out to be an unrealistic description of real networks.

Near the end of the 1960s, empirical evidence of real-world networks started to be gathered and characterized, aided by the advent of the Internet and an ever-increasing computing power. The topology of real networks turned out to be quite different from the purely random networks in many respects. One of the first large networks to be characterized were citation networks among scientific publications in 1965 by Price (4–6). Price found that the number of edges (citations) that a research paper either contained or received was not sharply distributed around a mean value, as random networks would predict, but were distributed 'fairly flat'. That is, there are many more highly-cited papers than expected, and the number of papers that are cited k times eventually decreases to zero as k^{-a}. Price wondered whether 'the more a paper is cited, the more likely it is to be cited thereafter'. Three decades later, this 'heavy-tailed' distribution of connectivity and the 'rich get richer' organizing principle had been found in many other real networks, such as the World Wide Web, autonomous system networks, metabolic networks, telephone call graphs, and even networks of human sexual contacts. Many additional non-random characteristics of real network topologies were found around the same time, including assortativity, node centrality, clustering, and community structure. See References (7–9) for extensive reviews of network characteristics and relevant references.

The take-home message is that many real networks have a complicated structure. The study of how the network's structure shapes its function, and how its function in turn shapes the network structure, is now known as *complex networks* research.

1.1.2   From local processes to global behavior

So far we have only described real systems by the topology of interactions between the nodes. The scientific literature is rich with statistical analyses of such topologies owing to the availability of large and detailed datasets (10). Nonetheless, a second crucial ingredient is needed to understand the behavior of complex networks: the dynamical processes that govern the nodes. The two ingredients, structure and function, are intimately connected, but poorly understood.

Even if a simple process is placed in every node, the network of node-to-node interactions can lead to a complex behavior of the system as a whole. One of the first demonstrations was presented by Hopfield in 1982 (11) in the context of neural networks. He showed that a network of 100 identical nodes which have a certain 'on-off' dynamics (12) is capable of implementing error-tolerant content-addressable memory. Even today his concept is used as a model for understanding (human) memory (13). He also suggested a mechanism how the network can learn new information and forget what was learned before. Since then, researchers have documented a wide variety of emergent complex phenomena in brain networks, such as scale-free avalanches of neural activity and sudden spontaneous transitions in neuromagnetic field patterns (14). Often, such large-scale network patterns in turn feed back into the small-scale individual neuronal interactions which created them, making it difficult to tease out which dynamics take place at which spatiotemporal scale (15, 16). A neural network is one of the canonical systems where small-scale interactions among simple nodes somehow translate into large-scale complex behavior, without any leader node or a signal from the environment.

A second and even earlier demonstration was the 'random genetic net' introduced by Stuart A. Kauffman in 1969 (17). In this early work he constructed random networks of a few hundred 'binary on-off devices' and interpreted the emergent dynamics as a model of the gene regulation process inside a cell. In his computer model, links denote chemical reactions where genes activate or deactivate other genes. Kauffman found that the random

genetic nets behave with remarkable order and stability, which are characteristics of real cells. Another observation was that the system goes through "behavior cycles whose length predicts cell replication time as a function of the number of genes per cell". Since then, the model has been applied to other processes as well (18). Nowadays such networks are called random Boolean networks or Kauffman networks, which are still studied in various forms.

The notion that the nodes are 'simple' in these examples is relative: even though, say, neurons are themselves intricate elements, the complexity of their collective dynamical behavior cannot be anticipated from the dynamics of individual neurons alone. Emergent complex behavior from networks of 'simple', uncoordinated nodes is not only found in brain networks but in a wide variety of other natural phenomena, such as gene regulatory networks (19, 20), socio-technical networks (21, 22), financial trading markets (23), and coupled oscillator networks which model various biological synchronization phenomena (24, 25). Colloquially speaking, "the whole is more than the sum of the parts." (26)

The problem that complex network researchers face today is that the set of dynamical processes that describe the nodes is endless. In brain networks, neurons have ion pumps and channels to create electric potentials within them, and occasionally send ion concentrations through their axon. In protein-protein interaction networks, each protein is a chain of amino acids which non-trivially folds into a lump: the reachable binding sites on the outside define the protein's function. In the ultimate case, techno-social networks consist of humans whose behavior still defies our understanding.

Nowadays, a common approach to model such systems is to replace the node dynamics with a simple, prototypical node dynamics. The implications of network topologies are often modeled using one of a small number of prototypical node dynamics. Prominent prototypes of dynamics to place on the nodes are the following.

- *Percolation of particles.* The classical percolation theory (27) is concerned with the movement of particles through porous media, such as water molecules seeping through cracks in rocks, also called 'soil physics'. The voids in the porous media are somehow linked together, lending itself naturally to a networks description (28). There are two main variants of how percolation is used as a tool to characterize complex networks. In the first variant it is assumed that an abundance of particles move so fast that the only limit is the connectedness of the network, so these studies are concerned with network characteristics such as the size of the big component and the redundancy of paths. This variant is typically used to estimate the robustness of networks (29–32). The second variant assumes a smaller number of slow-moving particles, which become random walkers (Brownian motion particles) over the network topology. One application of this variant is to measure the so-called 'communicability' of networks (33), that is, their ability to support communication. A second application is to find the most influential nodes in a given network topology, termed the 'influence maximization problem' (34, 35). Other applications include the study how information transfers through networks the presence of noise (36) and the community detection problem (37).
- *Spreading phenomena.* The most prominent application of the spreading phenomenon is the study of how infectious diseases spread through a population (38–43), and how to prevent it (44). The basic idea is that nodes with the 'infected' status attempt to infect their direct neighbors in the network, thus spreading the 'infection' like an oil stain. This prototype has been used to describe the adoption of innovation and behavior in social contexts (45–49), the spreading of rumors (50, 51), the dissemination of routing information in communication networks (52–55), the flow of information and topics (45, 56–60) in e.g. blog networks, the spreading of financial distress among banks and funds (61), the

9

resilience of networks to spreading phenomena (62, 63), and the identification of influential nodes (60), among others.

- *Dynamical oscillators.* In 1948, Wiener (64) wondered: "How is it that thousands of neurons or fireflies or crickets can suddenly fall into step with one another, all firing or flashing or chirping at the same time, without any leader or signal from the environment?" Such self-organized synchronization is a widespread phenomenon, including the electrically synchronous pulsation of pacemaker cells; cooperative behavior of insects, animals, and even humans; metabolic processes in our cells; and the synchronous firing activity of neurons in the brain. In order to model the phenomenon, each node is modeled as an innately oscillatory device which is weakly coupled with other nodes in a network (24, 42, 65–70). Depending on the network structure of the interactions, the nodes (oscillators) may synchronize their frequencies or not. Different topological features, such as the network diameter and the betweenness centrality of nodes, lead to a different 'synchronizability' of the network (71, 72).

- *Magnetic spins.* Whereas oscillators are used to describe spontaneous synchronization, spin models are used to describe the spontaneous emergence of order. The inspiration of the model comes from the physics of magnets, which consist of a regular structure (crystal) of spinning charged particles (spins) whose orientation is variable. Each spin induces a small local magnetic field to which neighboring spins tend to align, and vice versa. Magnetization emerges if a majority of the spins are aligned, which is an ordered state. The most popular mathematical spin model is the Ising model, where each node can be in one of only two states: 'up' or 'down'. The spin-spin interactions tend the system towards an ordered state, and a global temperature tends the system towards an unordered state. One of the characteristics of the topology of interactions is the 'critical temperature' (or its non-existence) that it induces, which is the temperature that is needed to move the system from order into

disorder. Not only is the spin model is used to assign dynamical properties to networks (42, 73, 74), it is also used to model diverse phenomena such as social opinion forming (75–77), protein folding (78–81), and the collective behavior of neural networks (11, 12, 82, 83).

The main reason for such reductionism is to make modeling and analysis tractable, rather than their realism. Another reason is that a large body of theory had already been accumulated about each of these prototypes in physics before they were applied to networks. The underlying assumption seems reasonable, namely given a network topology, certain aspects of node dynamics are likely more relevant than others in order to model the system faithfully. Still, a formal framework to select these aspects is missing, not least because it is unknown how a network of small-scale interactions combines into collective behavior.

The task for the coming years seems to be to gather data of node dynamics in real complex networks. After all, it was the sudden availability of large-scale data of the structure of real networks that spurred network theory in the past two decades. As Barabasi conjectures (10), "if data of similar detail capturing the dynamics of processes taking place on networks were to emerge in the coming years, our imagination will be the only limitation to progress."

1.1.3 Information processing

Clearly, a unifying framework is needed that reveals how a (microscopic) change of one node's state influences other nodes to change their state, vice versa, and so on, somehow creating the (macroscopic) behavior of the network. It should prevent the need to characterize dynamics on a case-by-case basis, and it should enable comparing the dynamics of disparate systems. We believe that one possible way to implement such a framework is to describe how a system inherently processes *information*, in the sense of Shannon's information theory (84). In this interpretation, information is inherently stored in the state of each node. This information transfers from

one node to another through the interactions, and becomes stored in the other node's state. Information is also lost due to randomness or thermal noise, where some of the stored information is replaced by random values. Information processing is an inherent process, generated by the dynamics that the network is executing.

This idea existed before, albeit in colloquial phrases and predominantly in the quantum computing literature. In his search for the ultimate quantum computer, Seth Lloyd looks at the nature around us (85): "Every physical system registers information, and just by evolving in time, by doing its thing, it changes that information, transforms that information, or, if you like, processes that information." Slightly more specific, each particle in an isolated gas 'knows' something about the momenta of neighboring particles due to the transfer of momentum during collisions. That is, the momentum of a particle is the result of its recent collisions with other particles. This information is in turn transferred to other particles in subsequent collisions, and so on. This is also true at larger scale. For instance, if these particles are placed in a piston then the piston exerts force on the gas, and at the same time the particles create a pressure that counters the piston's movement. Or as Wiesner (86) phrases it in her attempt to quantify the intrinsic computation of quantum systems, "a quantum finite-state automaton in its most general form takes in classical information, processes it using quantum mechanical resources, and outputs classical information."

This input/output description at the macroscopic scale is the underlying idea of most previous work to quantify the information processing of dynamical systems. If a system of coupled units[1] is influenced by a time-varying input signal, then the system can be thought of as storing information about the past signals and inherently computing nonlinear functions of them (87–89). In this view, the system is usually treated as a black-box which produces a

---

[1] In this dissertation, the terms 'unit' and 'node' are used interchangeably to refer to the connected entities in a network. The word 'unit' is used to imply that it performs some local dynamics, such as a neuron in a brain network; the word 'unit' is used to refer only to a topological entity.

time series of system states (a list of the node states), which are interpreted as the results of the computation that the system performed. A notable operationalization of this idea is the decade-old 'reservoir computing' (90–92), which is a neural networks-inspired approach to exploit this intrinsic computing of dynamical systems to perform useful tasks such as machine learning. In effect, this line of research addresses the question of *what* a dynamical system computes, rather than *how* it is computed.

As a first step to address the *how*, Crutchfield et al. (93–96) take a slightly 'less macroscopic' viewpoint. Instead of processing external signals, they interpret a system to process its own information. Still a black box, the time series of a system's state is used to compute information and complexity measures of the underlying process. The most prominent quantities are entropy rate, statistical complexity, and excess entropy, which characterize the process' inherent randomness, its structural complexity, and how much information is transmitted from previous states to future states. One of the main goals of this line of research is to quantify how difficult it is to learn a process' hidden organization from observations. A related goal is to determine how ordered or chaotic the dynamical behavior of a system is, in a model-independent manner. The complexity measures used here can be considered as a variation on the classical algorithmic (Kolmogorov-Chaitin) complexity (97), where random output is considered 'simple' instead of 'complex'.

As a first step to characterize the information processing *inside* a system, Crutchfield et al. exploit the equivalence between a one-dimensional time series of system states and a one-dimensional array of node states with nearest-neighbor interactions. In doing so they quantified for a one-dimensional lattice of Ising spins the total amount of information that passes through a single node from its left side to the right, and vice versa. Unfortunately, as Wiesner observed (86) in 2010, "the analysis so far is confined to time sequences or, equivalently, one-dimensional spatial sequences." This is exemplified by an attempt to generalize the approach to

a two-dimensional lattice of Ising spins (98), which they show is not trivial and they present three possible methods which are not equivalent.

It is clear that a system's state somehow stores information about external signals and its own past states, part of which is transferred to future states. But how does a system do it? A system consists of dynamical nodes, so the information must be stored in the states of individual nodes. As these nodes update their state, this information is transferred among the nodes both in space and in time.

We are not aware of previous work that takes such a microscopic standpoint in the context of information processing. Yet we believe that it may lead to a unifying framework to translate small-scale interactions to large-scale behavior. It is the primary subject of this dissertation. In the remainder of this section we explain our interpretation of information processing in complex networks. First we review the key concepts from Shannon's information theory; then we calculate how information is stored and partially transferred in the simple example of an array of coin flips.

Information theory
The amount of information that is stored in a variable is the minimum number of yes/no questions that is needed to determine a value for the variable (84). The value of a variable that encodes the result of a fair 50%-50% coin toss can be uniquely identified by at least one yes/no question, namely 'did the coin toss result in heads?' We say that it stores 1 *bit* of information. More generally, a variable with N equally probable values stores $\log_2 N$ bits. This is the maximum amount of information that a variable with N possible values can store.

The information stored in a variable can be less than $\log_2 N$ if its values have different probabilities. Suppose, for instance, that we toss a coin of which we know it results in heads 90% of the time. Intuitively, the outcome of a toss is less informative because we already anticipate it in part. In the extreme case of a coin with two identical sides the outcome of a toss provides zero information because the question of its outcome is already completely answered beforehand.

In general, the number of bits that is stored in a variable s with possible outcomes $\{v_1, v_2, \ldots, v_n\}$ is the Shannon entropy

$$H(s) = -\sum_i p_i \log p_i \,,$$

where p_i is the chance that the value of s is v_i. This quantity has two meanings. Firstly it is the number of bits that one must obtain, through measurement or inference, in order to identify the outcome of s. The second meaning is that s is capable of storing $H(s)$ bits of information about other variables, which we explain next.

Let us interpret variable s_1 to encode the state of one dynamical unit and variable s_2 to encode the state of another dynamical unit. We refer to s_1 and s_2 as states and their values as instances. According to the second meaning of entropy, learning the instance of s_1 can provide between zero and $H(s_1)$ bits of information about the instance of s_2. It is non-zero in case the states s_1 and s_2 are correlated or cause-and-consequence, such that the fact that s_1 is in a particular instance tells us something about the instance of s_2. If s_1 and s_2 are independent processes then this *mutual information* is zero. If, on the other hand, s_2 encodes the state of a tossed coin that tends to be equal to the state of another tossed coin s_1, then the more s_2 depends on s_1 the more information about s_1 is stored in s_2 (and vice versa).

The amount of this mutual information is

$$I(s_1 \mid s_2) = H(s_1) - H(s_1 \mid s_2), \tag{1}$$

where $H(s_1 \mid s_2)$ is the conditional variant of $H(s)$. In words, knowing s_2 reduces the number of unknown bits about the outcome of s_1 from $H(s_1)$ to

$H(s_1 | s_2)$. We can interpret $I(s_1 | s_2)$ as the amount of information that the variable s_2 stores about the variable s_1.

Information can be said to transfer between interacting states. Here, an interaction between two states means that one state (partly) depends on the other state, vice versa, or both. Suppose that two interacting dynamical units s_1 and s_2 form a system, and suppose that both units are equally influenced by other factors outside this system. The information that is stored by the state s_1 about the other state s_2 now consists of two parts: an amount I_{corr} which both states have in common because they are subject to the same external influence (creating a correlation), and an additional I_{int} which is due to the interaction between the states (creating additional correlation) (96). The information I_{int} is present in the variable s_1 at first and would not be in s_2 if there would be no interaction, so we can say that information transfers through interactions from one state to another. Although I_{corr} did not transfer directly through the interaction between s_1 and s_2, it did transfer through the external interactions in a similar way to how I_{int} transfers between s_1 and s_2. The reason that it creates mutual information between s_1 and s_2 is because both states store the same I_{corr} bits of information about external factors, which make the stored information in both states overlap with each other.

Information storage and transfer: an example
Suppose that the outcomes of coin flips $s_1, s_2, s_3, \ldots$ depend on each other such that s_2 tends to be equal to s_1, then s_3 tends to be equal to s_2, and so on. The question is how far the information from s_1 can travel in this one-dimensional system.

If the coin flips do not depend on each other, i.e., each s_i reproduces s_{i-1}

with a 50% chance, then the outcome of a coin flip provides no information about the outcome of any other coin flip. Hence, information about s_1 is not transferred and remains local. See Figure 1a.
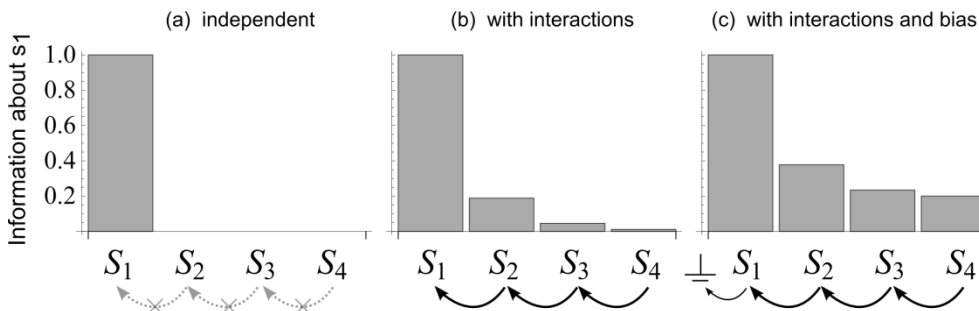
Figure 1: Three different modes of information transfer in a one-dimensional sequence of coin tosses. (a) If a subsequent coin toss does not depend on the previous outcome, then information remains local and is not transferred. (b) If each coin toss tends to be equal to the previous outcome, then information about the first outcome transfers to subsequent coin flips, while its magnitude diminishes to zero. (c) If each coin toss is biased then each outcome provides the transferred information in addition to the constant prior information due to the bias. The dissipation time of information is not affected by the presence of prior information.

Suppose now that each outcome s_i is equal to s_{i-1} with a 75% probability. The second coin flip s_2 can infer the probability distribution of the outcome s_1 by using Bayes' theorem $p(s_{i-1} \mid s_i) \cdot p(s_i) = p(s_i \mid s_{i-1}) \cdot p(s_{i-1})$, which in this example means simply that s_1 is distributed 75%-25% over its two possible states. Using Eq. (1) we find that s_2 stores 0.19 bits about s_1, or in other words, 19% of the state s_2 is actually a reflection of the state s_1. The remainder 81% of its state is still randomness or noise, as before. Similarly, s_3 can use Bayes' theorem to find that its state is equal to s_1 with probability $0.75^2 + 0.25^2 = 68\%$, so according to Eq. (1) it received 0.046 bits of information from s_1. Clearly, the 1 bit of information about s_1 is imperfectly transferred through the system and eventually vanishes. See Figure 1b.

17

In the more general case there can be prior information about s_1 already stored in the system. This information is not transferred through dynamics but can be due to an external force. For instance, let s_1 be the outcome of an unfair coin flip that is distributed 75%-25%. Even in the absence of interactions, each subsequent outcome s_i can already infer 0.19 bits of the state s_1. Information received due to interactions will be additional to this 'baseline' information, see Figure 1c.

Information dissipation length
Clearly, there is a simultaneous transfer and decay of the information that was initially stored in the state s_1. The more information about s_1 becomes stored in another state s_i, the more s_i is influenced by s_1. The distance that this information can travel before it disappears, therefore, is a measure of the extent that s_1 influences the global system state: if it is low then its effects remain local, whereas if it is high it may lead to a system-wide change of state.

We name this distance the *information dissipation length* (IDL). Along with its temporal variant, the *information dissipation time* (IDT), it is a central quantity in this dissertation. The IDT is further explored on networks in Chapter 2; the IDL is explored in real data in Chapter 3.

We can calculate the IDL of the first coin exactly in this example. All coins and their interactions are equivalent, so we expect a constant rate $1/f$ of losing information at each subsequent coin flip:

$$
f = \frac{I(s_1 \mid s_x)}{I(s_1 \mid s_{x+1})} = \frac{H(s_1) - H(s_1 \mid s_x)}{H(s_1) - H(s_1 \mid s_{x+1})}
$$

$$
= \frac{1 + \dfrac{\left(q + q^x\right)}{2q} \log_2\left[\dfrac{q + q^x}{2q}\right] + \left(1 - \dfrac{q + q^x}{2q}\right) \log_2\left[1 - \dfrac{q + q^x}{2q}\right]}{1 + \dfrac{\left(q + q^{1+x}\right)}{2q} \log_2\left[\dfrac{q + q^{1+x}}{2q}\right] + \left(1 - \dfrac{q + q^{1+x}}{2q}\right) \log_2\left[1 - \dfrac{q + q^{1+x}}{2q}\right]} .
$$

Here, the logarithm has base 2 so that the unit of information is bits. The substitution $q \equiv 1 - 2p$ was applied to fit the equation on the page. For the conditional probability function $p(s_1 \mid s_x)$ we used

$$p(s_1 \mid s_x) = p(s_1 = s_x) = \frac{1 + q^{x-1}}{2} \, .$$

That is, for s_x to infer the probabilities of the two states of s_1 it is, in this case, equivalent to calculating the probabilities of s_1 being equal to s_x. The order of the two probabilities does not matter, because Shannon's entropy is symmetric about $1/2$. The probability $p(s_1 = s_x)$ is equal to the probability that, out of the $x - 1$ coin flips, the number of 'failures' (the outcomes of coin flips that are not equal to their predecessor) is an even number. The above relation can be verified by induction, i.e., verifying the relation for $p(s_1 = s_1)$ and then for $p(s_1 = s_x)$ given that it is valid for $p(s_1 = s_{x-1})$.

An initial numerical exploration suggests that f is indeed a constant rate, except for a small deviation for the lowest n. Therefore we calculate the limit of f as $x \to \infty$, as follows. Since $0 < p < 1$, all terms q^x and q^{x+1} go to zero, which makes both the numerator and the denominator go to zero. Applying L'Hôpital's rule instead, $\lim_{x \to \infty} f$ becomes

$$\lim_{x\to\infty} f = \backslash$$

$$\lim_{x\to\infty} \frac{\dfrac{d}{dx}\left(1+\left(\dfrac{1}{2}+\dfrac{1}{2}q^{x-1}\right)\log_2\left[\dfrac{1}{2}+\dfrac{1}{2}q^{x-1}\right]+\left(\dfrac{1}{2}-\dfrac{1}{2}q^{x-1}\right)\log_2\left[\dfrac{1}{2}-\dfrac{1}{2}q^{x-1}\right]\right)}{\dfrac{d}{dx}\left(1+\left(\dfrac{1}{2}+\dfrac{1}{2}q^{x}\right)\log_2\left[\dfrac{1}{2}+\dfrac{1}{2}q^{x}\right]+\left(\dfrac{1}{2}-\dfrac{1}{2}q^{x}\right)\log_2\left[\dfrac{1}{2}-\dfrac{1}{2}q^{x}\right]\right)}$$

$$= \lim_{x\to\infty} \frac{\dfrac{1}{2}q^{-1+x}\left(\log_2\left[\dfrac{1}{2}-\dfrac{q^{-1+x}}{2}\right]-\log_2\left[\dfrac{q+q^x}{2q}\right]\right)\log q}{\dfrac{1}{2}q^{x}\left(\log_2\left[1-q^x\right]-\log_2\left[1+q^x\right]\right)\log q}$$

$$= \lim_{x\to\infty} \frac{\left(\log_2\left[\dfrac{1}{2}-\dfrac{q^{-1+x}}{2}\right]-\log_2\left[\dfrac{q+q^x}{2q}\right]\right)}{q\left(\log_2\left[1-q^x\right]-\log_2\left[1+q^x\right]\right)}.$$

Taking this limit would lead to $0/0$ again, so we apply L'Hôpital's rule for the second time:

$$\lim_{x\to\infty} f = \lim_{x\to\infty} \frac{-\dfrac{q^{x-1}\log q}{1-q^{-1+x}}-\dfrac{q^{x-1}\log q}{1+q^{-1+x}}}{-\dfrac{q^{1+x}\log q}{1-q^x}-\dfrac{q^{1+x}\log q}{1+q^x}}$$

$$= \lim_{x\to\infty} \frac{-\dfrac{2q^{x-1}\log q}{1-q^{2x-2}}}{\dfrac{2q^{x+1}\log q}{-1+q^{2x}}}$$

$$= \lim_{x\to\infty} \frac{q^{2x}-1}{q^{2x}-q^2}.$$

As the last step we use the constraint $-1 < q < 1$ and substitute q back to arrive at the expression

$$\lim_{x \to \infty} f = \frac{1}{(2p-1)^2} \equiv f \,.$$

For the example of $p = 0.75$ used in Figure 1, this evaluates to 4.

In words, each subsequent coin flip s_i stores one quarter of the information that its predecessor s_{i-1} stores. We can define the information dissipation length as the characteristic halftime of information, so that

$$\text{IDL}_{\text{coins}} = \log_{1/f} \frac{1}{2} = \frac{-1}{\log_2 (1-2p)^2}$$
$$\overset{p=\frac{3}{4}}{=} \frac{1}{2}. \tag{2}$$

Finally we illustrate how the IDL of the system of coins depends on the 'copy' probability p in Figure 2. Note that replacing x by a temporal coordinate, e.g., letting a single coin depend on its own previous outcome, would lead to an analogous expression for IDT.
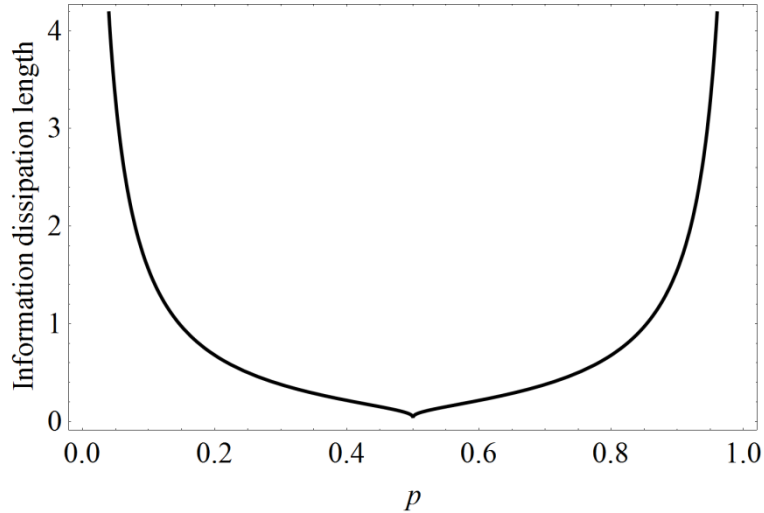
Figure 2: The information dissipation length (IDL) of a sequence of conditional coin flips, where each coin flip has the same outcome as the previous coin flip with probability p**. As** $p \to 0$ **or** $p \to 1$ **the IDL diverges to infinity; as** $p \to 1/2$ **the IDL goes to zero, as expected.**

Information dissipation in networks

In a general system of interacting units we can let s_i^t correspond to the state of unit i at time t. The more random is the interaction among connected units, the less information is transmitted between the states and therefore the lower the correlation between the states of the units. These connections form a network in which information about a unit s is transmitted to its neighbors, which store it in their state. Each neighbor subsequently transmits information about their own state, which partly consists of the information from unit s, and so on, inducing the percolation and mixing of many different pieces of information through the system.

The concept of information dissipation is a first characterization of how information is processed among the nodes in a network. It is the primary conceptual contribution of this dissertation.

1.2   Our Thesis

Our Thesis is that the ability to quantify the inherent information processing in networks of dynamical nodes leads to a unifying framework for a better understanding of the behavior of complex adaptive systems. This Thesis consists of two parts:

1. The information processing concept provides a set of analytical tools to translate the microscopic node dynamics into the macroscopic network's behavior.
2. The information processing concept provides a single language to characterize the dynamical behavior of disparate systems.

1.3   Outline of the Dissertation

In Chapter 2 we formulate the time it takes for information about a single unit to dissipate through the network. We name this quantity the *information dissipation time* of the node and relate it to the dynamical importance of the node, i.e., the impact that the node has on the dynamical behavior of the network as a whole. This quantity is derived as a function of a node's connectivity (degree) for large networks with any degree distribution. We find counter intuitively that the information dissipation time (i.e., dynamical importance) diminishes for highly connected nodes. We validate this finding in computer simulations of networks of Ising spins with a heavy-tailed degree distribution. The finding is consistent with empirical evidence from the literature of a social viral marketing network, a network of human proteins, and neuronal activity in *in vitro* neural networks. Each source of empirical evidence left the phenomenon unexplained. We are able to provide an alternative explanation for all three cases based on the dissipation of information of individual nodes.

In Chapter 3 we formulate the spatial variant of the measure used in the previous Chapter, namely the *information dissipation length* of a node. We relate this quantity to the capability of the network as a whole to undergo a self-organized systemic change, i.e., the event where the majority of nodes change their state in a synchronous manner. We estimate this measure using

real data of daily prices of interest-rate swaps in the USD and EUR financial derivatives markets and demonstrate that it could have provided an early warning signal to the bankruptcy of Lehmann Brothers. Their bankruptcy was a pivotal financial event that marks the beginning of the recent banking crisis. We also demonstrate that previously introduced leading indicators would not have provided such a clear advance warning, as well as widely used financial indices such as the onset of a so-called swap basis and changes in the spread levels. These results show that the information dissipation time measures a characteristic of the financial market that is not adequately captured by other leading indicators. In addition it substantiates the concept that the dissipation of information can be used to characterize the network's collective behavior, in this case its capability to self-organize towards a systemic crash.

The previous two Chapters were concerned with translating the local dynamics into the global dynamics. In Chapter 3 we study how the global dynamics of the HIV epidemic contain information about the local dynamics of men-who-have-sex-with-men (MSM), which constitute the majority of the HIV spreading network. The epidemiological process of HIV among individual MSM, each with his own immune system and promiscuity, shapes the phylogeny of the virus population. This phylogeny is the (anonymized) set of genetic sequences of patients available in hospital databases, combined with a measure of similarity of each pair of sequences. Phylogenetic data is inherently ambiguous and incomplete. Nevertheless we show in this Chapter that phylogenetic data, consisting of 14560 subtype-B sequences from 2001 through 2007 from the UK HIV Drug Resistance Database, indeed contains information about key epidemiological parameters, both at the population-scale and at the individual scale. We use computer simulations of the HIV epidemic among individual MSM where infections occur over a dynamic network with a heavy-tailed degree distribution. All parameter values and their confidence intervals are taken from literature.