Matthias Wölbitsch, BSc

Introducing Peer Influence Mechanisms into Activity-driven Network Generators

Master's Thesis

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Computer Science

submitted to

Graz University of Technology

Supervisor

Assoc.Prof. Dipl.-Ing. Dr.techn. Denis Helic

Institute of Interactive Systems and Data Science
Head: Univ.-Prof. Dr. Stefanie Lindstaedt

Graz, April 2017

This document is set in Palatino, compiled with pdfLaTeX2e and `Biber`.

The LaTeX template from Karl Voit is based on KOMA script and can be found online: https://github.com/novoid/LaTeX-KOMA-template

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date	Signature

Abstract

Time-varying networks open up the possibility to explore dynamical systems with respect to their temporal dimension in a more precise way. One class of these systems that is particularly interesting are networks of human interactions. They are characterized by the complex bursty patterns of human behavior. The recent introduction of the activity-driven time-varying network framework led to an increased effort to model such social systems more accurately. However, all of these approaches rely solely on intrinsic activity patterns of individuals and disregard possible external influences entirely. In this thesis we propose an activity-driven network model by introducing a peer influence mechanism into the network dynamics. Thus, we allow for individuals to motivate their neighbors in the social network to become active as well. We examine the ramifications of this mechanism on the topological and activity-related properties of synthetically generated networks and reveal its complex influence on the dynamics. Furthermore, our results indicate a positive effect on the emerging activity patterns.

Keywords: activity-driven model, time-varying networks, peer influence, user activity, network science

Contents

Abstract iv

1. Introduction 1
- 1.1. Background . . . 1
- 1.2. The Web as a Living Lab . . . 2
- 1.3. Temporal Data . . . 3
- 1.4. Motivation . . . 4
- 1.5. Thesis Outline . . . 5

2. Related Work 7
- 2.1. Graph Theory Basics . . . 7
- 2.2. Social Networks . . . 11
- 2.3. Time-varying Networks . . . 15
- 2.4. Generative Network Models . . . 18
  - 2.4.1. The Erdős-Rényi Model . . . 18
  - 2.4.2. The Barabási-Albert Model . . . 21
  - 2.4.3. Watts-Strogatz Model . . . 23
- 2.5. User Activity Models . . . 24
  - 2.5.1. Stochastic Models . . . 25
  - 2.5.2. Queuing Models . . . 27
  - 2.5.3. Time-varying Network Models . . . 27
- 2.6. Peer Influence . . . 33

3. Model 38
- 3.1. The Underlying Model . . . 38
  - 3.1.1. Description . . . 38
  - 3.1.2. Properties . . . 44

3.2. Peer Influence Extension 45
3.2.1. Idea and Incentive 45
3.2.2. Description 48

4. Results **55**
4.1. Time-dependent Integrated Network Properties 56
4.2. Network Activity 62
4.3. Inter-event Time Distributions 65
4.4. Degree and Tie Strength Distributions 69
4.5. Tie Strength Rescaling Scenarios 72

5. Conclusion **76**
5.1. Recap 76
5.2. Limitations 78
5.3. Future Work 79

A. Power-law Probability Distribution **82**
A.1. Probability Density Function 82
A.2. Cumulative Distribution Function 83
A.3. Expected Value 84
A.4. Example: $\gamma = 2.5$ and $\varepsilon = 10^{-2}$ 84
A.5. Inverse Transform Sampling 86

Bibliography **88**

List of Figures

2.1. Examples for graphs 8
2.2. Clustering coefficient examples 11
2.3. Zachary's karate club network 12
2.4. Graphical representations for time-varying networks 19
2.5. Watts-Strogatz model example 24
2.6. Integrated network example 31

3.1. Egocentric network example 40
3.2. Probability distribution for the formation of new ties 41
3.3. Cyclic closure mechanism example 43
3.4. Influence of p_Δ and δ on the community structures 46
3.5. Extended egocentric network example 50
3.6. Peer influence sigmoid function examples 52

4.1. Average local clustering coefficient as function of time 57
4.2. Segments in the average local clustering coefficient evolution 59
4.3. Percentage of reinforced ties as function of time 60
4.4. Average degree and tie strength as function of time 61
4.5. Time-dependent number of activations 63
4.6. Percentage of peer influenced activity and cumulative activity as function of time 64
4.7. Log-log plot of the inter-event time distribution 68
4.8. Bulk of the inter-event time distribution 68
4.9. Degree distribution 71
4.10. Tie strength distribution 71
4.11. Network activity in different rescaling scenarios 74

A.1. Example probability density function of a power-law distribution. 85

A.2. Example cumulative distribution function of a power-law distribution. 85
A.3. Inverse transform sampling example. 87

List of Tables

3.1. Overview proposed model parameter 54

4.1. max and arg max of $C(t)$ 58
4.2. Burstiness of inter-event time distributions 67

1. Introduction

1.1. Background

Plenty of problems that arise in the real world can be solved by first abstracting them in a more general form and solving them by using already established and well-known tools afterwards.

One famous example for this process is attributed to the Swiss mathematician Leonhard Euler. The city Königsberg (now called Kaliningrad) was split into four parts in the 18^{th} century, due to the pathway of the river Pregel. These four areas of the city were connected by seven bridges. The residents of Königsberg had an ongoing challenge to find a way through the city that crosses each bridge exactly once and ends once more at the starting point of the tour. This is known as the Königsberg bridge problem [Pao11; Coo12]. Euler tackled this challenge by eliminating irrelevant details (e.g., the length of the brides, or the size of the areas) and abstracting the problem to its essence. The regions of the city became points and the bridges between two areas became line segments that connect the corresponding points. This abstracted topological view on the problem allowed Euler to solve the Königsberg problem and show that, in fact, no such tour through the city (also known as an Eulerian cycle) exists. For such a tour to be possible requires the number of bridges that are accessible in each region to be even, which is not the case for the Königsberg problem, where the number is odd in each region.

This particular way of abstracting real world objects and their connections into points and lines and applying mathematical methods and reasoning

to it laid the foundation for the mathematical field of graph theory. Results developed in this area are applied today to countless problems, from finding efficient ways to manufacture circuit boards [Coo12], to calculating fast routes for the data packets that are sent over the internet [WW99]. Graph theory also spawned the field of networks science [New10], which deals with large real-world systems that can be modeled as graphs.

For instance, large infrastructure networks, such as power grids can be well described by graphs. They relate parts of the infrastructure, like power stations or transformers, that are connected to each other via power lines [WS98].

Another field that benefits from graph theory as well is sociology, and in particular social network analysis, which investigates the complex behavior of people and groups in the context of social interactions [New10]. Sociological studies are often performed with a small number of participants, since the collection of the required data is time consuming and usually done manually using methods like questionnaires, interviews, or by simply observing the people that are part of the study [WF94].

1.2. The Web as a Living Lab

However, ever since the emergence of the web, people are able to interact with each other more easily. Websites like online social networks (e.g., Facebook, Twitter, Reddit,...), or collaboration forums (e.g., StackOverflow) seem to be very popular. Most of these websites are ranked on top positions on Alexas' list of top 500 visited websites globally [Inc17]. Wellman et al. [Wel+01] showed that these websites (and the internet in general) does not increase or decrease the social capital of people (i.e., their relationships with friends and family, or their commitment to participate in organizations), but supplements it by providing easier ways to organize and plan real-world activities in an online setting.

This availability of large amounts of data that is generated on these websites

can also be beneficial in the context of social network analysis. For example, StackExchange, a website that maintains a variety of communities in which users can ask and answer questions on different topics (e.g., StackOverflow for programming related questions), provides an easy access[1] to their data (e.g., questions, answers, users,...). This data is used in many studies on a wide range of topics (e.g., [Dan+13; Wal+16; Has+16]). A large mobile phone calls data set is used in a variety of papers [Onn+07; KPV14; Mur+15; LSK15] as well. It was collected by an European mobile phone provider with approximately 20% market share and consists of over 630 million logged phone calls between more than six million people. Another interesting data set was used in the study by Sekara, Stopczynski, and Lehmann [SSL16], in which they propose a framework to describe gatherings of people and their (temporal) properties. The data set consists of data from various sources for 1,000 people that was collected over a period of 36 months with a high temporal resolution (i.e., in five minute intervals). The data set contains information about the phone calls, text messages, social media activity, geolocation, and the proximity to other study participants for every subject.

1.3. Temporal Data

All of these large data sets share a crucial feature: they all include temporal information. Every post, tweet, phone call, or text message has a timestamp attached to it. This allows to pin these activities to users at specific points in time and to infer a chronological order between them.

However, not all studies include this additional information into their work. The reasons for this can vary. For instance, some use-cases only require quantitative information (i.e., how often something happens between two objects) and not exactly when it did (e.g., [Kum+07; Bag08]). This corresponds to the elimination of irrelevant details in the abstraction of a problem.

[1] `https://data.stackexchange.com/`

Another reason is that graphs by them self are not able to include the temporal information, since they only represent static relationships between objects. However, there exists an extension that provides the possibility to integrate the time information when required. This type of graphs are called temporal or time-varying networks [HS12; Hol15] and extent graphs in a way that relationships between objects become time dependent. This basically means that the connection between objects is only present at certain points in time, which leads to a more realistic, but also more complex abstraction.

An area that definitively requires the incorporation of time information is the modeling of human behavioral patterns. It has been shown that activities performed by people (e.g., the writing of e-mails, text messages, tweets,...) are not randomly distributed in time, but follow certain patterns instead [Bar05]. Human activity can typically be described as bursty. For instance, it is evident that people tend to write multiple e-mails in a relatively short period of time. This high activity phase is then generally followed by longer periods of inactivity, in which, for example, no e-mails are written at all.

The best way to describe these patterns is by using a probability distribution for the intervals between two consecutive activities (i.e., the inter-event time distribution). The distribution is characterized by its high probabilities for short inter-event times and its long tail that allows for the longer phases with no activity. Most of the time a power-law distribution of the form $p(\tau) \sim \tau^{-\gamma}$ is used.

1.4. Motivation

Inter-event time distributions are also relevant in the context of time-varying networks. For instance, Lambiotte, Tabourier, and Delvenne [LTD13] study the effects of the inter-event time distribution of link activations on dynamic spreading processes in temporal networks.

The framework proposed by Perra et al. [Per+12a] shifts the focus from the

activation of the connections to the activations of the objects. It is based on the simple idea that each entity in the network can become active based on an inherent activity potential and subsequently connects to others in each time step. Therefore, it can, for example, be used to model the activity of users in a social network. However, this model has some disadvantages, due to its simplicity and unrealistic assumptions. For example, it is not able to reproduce inter-event time distributions of the activations and topological properties in the network that can be observed in many real-world networks. Nevertheless, there are applications for this basic framework in many different fields (e.g., [RFP14; RPP16]), and various extensions to it, which are addressing the already mentioned issues (e.g., [LSK15; MSP15; MSP16]).

This simple framework is the foundation on which this thesis is build as well. All models that originated from the original paper by Perra et al. [Per+12a] so far are based on the idea that entities can only become active due to their intrinsic activity potential. However, in the real world people are often heavily influenced by their peers and friends [Wal+16]. They are not doing things solely because of their own determination to do so, but also because their friends are doing it and the with it associated peer pressure. Therefore, in the context of social and collaboration networks is it feasible that more active users have a positive influence on their peers and possibly motivate them to become active as well.

The goal of our work is it to define a model based on the activity-driven time-varying network framework, which is able to include peer influence effects in the activation process of entities, and to study the implications of this mechanism on the network in general.

1.5. Thesis Outline

The content of this document is structured as follows. Chapter 2 starts with basic graph-theoretic definitions (section 2.1) and a detailed introduction into the topics of social and temporal networks (section 2.2 and section 2.3, respectively). Additionally, an overview of important generative network

models and their properties is given in section 2.4. Section 2.5 discusses related work in the context of user behavioral models. It provides a possible explanation for the origin of bursty human behavior with the queuing model, and provides an overview of activity models that are based on time-varying networks, with focus on work based on the activity-driven temporal network framework. The last section (section 2.6) of this chapter is related to peer influence, its ramifications, and applications in different fields.

In chapter 3 the proposed time-varying peer influence network model is discussed in great detail. First, the model and the ideas on which it is based are outlined in section 3.1. The extension that incorporates peer influence effects into the underlying model and the motivation behind it is described in section 3.2.

Chapter 4 contains the evaluation and analysis of the proposed model on synthetic networks. This includes, for example, the examination of ramifications of the peer influence model on topological properties. Furthermore, the effects on the overall activity in general, and with respect to different peer influence scenarios, is discussed as well in this chapter.

The last chapter of this document (chapter 5) concludes the work by summarizing the results and discussing possible limitations. Furthermore, different applications and possible extensions for the peer influence model are presented.

2. Related Work

2.1. Graph Theory Basics

In this section some graph theory related notation is defined, which is used during this thesis. It is for the most part based on [TS92] and [Die12].

A graph is a mathematical construct that can be used to model and explore the relationship between objects. More formally, a graph is an ordered pair of finite sets $G = (V, E)$, whereas V denotes the set of *vertices* (i.e., the objects) and $E \subseteq [V]^2$ the set of *edges* (i.e., the relationships between the objects). It is common to write $V(G)$ and $E(G)$ to refer to the set of vertices and the set of edges respectively, that are associated with a graph G.

An edge $\{v_1, v_2\} \in E(G)$ is an unordered pair of two vertices. Therefore, there is no distinction between the two edges $\{v_1, v_2\}$ and $\{v_2, v_1\}$. A graph with this property is called *undirected*. However, it is also possible to define edges as ordered pairs, such that each edge has a defined start and endpoint. Such a graph is called *directed*.

An edge of the form $\{v_i, v_i\} \in E(G)$ is called a *self-loop* of the vertex v_i. Furthermore, it is possible that two distinct vertices are joined by multiple edges. Such edges are referred to as *parallel* edges. A graph that has no parallel edges and no self-loops is called *simple*. Figure 2.1 depicts an example for a simple graph and for a graph with parallel edges and a self-loop vertex.

It is also possible to assign each edge in a graph a real number, that adds

(a) An undirected simple graph.

(b) An undirected graph with a self-loop and parallel edges.

Figure 2.1.: Graphical representation of two graphs with different properties. The vertices are represented by red dots and the edges are the line segments connecting them.

additional information to it [Cor+09]. This number is usually called *weight* and is determined by the function $w : E(G) \to \mathbb{R}$. This type of graph is called *weighted* graph. However, all further mentions and definitions for graphs are referring to undirected simple graphs, unless stated otherwise.

It is possible to perform operations on graphs. For instance, the union of two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ results in a graph $G = (V_1 \cup V_2, E_1 \cup E_2)$. Other binary operations, such as the intersection of two graphs, can be done analogously. Unary operations on graphs are possible as well (e.g., the removal of vertices or edges).

The *order* of a graph is the number of its vertices (i.e., the cardinality of the vertex set) and is denoted as $n = |V(G)|$, whereas the cardinality of the edge set is usually denoted as $m = |E(G)|$. The neighborhood of a vertex v_i is defined as $N(v_i) = \{v_j \in V(G) : \{v_i, v_j\} \in E(G)\}$. It is the set of vertices that are adjacent to the vertex v_i. The cardinality of this set is called the *degree* of the vertex and is denoted as $d(v_i) = |N(v_i)|$. A vertex without any neighbors (i.e., with a degree of zero) is called isolated.

It is often very useful to measure degree properties for a graph. For example, the minimum degree $\delta(G) = \min\{d(v_i) : v_i \in V(G)\}$, the maximal degree $\Delta(G) = \max\{d(v_i) : v_i \in V(G)\}$, or the average degree

$d(G) = \frac{1}{n} \sum_{v_i \in V(G)} d(v_i)$. An alternative way for the calculation of the average degree is $d(G) = \frac{2m}{n}$. This works since each edge is counted twice in the summation of the vertex degrees in the original formula. These global measures can be used to gain insight into a graph's basic structure.

Another global property related to the degree is the degree distribution p [Bar16] of a graph. It yields the probability that a randomly selected vertex has a degree of k. Since it is a probability distribution $\sum_{k=0}^{\infty} p(k) = 1$ must hold. The degree distribution for a given graph can be calculated by $p(k) = \frac{|\{v \in V(G) : d(v) = k\}|}{n}$, for all possible values for k (i.e., by calculating the normalized degree histogram).

A *path* on a graph can be defined as a finite sequence of vertices $v_1, v_2, \ldots, v_k$, such that between any consecutive pair of vertices exists an edge. Furthermore, all edges between the vertices and the vertices itself must be distinct. The first and the last vertices in the sequence are called the end vertices or terminal vertices of the path.

The *path length* is the number of edges on the path. Two vertices are connected if it is possible to find a path with these two vertices as end points. A vertex is, by definition, connected to itself. If there exists a path between all pairs of vertices, then the graph is called connected. It is always possible to partition the vertex and edge set of a disconnected graph in such a way, that there are no edges between vertices in different partitions. These partitions are called the *components* of the graph.

The *clustering coefficient* of a vertex is a measure for the cliquishness of its neighborhood, and was introduced by Watts and Strogatz [WS98]. A *clique* in a graph is a subset of vertices, such that there exists an edge between every pair of vertices in this set. The clustering coefficient $C(v_i)$ is defined as the fraction of all possible edges between the neighbors of the vertex v_i. There exists at most $\binom{d(v_i)}{2} = \frac{d(v_i)(d(v_i)-1)}{2}$ edges between the vertices in the neighborhood. Therefore, the clustering coefficient can be calculated by

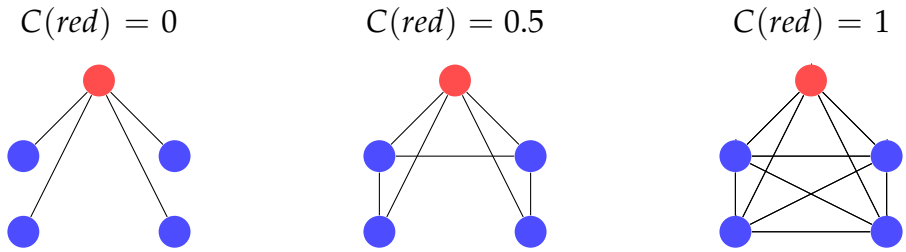
$$C(v_i) = \frac{2\,|\{\{v_j, v_k\} \in E(G) : v_j \in N(v_i) \wedge v_k \in N(v_i)\}|}{d(v_i)(d(v_i)-1)}. \tag{2.1}$$

This is, of course, a local property of one vertex and is therefore sometimes called *local clustering coefficient*. Figure 2.2 shows some examples for neighborhoods with local different clustering coefficients.

It is also often useful to consider the average local clustering coefficient $C = \frac{1}{n}\sum_{v(i)\in V(G)} C(v_i)$ of a graph. However, there is another definition of a *global clustering coefficient*, which is often called *transitivity* [Boc+06]. It is defined as ratio of triangles (i.e., cliques consisting of exactly three vertices) to the number connected triples in the graph, i.e.,

$$T = \frac{3 \times \text{\# of triangles}}{\text{\# of connected triples of vertices}}. \tag{2.2}$$

A connected triple is made up of three vertices as well, but does only have two edges between them. Hence, a triangle consists of exactly three triples. The global clustering coefficient is a measure for the extent of transitive connections in a graph (i.e., if there is an edge between vertices A and B, and between B and C, how likely is it that there is also an edge between A and C). The two global clustering measures are, however, not equivalent to each other and may yield very different values [New10] for the same graph. The definition of the local clustering clustering coefficient introduces a bias towards vertices with smaller degree, due to smaller values in the denominator. Therefore, the value for the average local clustering coefficient is possibly larger than the global clustering coefficient for the same graph, if the number of low-degree vertices is sufficiently large.



(a) In this example none of the four neighbors shares an edge with any other neighbor of the red vertex. Therefore, the clustering coefficient of the red vertex is $\frac{0}{6} = 0$.

(b) In this case, half of the possible edges between the neighbors are present. The clustering coefficient of the red vertex is $\frac{3}{6} = \frac{1}{2}$.

(c) The neighbors of the red vertex form a clique. Hence, the clustering coefficient of the red vertex is $\frac{6}{6} = 1$.

Figure 2.2.: Examples for the local clustering coefficient of a vertex with a small neighborhood. The blue vertices are the neighbors of the red vertex. The possible number of edges between the four neighbors is $\binom{4}{2} = 6$.

2.2. Social Networks

It is a common practice to use graphs to model the complex systems that arise in the real world. For example, the web can be represented as a graph, where vertices correspond to websites and edges to the hyperlinks that connect them. However, when modeling these large networks, it is common to use a terminology that is slightly different than the one used in the mathematical field of graph theory [Bar16]. In the context of networks vertices and edges are usually called *nodes* and *links* respectively.

Social networks [New10] are another type of network that can benefit from the usage of graph theory methods. The study of these networks is considered to be a part of the field of sociology and researchers may also use slightly different terminology for the vertices and edges in their work. Nodes (or vertices) often represent people in social networks and are sometimes referred to as *actors*. However, it is also possible that they depict other entities, such as departments, companies, or countries (i.e.,

larger groups of people). The links (or edges) between these entities can denote, depending of the context, different things as well, and are sometimes referred to as *ties*. For example, links between persons can show social relationships (e.g., friendships), collaborations in projects (e.g., co-authorship of scientific papers), or other social interactions. Links between companies could represent trading relationships or the like.

Social networks are often only mentioned in relation to large online communities, such as Facebook or Twitter, but there is no necessity that a social network must exist in an online form. For instance, the network of acquaintances or friends in a school is considered a social network as well. Figure 2.3 shows a famous example of a small real-world social network.

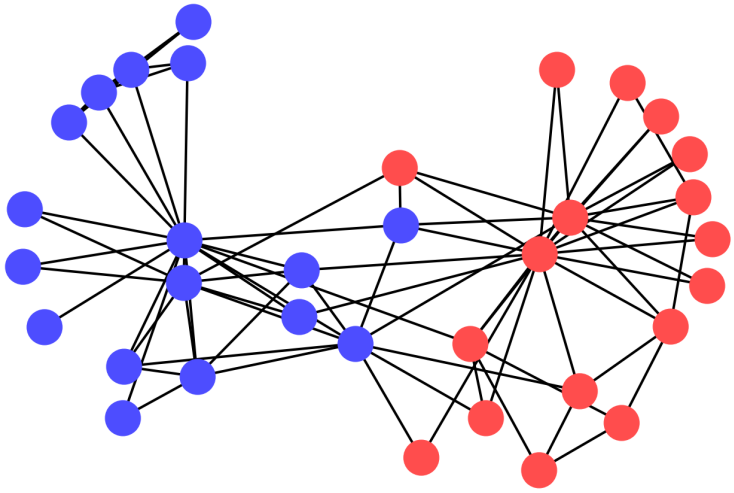


Figure 2.3.: Zachary's karate club network [Zac77] is a social network that shows the relationship between 34 members of an university-based karate club in the US in the early 1970's. There exists an edge between two members if there were social interactions outside of the normal club activities between them (i.e., two members are considered friends). The graph shows a separation into two communities (red and blue nodes). This was caused by a dispute between the members of the club that was focused around two key persons.

Two fundamental terms in the field of social network analysis are *dyads* and *triads* [WF94]. These two concepts describe the relationship between

multiple actors. Dyads denote the linkage between pairs of actors and are used to study pairwise relationships in social networks. A triad on the other hand describes triples of actors and the ties between them. This concept is especially important for the question of transitivity of certain relationships. For instance, "is the friend of my friend also my friend?". The transitivity of trust, especially in the context of cryptography, is an interesting issue as well [CH97].

Another important concept is the strength of ties, which was introduced by Granovetter [Gra73]. The strength of a tie is influenced by factors like the time that two actors spend together, or the intimacy between them. Depending on the strength, a tie can either be strong, weak, or absent. Absent does not only include non-existing ties, but also ties with a strength that is below a certain threshold. This concept of strong and weak ties can be used to explain the formation of triads. If there already exists a strong tie between the two actors A and B, and between B and C, then there exists at least a weak tie between A and C. This is true, due to the opportunities for the formation of a tie between A and C, that will result from a common strong tie to the actor B. Furthermore, the spreading of information in social networks can be explained more accurately by taking the strength of ties into account. Weak ties can help the flow of information by acting as bridges between sparsely connected parts of the network.

A property that many real-world social networks share are *community structures* [GN02]. Communities are groups of actors that are more connected to actors in the same group, than to actors in different ones. This basically means that in networks with community structures exist subsets of densely linked nodes and very few link links between the subsets. The network shown in figure 2.3 has two known communities highlighted by different node colors in the graph.

The detection of community structures in networks is an important research topic, since the identified communities may correspond to actual social groupings. For example, the detected communities in a social network that models the friendship between students may represent the real corresponding social groups (i.e., the circles of friends). There is a variety of different methods and approaches to perform this task. Examples are algorithms

based on hierarchical clustering, or edge betweenness (i.e., the number of shortest paths going through an edge) [For10].

However, it is often not feasible to study the network and its structure on a full global scale, due to processing limitations. One alternative way is to consider single actors and their immediate neighborhood instead. These small sub-networks are called *egocentric* or *personal* networks [New10]. Usually some number of these egocentric networks is sampled from the entire network and is examined for their local properties, such as the degree distribution or the local clustering coefficients.

Another attribute that many social networks share is that they are *scale-free networks* [Bar16]. Formally, a network is a scale-free network if its degree distribution is a power-law (i.e., $p(k) \sim k^{-\gamma}$, where γ is the parameter of the distribution that denotes the degree exponent). The value for γ for most real-world (social) networks is in the range between 2 and 3. A consequence of the power-law is that the distribution of the degrees is right-skewed and long-tailed. This means that there is a large number of nodes in the network that have only a few links (i.e., a small degree) but there is also the chance that there exists a few nodes with a very high degree. Such nodes are usually called *hubs* and may correspond in the context of social networks to very influential actors that can play an important role in the network.

A property that social networks often show as well, is the presence of the small-world effect. The characteristic of small-world networks [Wat99] is that even if the network is very large and sparsely connected, the average shortest path length is usually very small. For example, the famous experiment by Travers and Milgram [TM69] showed that it is possible to send a letter to a target by a chain of about six people on average, where each person in the chain is only allowed to forward the letter to acquaintances. However, even tough there are some issues with the experimental setup of this study (e.g., a sample selection biases, or the sample size [Sch09]), there is additional work in this area that supports the original findings. For example, Leskovec and Horvitz [LH08] study of an online social network with 180 million nodes and 1.3 billion edges showed that the average path length in this network is 6.6, which is only slightly larger than in the original work. Nonetheless, they showed that there exits paths up to a length of 29 as

well. A network with the small-world characteristic does usually exhibit two important properties. First, the average shortest path length is small and only grows logarithmically with the size of the network, and second, it is highly clustered. Note that the small-world effect does not only occur in social networks but also in many other real-world networks as well. Examples are the US power grid network or the collaboration network of actors in movies [WS98].

2.3. Time-varying Networks

This section contains an overview on the concept of time-varying networks [HS12; Hol15]. Since this type of network is used in many different scientific fields it also has a variety of names. For example, temporal networks, dynamic networks, evolving graphs, or the name that is mainly used in this thesis, time-varying networks.

As already mentioned in the section about social networks, the structure, or topology, of networks can be used to understand dynamic processes and their underlying behavior. However, there are many dynamical processes that are modeled using static networks in which links are not active all the time. One example would be a communication network, such as the network of phone calls between people. Another example would be a social or collaboration network, where actors do not interact constantly but in irregular intervals. These link activations at certain times can be very important to explain the dynamic process and are simply lost when approximated by a static graph.

The idea of time-varying networks is to introduce another dimension (i.e., time) to the network and move the information of when something happens from the dynamic process to the network itself. As a general rule, a time-varying network is applicable when the structure (i.e., the topology) of the system and the temporal process are connected to each other. This means that the time scale on which the network itself evolves should be similar to the time scale of the dynamic process that takes place on it. For example,

a time-varying network is not a suitable model for the internet, since the infrastructure (i.e., the topology) changes very slowly in comparison to the transmission of the packages that are routed through the network (i.e., the dynamic process).

The underlying concept of time-varying networks is called *contacts* and can be seen as interactions between two nodes at a certain time. The duration of the interaction is negligible and thus assumed to be instantaneous. This is true at least in context of this thesis, however, it is also possible to include the duration of interactions if necessary. Contacts can be interpreted as an extension of links in static networks. The unordered pair $\{v_i, v_j\}$ becomes an ordered triple (v_i, v_j, t). The order is in this case important, since the third object in the triple must refer to the time of interaction t between the nodes v_i and v_j.

The usage of static networks in models for dynamic processes, which represent some time-dependent sequences of contacts between pairs of nodes, often results in a loss of information. This sacrificed accurateness can possibly be avoided by using time-varying networks instead. However, these temporal networks also introduce more complexity to the model, and one have to weight the gain in information versus the additional modeling effort. For example, the case in which the information on how often something between two actors happened is way more important than when exactly it did, is a strong indicator for the preferred application of weighted graphs over their temporal counterpart (see figure 2.4a for an illustration of a weighted graph).

However, a simple graph without weights can also be used to approximate the interaction sequence of a time-varying network [Hol13]. The idea is to calculate a total weight for each pair of nodes in the network. If the weight exceeds a certain threshold $\Omega \in \mathbb{R}_0^+$ then there will be an edge between the two nodes in the static approximation. Each contact between the pair in the sequence of contacts C contributes to the total weight. However, contributions to the total decay exponentially with time. Hence, the total weight $w_{i,j}$ between two nodes v_i and v_j is $w_{i,j} = \sum_{(v_i,v_j,t)\in C} \exp(-t/\tau)$, where τ is an additional parameter that controls the exponential decay. For

a static approximation, which should contain an edge if there was at least one contact between the two nodes, the threshold can be set to $\Omega = 0$. Networks that are generated using this approach are called exponential-threshold networks.

There are many different possibilities to represent a temporal network without the loss of information as well. The simplest way to to this is by using the actual contact sequences. A contact sequence is basically a list that contains all the contact triples. This is a very raw form data (i.e., in essence a spreadsheet with three columns) and is therefore very easy to parse, and to use in algorithms. However, the sequence is not very well suited for the analysis of the underlying dynamic process by humans, due to the lack of illustrations.

Another way to represent time-varying networks are graph sequences. The idea behind them is to generate a static graph that contains all contacts between nodes in a given time window. This method has the advantage that all tools that work for static graphs can also be applied to each of the graphs in the sequence. The problem with this representation is that the time resolution should be selected carefully to avoid the creation of graphs with no, or only a few, edges for most time steps. Figure 2.4b depicts an example of a graph sequence for a time-varying network with four nodes.

There exists more visual-focused representation methods as well. One example would be the assignment of the series of contact timestamps between two nodes to the corresponding link in the static network (see figure 2.4c for an example). This allows the usage of the variety of graph layout algorithms to visualize the network, but does not work very well for large networks due to the lack of space for the timestamps on the links and the large numbers of nodes.

A different idea for the visualization of a time-varying networks are contact timelines. The interactions between nodes (i.e., the tuple of nodes that are interacting with each other) are placed on one axis and the time is placed on the other one. A marker is set for each pair that interacts with each other at a certain time. This allows for the visual detection of interaction patterns. However, similar to the last representation method, this one is only

reasonable for small networks as well. Figure 2.4d shows an example for this type of representation.

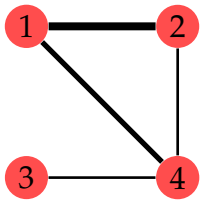
It is also noteworthy that most of the introduced measures for networks do not apply for temporal networks. They must be redefined or extended. For example, the concept of degree distributions does not exists in the context of time-varying networks. But there are new measures like the *inter-contact time distributions*, which describes the frequency of contacts between either a specific pair, or any two nodes. Paths cannot be used in temporal networks as well, and are replaced by measures like *latency* (i.e., how long since the last contact between two nodes) or *temporal distance* (i.e., how long does it take to get from one node to another while taking the contacts into account). There is also the idea, and many approaches, to extend community detection mechanisms for the usage in time-varying networks by running community detection algorithms for a static approximation of a network at time t and then refine the result by including community information from previous time steps.

2.4. Generative Network Models

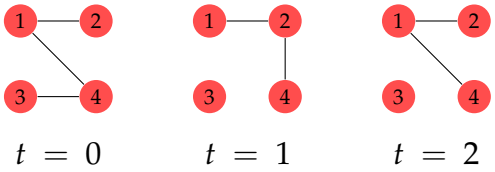
This section contains descriptions of different generative network models, which can be used to build graphs with certain characteristics. They are often a useful tool when studying real-world networks (e.g., social networks) to gain a deeper understanding on the processes that creates them.

2.4.1. The Erdős-Rényi Model

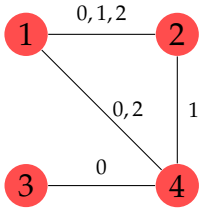
The Erdős–Rényi (ER) model [ER59; New10] was in its first form described by the two famous mathematicians Paul Erdős and Alfréd Rényi in 1959. The model generates a random graph with n nodes and m links by choosing one of the $\binom{\binom{n}{2}}{m}$ possible graphs of this size with equal probability at random. It is sometimes also denoted as the $G(n, m)$ model.



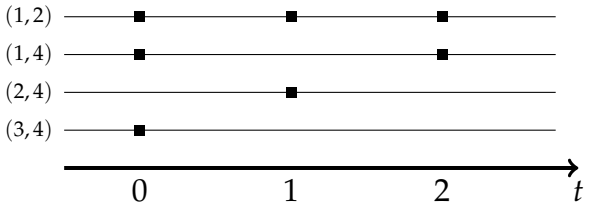
(a) Static approximation of the contact sequence as a weighted graph. The width of the lines between the nodes represents the weight (i.e., the total number of interactions between pairs of nodes).



(b) Visualization of the graph sequence representation of the three time steps of a time-varying network. There exists an edge at the graph at time t, if there was a contact between the two nodes at this time step.



(c) Visualization of C as a graph that contains an edge between two nodes if there was at least one contact between them. Furthermore, the edges are annotated with a time series of time steps that indicate when the interactions took place.



(d) Visualization of the contact sequence as a timeline of contacts. The vertical axis shows the interactions that happened between the nodes in the network and the horizontal axis shows the three time steps. There is a marker (depicted as a black rectangle) if there was a contact between the pair at a given time t.

Figure 2.4.: (b)–(d) show different possible representations for the contact sequence $C = (1, 2, 0), (1, 4, 0), (3, 4, 0), (1, 2, 1), (2, 4, 1), (1, 2, 2), (1, 4, 2)$, and (a) shows a static network of the same contact sequence, where the number of contacts between two nodes is reflected by the edge weights.

The basic idea behind this model is that large and complex networks often seem random and can be examined by using random graphs and statistical methods [ABo2]. Therefore, this very simple, yet powerful, model can be utilized to explore effects that take place in real-world systems.

There is another very similar model, the $G(n, p)$ model, in which the number of links is not fixed beforehand. The link quantity is determined by p, the probability of the presence of a link between any pair of nodes in the network. Hence, the edges of a random network are determined by flipping a biased coin for each of the $\binom{n}{2}$ possible edges.

The probability for an arbitrary network with exactly m links under the $G(n, p)$ model is $p^m(1-p)^{\binom{n}{2}-m}$. Therefore, the probability that the model will generate a network with m links is $\mathbb{P}[m] = \binom{\binom{n}{2}}{m} p^m(1-p)^{\binom{n}{2}-m}$ (i.e., the probability of such a network times the number of possible networks). This corresponds to the binomial distribution $B(\binom{n}{2}, p)$. Hence, the expected value for the number of links is

$$\mathbb{E}[m] = \sum_{m=0}^{\binom{n}{2}} m\mathbb{P}[m] = \binom{n}{2} p. \tag{2.3}$$

The expected value for the average degree c of a network generated with this model is deduced in equation (2.4). The probability that an arbitrary node has a degree of exactly k is given by $p(k) = \binom{n-1}{k} p^k(1-p)^{n-1-k}$ (i.e., k of its possible $n-1$ links must exist and there are $\binom{n-1}{k}$ possible combinations for the k links). Therefore, the degree distribution of this model is a binomial distribution as well. However, it is also possible to approximate the degree distribution with a Poisson distribution $p(k) = \exp(-c)\frac{c^k}{k!}$ for large values of n.

$$c = \mathbb{E}[d(G)] = \sum_{m=0}^{\binom{n}{2}} \frac{2m}{n}\mathbb{P}[m] = \frac{2}{n}\sum_{m=0}^{\binom{n}{2}} m\mathbb{P}[m] = \frac{2}{n}\binom{n}{2} p = (n-1)p \tag{2.4}$$

A nice property of the Erdős–Rényi model is that it can be used to study the formation of giant components. The giant component of a network is a component that contains a large fraction of the nodes. Others have adopted this model to study dynamic processes on it. For example, Wang et al. [Wan+03] are using it to study the spreading of viruses in networks, and Crucitti, Latora, and Marchiori [CLM04] for their research on cascading failures. Hence, even such a simple model can be applied to study phenomenons that are part of many real-world networks. Furthermore, it can be used to replicate the small average shortest path length, which grows logarithmically with the size of the network, and can also observed in many real-world networks.

However, there are quite a few problems with this model as well. The degrees in real-world networks are usually not binomial (or Poisson) distributed, but follow a power-law distribution. Other examples for shortcomings of the model are the inability to generate community structures and hubs.

2.4.2. The Barabási-Albert Model

One model that addresses the problem of missing power-law degree distributions is the Barabási-Albert (BA) model [AB02]. The model is named after its creators Réka Albert and Albert-László Barabási. It tries to emulate the dynamic process that is responsible for the creation of scale-free degree distributions and yields the generated network as result.

One of the main differences to the ER model is that the size of the network is not fixed. The model starts with a small number of nodes and adds new ones to the network over time. A newly added node forms links with already existing nodes. What nodes are selected depends on how important they are. The more important a node is, the more likely it is that the new nodes form connections with it. This process is called *preferential attachment*. It can, for example, be used to explain the evolution of the online encyclopedia Wikipedia [Cal+06]. Eisenberg and Levanon [EL03] showed that preferential

attachment is the mechanism behind the evolution of protein networks. Furthermore, new websites on the World-Wide-Web follow the same pattern, by linking already popular website more often [BA99].

More formally, the model starts with a small number m_0 of nodes and the following two steps are executed in every iteration:

1. Add a new node to the network.
2. Choose $m \leq m_0$ already existing nodes at random proportional to their degree and form a link between them and the newly added node.

Therefore, the degree of a node is a measure of its popularity and an existing node v_i will be selected with a probability $\Pi(v_i) = {}^{d(v_i)}/_{\sum_j d(v_j)}$. The process yields a network that consists of $m_0 + t$ nodes and mt links after t time steps.

It can be shown using different methods (e.g., continuum theory, master equations, or numerically), that the node degrees follow a power-law distribution with a degree exponent of $\gamma = 3$. Furthermore, asymptotically does γ not depend on m, the number of links that are generated in each iteration. The degree distribution is also (asymptotically) independent of the time, and therefore on the size of the network. This property of the model reflects the fact that there exists real-world networks of different sizes with power-law degree distributions.

However, like the ER model, the BA model has its shortcomings as well. For example, the average path length of generated networks does not comply with real-world networks. Another property that cannot be reproduced by this model are the community structures and the with it associated high clustering coefficient that many real-world networks (especially social networks) have [RH11].

Another interesting question regarding the Barabási-Albert model is if both mechanisms, the network growth and the preferential attachment, are necessary to produce scale-free networks. The result of numerical simulations and formal tests of the model with one of the two mechanisms disabled

shows that both are indeed required. Missing preferential attachment results in exponentially distributed node degrees, and while the absence of the network growth mechanism leads to a power-law degree distribution in the beginning, a shift to a normal distribution occurs over time [AB02]. This indicates that, in fact, both mechanisms are required to yield networks with the scale-free property.

2.4.3. Watts-Strogatz Model

The model proposed by Watts and Strogatz [WS98] is able to produce networks with the small-world property. The goal of it is to be as simple as possible while being able to replicate the effect [Wat04]. This is done by starting with a ring lattice graph with n nodes. In such a graph are the nodes arranged uniformly on a ring and are connected to their k nearest neighbors on it. This very regular graph has a high average local clustering coefficient and a high average shortest distance between any pair of nodes that grows linearly with the size of the network. Therefore, it does not have the properties of small-world networks yet.

However, a simple rewiring process can introduce the small-world effect by changing the endpoints of some randomly selected edges. This rewiring process depends on the parameter $p \in [0, 1]$, which determines the probability for an edge to be rewired. Note that this parameter is also sometimes refereed to as β, since the model is known as Watts' beta model as well.

The rewiring process itself is very simple. Each edge is rewired with probability p. The new endpoint of an edge is selected uniformly at random, but self-loops and parallel edges cannot be introduced in the network.

A rewiring probability of $p = 1$ changes the endpoint of every edge and produces a random network with properties similar to a network generated with the Erdős-Rényi model. It exhibits a small average shortest path length, but also a small average local clustering coefficient.

However, networks generated with values for the rewiring probability in the range $0 < p < 1$ can show the small-world property. The effect of the rewiring process on the average path length in the network is substantial. It drops immediately to a low value even if only a few edges get rewired. The rewiring essentially produces shortcuts in the network by connecting nodes that are usually far apart, which decreases the average distance between any two nodes dramatically while keeping the average local clustering high. Therefore, the model is able generate networks with small-world properties.

Nevertheless, if values of p gets too large, the average local clustering coefficient decreases significantly and the generated network resembles more and more a random network. Figure 2.5 illustrates the transformation of the network for increasing values of the rewiring probability.

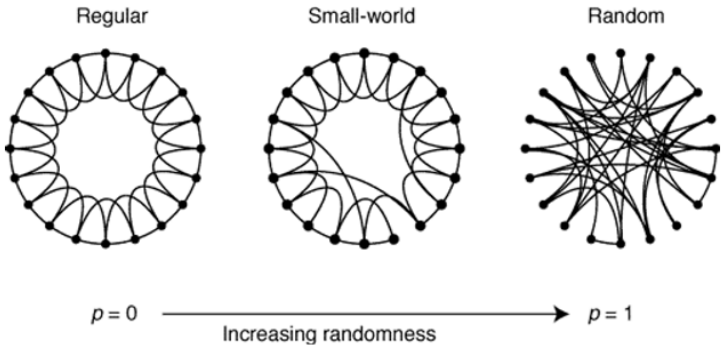


Figure 2.5.: Depiction of networks generated with the Watts-Strogatz model with an increasing value for p. The initial ring lattice network consists of $n = 20$ nodes each with degree $k = 4$. Note that to ensure that the generated network stays connected n and k should be selected in such a way that $n \gg k \gg \ln(n) \gg 1$, which is not the case for this toy example. Figure borrowed from [WS98].

2.5. User Activity Models

The modeling of the activity of users in complex systems (e.g., in social networks) can be a challenging task. Usually the specific activities, such

as writing of e-mails, posts on Facebook, or tweets, are not of particular interest. More relevant is how these events or activities are laid out in time and what the distribution of intervals between two consecutive events is (i.e., the inter-event time distribution).

More formally, the sequence of timestamps of events for user i is written as $t_{i,0}, t_{i,1}, t_{i,2}, \ldots$ with $t_{i,j} \in \mathbb{N}_0$, which is ordered in the sense that if $t_{i,l} \geq t_{i,k}$ then $l \geq k$ and vice versa. The inter-event times for the user i are defined as $\tau_{i,j} = t_{i,j} - t_{i,j-1}$, for $j = 1, 2, \ldots$ and their distribution is denoted as φ_i.

There are various models that try to capture the patterns of human activities based on different approaches and some are discussed in this section.

2.5.1. Stochastic Models

One of the simplest methods to model user activity is by using a Poisson process to describe the inter-activity times [Bar05; Váz+06; ML16]. This stochastic process is defined by the event rate λ, which states how often an event should occur in a given time window. Two important properties of the Poisson process are that the inter-activity times are exponentially distributed and independent of each other. This leads to the effect that events take place in regular intervals (i.e., at the given rate) and that it is almost impossible to have long periods of time between to consecutive activities.

However, it has been shown, that human activity patterns (e.g., email communication) cannot be modeled very accurately by a Poisson process due to its assumptions [Bar05]. Most activities are executed in bursts, followed by longer periods of inactivity. For example, a person may have a dedicated time in the day for answering emails. This behavior can be much better explained by a power-law distribution of the inter-activity times, since its long tail allows for longer periods of inactivity.

There are, however, approaches that try to tackle this problem by using extensions of Poisson processes. For example, Malmgren et al. [Mal+08] use a

mixture of homogeneous and non-homogeneous Poisson processes to model user e-mail activity more precisely. The rate of a non-homogeneous Poisson process does depend on the time, whereas the rate of a homogeneous process is constant.

An approach that does not only generate power-law distributed inter-event times, but also captures other patterns of human behavior, such as periodic spikes (e.g., higher activity every 24 hours) or a bimodal distribution of inter-event times (e.g., phases of high activity that are separated by phases of rest) was proposed by Ferraz Costa et al. [Fer+15]. Their Rest-Sleep-and-Comment (RSC) model is based on the idea that a user can be in one of multiple states. In the active state, a user generates events with a certain probability at a given rate, which depends on how much time has passed since the last event. The rest and sleep states are used to model the inactivity of a user. The difference between the two states is that the rest state produces null-events (i.e., it only increments the time) at a certain rate, whereas the sleep state is used to increment the time only once, but by a larger extent. An active user can become inactive (i.e., go to the rest state) or stay active. A user in the rest state can either become active, stay inactive, or go to the sleep state. The user cannot stay in the sleep state, only go into the rest state again. The state transition probabilities are parameters of the model. This model was the foundation for a classifier, that is able to detect whether an activity sequence was generated by a bot or by a human with very high accuracy.

Another possibility to model the user activity in social networks is by using coupled Hidden Markov Models (CHMM's) [Rag+13]. The Markov model has two hidden states that describe the user activity (active or inactive) and yields inter-activity times with respect to the current state. The CHMM model takes the social network influence of other users into account as well, by explicitly coupling the stochastic processes of groups of people. This is done by letting the transition probabilities between the states of the HMM for a single user be dependent on the activity of other users, that are in the circle of friends of the user (i.e., neighbors in the social network). If the activity of the neighbors exceeds a certain threshold the probability for a user to become active gets larger. The evaluation of this approach shows that this model is able to learn the complex human activity patterns and allows for

highly accurate predictions.

2.5.2. Queuing Models

One approach, which successfully generates human activity patterns that follow a power-law distribution, are queuing models [Váz+06]. The idea behind them is to think of a user as a queue that is constantly filled with new tasks (e.g., answer an e-mail, go shopping, do the dishes,...). Each task takes a certain amount of time to finish and is prioritized by the user on arrival. Furthermore, the queue is usually bounded in size, since people typically can only keep track of a certain amount of tasks at a time. At each time step the user selects the task with the highest priority from the queue and executes it.

It can be shown that the time it takes for a task to be handled (i.e., the waiting time) follows a power-law distribution. There is evidence that the waiting-time distribution of the queuing model is responsible for the inter-activity time power-law distribution of a specific activity. This can possibly be explained by the observation that people tend to group tasks in categories and reinserting them into the queue with a lower priority after they are done. For example, a person does not keep track of every unanswered email in the queue, but has an "answer email" task that contains all emails that need to be replied-to. Therefore, this may lead to the answering of multiple emails in a short period of time, followed by a longer period of no email correspondence, due to the presence of other tasks with higher priority.

2.5.3. Time-varying Network Models

The prior discussed user activity models are designed to solely describe the activity profile of a single person, or require at least a separate stochastic process for each one. However, there are models that allow for the description of the activities for multiple people at once using time-varying networks.

The difference between models for temporal networks and static network models (as described in section 2.4) is, that the latter are purely connectivity driven [Per+12a]. This means that they are designed to generate specific topological properties in the networks (e.g., the formation of community structures, or short average path lengths), but do not consider the dynamic processes, which are responsible for these structures.

One of the simplest methods to generate a time-varying network of user activity was proposed by Holme [Hol13]. The idea is to generate a static network and assign each link a (possibly empty) set of timestamps, that represents the contact sequence between the two nodes. To archive this, the first step is to generate a static network using the configuration model [New10]. This model assigns a number, drawn from a probability distribution (e.g., a power-law distribution), to each node in the network. This number represents the number of "half edges" of the node. These "half edges" can be seen as dangling edges that will be connected at random to other "half edges" of different nodes. Therefore, this number corresponds to the degree of the node. Self-loops and parallel edges are avoided during the matching process to generate a simple static network. Next, each link in network is assigned a time window at random, in which contacts are possible (i.e., the activity interval). In the last step of this approach, a time series of events is generated by drawing inter-event times from a power-law distribution. This time series is then split into parts and mapped onto the activity windows of the nodes, thus, generating the contact sequences.

A different approach by Perra et al. [Per+12a] is based on the idea of *activity potentials*. Each node v_i in the network of size n is assigned a quantity called the activity potential x_i, which denotes the probability that the node will be active in a time window Δt. The activity potential for a user in real-world network can be determined by calculating the ratio of the number of interactions of this user to the total number of interactions in a time window. Measuring the activity potentials of multiple users usually yields long-tailed distributions of the quantity, which corresponds to typical heterogeneous human activity patterns [Váz+06; Jo+12]. Furthermore, the size of the time window, which is used to estimate the probabilities, seems not to effect the resulting distribution in a significant way.

The first step of this activity-driven model is to initialize each node in the network by assigning it an activity/firing rate $a_i = \eta x_i$, where the activity potential is drawn from a suitable probability distribution $f(x)$ and η is a rescaling factor that is chosen in such a way that the expected number of active nodes in the time window is $\eta \mathbb{E}[x]n$. The range of possible activity potential values is $x_i \in [\varepsilon, 1]$. This lower bound ε of the activity potential is necessary to avoid possible divergences of the distribution for values that are very close to zero [CSN09].

The time-varying network can be generated after this short initial setup phase by repeating the following instructions for each time step t:

1. Create a new network G_t, that contains all n nodes but has no links yet.
2. Every node v_i becomes active with probability $a_i \Delta t$. Active nodes choose m distinct other nodes uniformly at random and form links with them.
3. Increment the current time $t \to t + \Delta t$.

Therefore, in this model the time-varying networks are represented as sequences of graphs (see section 2.3), which are called *instantaneous networks* in the context of this model.

The cardinality of the edge set at time t is given by $|E_t| = m\eta \mathbb{E}[x]n$, since every active node creates exactly m links. Therefore, the average degree of the instantaneous network at time t is

$$d(G_t) = \frac{2|E_t|}{n} = \frac{2m\eta \mathbb{E}[x]n}{n} = 2m\eta \mathbb{E}[x]. \tag{2.5}$$

The probability for a node to become active does not change over time and is, therefore, also independent of previous activities. This resembles the problem already encountered with models based on Poisson processes. The inter-event times for a node v_i will eventually be exponential distributed $\varphi_i(\tau) = a_i \exp(-a_i \tau)$ [MSP16]. Additionally, the model lacks of realism in

the sense that every node selects its neighbors in each iteration uniformly at random, which leads to networks with a random structure. Typically users are prone to repeat previous communication [KPV14].

Nevertheless, this model possesses a few considerable advantages. First, and most important, it produces not only activities for a single user, but generates snapshots of the total user activities in the network in each time window. Furthermore, it is a very simple model. It only requires a few parameters and the activity potential distribution governs the dynamical behavior in the network.

Moreover, this model can be used to explain the formation heterogeneous structures (i.e., hubs) in networks over longer periods of time. This is done by examining the *integrated network* $G_T = \bigcup_{t=0}^{T} G_t$, which is defined as the union of all instantaneous networks up to the time stamp T (see figure 2.6 for an example). It can be shown that the degree distribution of this integrated network has the form $p_T(k) \sim f(k/Tm\eta)$. Therefore, it is up to a rescaling factor the same as the activity potential distribution. Hence, the model is able to generate scale-free networks by drawing the activity potentials from a power-law distribution. The resulting hubs are not caused by preferential attachment like in other models, but due to the heterogeneous activity profiles of the nodes. The rescaling factor emerges from the fact that the model does not capture all features of real-world networks. For instance, memory effects are not present that would allow links, which were formed in earlier instantaneous networks, to be formed later again with higher probability.

Regardless of its simplicity, this activity-driven model can help to understand topological patterns and the dynamics of systems (e.g., epidemic spreading processes). Starnini and Pastor-Satorras [SP13], for example, study the topological properties of this model in a more formal way by mapping it onto a hidden variables network model. The idea behind hidden variables models is that the probability of the formation of a link between two nodes depends on some underlying characteristic of the nodes (i.e., the hidden variables). In this case, the hidden variable is the activity potential. A network that was generated using this model solely depends on probability

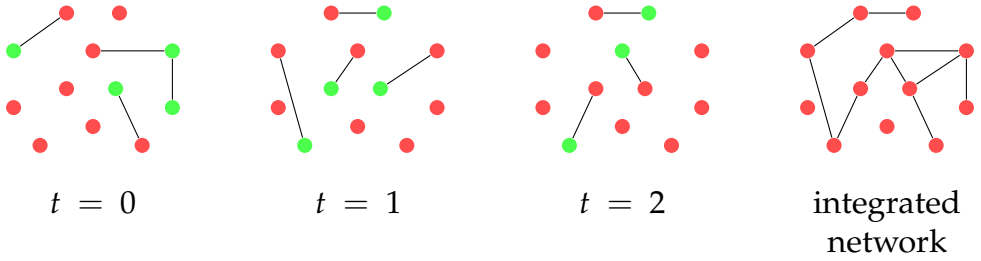


Figure 2.6.: Illustration for the formation of the integrated network for a small example with $n = 12$ nodes. The first three graphs show the instantaneous networks for the time steps $t = 0$, $t = 1$, and $t = 2$. In each time step a few nodes get active (green nodes) and establish links to $m = 1$ other nodes. The forth graph shows the integrated network up to $T = 2$. It contains all links that were established in the instantaneous networks. Parallel links are not allowed (i.e., a link is only added once to the integrated network even if it is present in multiple time steps).

distributions that are related to the hidden variables. Therefore, topological properties also depend on these probability distributions and can be expressed with respect to them. The properties of the degree distribution of the integrated network could be verified using this more formal approach. In addition, they showed that the clustering coefficient of the network is rather small and comparable to the clustering coefficient of random networks. They also propose to use the hidden-variable approach to study possible extensions of the model, which is, for example, done for the NOPAD model [MSP15], which is briefly discussed later in this section.

The activity-driven model is also a fundamental framework for many other studies that use it as start point for their work. For example, Perra et al. [Per+12b] use this model to study random walks on temporal networks, and Rizzo, Pedalino, and Porfiri [RPP16] are applying it for their research on the spreading of the infectious Ebola disease in Liberia. Another paper by Rizzo, Frasca, and Porfiri [RFP14] uses the activity-driven network framework to study how an epidemic affects the behavior of persons. They showed that a reduction of the activity potential of persons, due to the fact that they are already ill or because they are trying to protect them self from the disease, may help the slow down the spreading process. A paper on a similar topic by Liu et al. [Liu+14] proposes a framework to develop strategies on how

to contain the spreading of diseases. Mistry et al. [Mis+15] use the model to explore the spreading of opinions in social networks. They showed that activists are able to spread messages across the population more effectively and can help to reduce the cost of campaigns.

On the other hand, others try to improve the model and make it more realistic by introducing additional mechanisms. Laurent, Saramäki, and Karsai [LSK15] add additional social mechanisms (e.g., memory) to the model to allow for the formation of communities in the integrated network. This model is explained in great detail in section 3.1, since it is used as a foundation for our work presented in this thesis. Another extension by Moinet, Starnini, and Pastor-Satorras [MSP15; MSP16] solves the problem of inaccurate inter-events times by making the activity potential of each node time dependent. This model is known as the non-poissonian activity-driven (NOPAD) model and can generate inter-event time distributions that can also be observed in real-world networks.

Wang et al. [Wan+16] proposed the Activity-Security-Trust (AST) model, which not only considers activity as the explicit driving force behind dynamic processes, but also incorporates the implicit factors security and trust. Trust can be interpreted as the belief in honesty or fairness between two nodes and influences the possible link formation between them. The second mechanism, the security level, is like the activity potential a property of each node that determines how well a node is prepared against possible attacks. It can be seen as the probability of a node to accept the interaction initiated by another one.

Sunny, Kotnis, and Kuri [SKK15] add link lifetimes to the activity-driven framework. Every time a link is formed it is assigned a lifetime, that is drawn at random from a probability distribution. This link may then be part of multiple consecutive instantaneous networks and is only removed after its lifetime has decayed, in contrast to the simple activity-driven model, where links are meant to be instantaneous and are deleted after every time step. The authors use this new introduced mechanism to study how well link lifetimes are suited to model disease spreading processes.

While Laurent, Saramäki, and Karsai [LSK15] examine the local effects that

are responsible for the formation of complex network structures using the activity-driven framework, takes Alessandretti et al. [Ale+17] work also the global effects for the formation of links into account. Each node in the model is not only assigned an activity potential, but also an attractiveness quantity, which determines how popular a node is. Nodes that are more popular get selected more often from active node, which basically corresponds to a preferential attachment process. They not only show in their work that activity and the attractiveness are correlated in real-world social and collaboration networks, but also that this influences dynamical processes on the network in a significant way.

2.6. Peer Influence

Peer influence or peer pressure plays an important role in the field of sociology and psychology. There are a lot of studies that investigate how peer influence affects the behavior of people in different settings. This is not only done for real-world social networks (e.g., a social network of students in a school), but also for online social and collaboration networks, such as Facebook or StackOverflow.

These studies often investigate peer effects for harmful behavior like smoking or drinking[1] [Sim+01; PTR05]. Krasnova et al. [Kra+08] showed that peer pressure is an important factor for adolescents to participate in online social networks, and Huang et al. [Hua+14] concluded in their work that being exposed to pictures of drinking and smoking friends on social media websites may lead to an adaption of these behaviors.

However, peer influence is not necessarily a bad thing. For example, Smith and Fowler [SF84] showed that the introduction of peer monitors in a kindergarten classroom decreased non-participation and disruptive behavior of the other children. Another example is the study conducted by Christakis and Fowler [CF08]. They examined a relatively large social network of

[1] `https://xkcd.com/1534/`

approximately 12,000 people over more than 30 years and evaluated the spreading of smoking in the network. The results of this study not only showed that the number of smokers declined over the years, but also that clusters of smokers in the network seem to disappear at roughly the same time. This indicates that people, which quit smoking may trigger a cascading cessation effect in their social circle. For instance, they showed that in their data set the smoking cessation of a spouse reduces the probability for the other one to smoke by 67%.

Peer influence plays an important role in opinion dynamics models as well. The formation and diffusion of opinions in complex networks is examined in these models. Opinions can be trivial things like taste in food and entertainment, or more sophisticated issues like political views, and are usually heavily influenced by others [AO11].

One popular use case for these types of models are voter models, which try to maximize the number of voters for a political party in an election (i.e., the opinion) with a minimum amount of cost that is associated with winning voters over. For instance, the voter model by Masuda [Mas15] considers two different types of voters, the partisan voters (often also referred to as zealots), which do not change their opinion, and independent voters, which can. An independent voter adapts a political view proportionally to the presence of this view in its neighborhood and with respect to zealots that influence him or her. It can be used to determine the optimal set of nodes in the network of independent voters, that should be controlled and influenced by zealots of one party to obtain the maximum number of voters.

In another work by Estrada and Vargas-Estrada [EV13] is the effect of peer influence on reaching consensus in social groups examined. They model the influence between two user as their social distance, which is a function of the length of the shortest path between them in the social network. This allows for a node to receive indirect peer pressure from other nodes that are not directly connected to it (i.e., do not share an edge) as well. They showed that this indirect peer influence is not only important for reaching consensus in social groups, but also for social group leaders to emerge, which affects the opinions in the entire network and helps to reduce the time until consensus is reached.

Another interesting application for the usage of peer influence are network models that incorporate peer effects that influence the activity of nodes. For instance, Walk et al. [Wal+16] proposed a model that helps to study how the activity in large collaboration and social networks develops over longer time spans and what effects the micro-behavior of users (i.e., their intrinsic activity and the influence on their neighbors) has on the macro-behavior of the system. The macro-behavior of the system can be interpreted as the overall activity in the network. This helps to understand how some systems can archive a state in which they can become self-sustaining, in the sense that no external influences or stimuli are needed to keep the network permanently active, while others become non-active very quickly, even after lots of activity in the early stages of the network.

Their model is based on the idea that each user has an intrinsic activity, which declines over time. This can be interpreted as the growing loss of interest of users in the system. But there is also an opposite force, the peer influence of others, that motivates the user to contribute.

More formally, the model is defined as a dynamic process that takes place on a network, were users are represented as nodes and there exists a link between two users if they interacted or collaborated at some point. Furthermore, every user v_i is assigned a positive real-valued variable a_i, which represents the activity of the user. The dynamic process is then defined by the following system of coupled non-linear differential equations:

$$\frac{\mathrm{d}a_i}{\mathrm{d}t} = \underbrace{f(a_i)}_{\text{activity decay}} + \underbrace{\sum_{j \in N(v_i)} g(a_j)}_{\text{peer influence}} \quad \text{for } i = 1, 2, \ldots, n \tag{2.6}$$

The first part of the right hand side of the equation represents the intrinsic activity decay of a user, and the function f defines how fast the loss of activity should occur. In their work a linear function $f(a_i) = -\lambda a_i$ is used, where $\lambda > 0$ denotes the rate on which the activity decays. This leads to an exponential decay of the user activity. The second part of the equation denotes the positive peer influence of the users neighbors. This is based on

the idea that active neighbors may also increase the activity of a user, due to a boost in their motivation to participate in the network. The function g determines the level of positive peer influence of a neighbor, and is selected to be the algebraic sigmoid function $g(a_i) = {}^{q a_i}/\sqrt{a_c^2 a_j^2}$. This function introduces two additional parameter to the model, $a_c > 0$ controls what level of activity of a neighbor is required such that it influences a user in a significant way, and $q > 0$ determines the maximum amount of peer influence activity that a neighbor can transfer to another user.

In their work they perform a linear stability analysis on the system of differential equations and show that the stability of the system is determined by the relation between the activity dynamics ratio $\frac{\lambda}{\mu}$ (with $\mu = \frac{q}{a_c}$) and the topological structure of the network. This means to prevent the network from becoming or staying inactive, there must either be constant external influences to keep the activity up or the system needs to become unstable. This can be done, for example, by changing the network structure (e.g., removing links) or by manipulating the three model parameter.

Furthermore, they apply their activity dynamics model to some real-world data sets. For instance, different instances of StackExchange collaboration networks and various wikis. This is done by first estimating the ratio $\frac{\lambda}{\mu}$ from the activity in the data sets, and then simulating the activity in the networks. However, due to some limitations and approximations in the parameter estimation is a highly accurate prediction of the activity not possible. Nevertheless, the model is able to predict trends in the activity of an empirical network by simulating its activity dynamics quite well and, therefore, provide additional confidence in the validity of the model.

Additionally, two stability measures for social or collaboration networks are introduced in their work. First, the system mass is defined as the inverse of the normalized standard deviation of the activity dynamics ratio, and indicates the robustness of a system with respect to change of the activity dynamics parameter. The second measure is called activity momentum and is defined as product of the system mass and the over some time span averaged last observed activity in the network, and is an improved indicator for the robustness. For instance, a system with a large activity momentum

value handles the sudden inactivity of a considerable number of users better, since it can retain most of its activity. Therefore, much more "force" is needed to render a network with high activity momentum inactive.

Another interesting question in the context of peer influence effects in networks was raised in the work of Aral, Muchnik, and Sundararajan [AMS09]. They suggested that the behavior of a user in a network may not be entirely driven by the influence of its peers, but can be explained in large parts by homophily. Homophily (also known as assortativity or assortative mixing) is the effect that users tend to interact with other users that are similar to them self. They examined a temporal network of instant-message communication between users and studied how the adaption of a new feature in the messaging application could be explained. The idea behind their work is that users may not adapt the feature shortly after their friends because of peer influence, but do it simply because they behave similar to them anyway. For instance, users that have the property of being early-adopters are more likely to be friends with early-adopters as well, and therefore, start using the feature at about the same time. They proposed a statistical framework that is able to distinguish between peer influence and homophily, and showed that other methods overestimate the share of influenced adaption by a factor of four to eight, and that at least half of the adaption in their instant-messaging data set was introduced by homophily.

3. Model

The by us proposed user activity model with peer influence effects is described in detail in this chapter. It is yet another extension of the activity-driven network framework by Perra et al. [Per+12a], that was discussed in section 2.5.3. However, it is not directly based on it, but on the work of Laurent, Saramäki, and Karsai [LSK15], which relies on the activity-driven framework as well. This underlying model, which is explained in the first section of this chapter, allows for the formation of community structures and the development of strong and weak ties in the network. These two properties are crucial conditions for the occurrence of peer influenced activity. The additionally introduced mechanisms, which allow for peer influenced activity to actually happen in the network, are then discussed in section 3.2 of this chapter.

3.1. The Underlying Model

3.1.1. Description

Since this community-oriented model [LSK15] is based on the activity-driven framework by Perra et al. [Per+12a], an activity potential $a_i = \eta x_i$, which is drawn from a suitable distribution, is assigned to each of the n nodes. To reflect the heterogeneous activity patterns of people, a power-law distribution $f(x) \sim x^{-\gamma}$ is selected. The exponent for the activity potential distribution is fixed to $\gamma = 2.7$, which is a value that is similar to the exponent observed in real world communication networks. The lower

bound for the activity potential is set to $\varepsilon = 10^{-3}$, and the time rescaling parameter is fixed to $\eta = 1$, so that $a_i = x_i \in [\varepsilon, 1]$. Furthermore, for the sake of simplicity, and without loss of generality, we fix the size of the time window Δt and the number of generated links at each activation m to the value 1 as well.

The dynamic element of the model is basically identical to the activity-driven framework. A node becomes active in every time window with probability equal to its activity potential and selects other nodes to interact with. However, the intention of this model is it to produce adjustable community structures and weight-topological correlations in the integrated network. This can ultimately be archived by changing the way the communication partners are selected once a node becomes active. More specifically, to archive these more realistic topological properties, the following two additional social mechanisms are introduced:

1. Memory effects
2. Closure processes

Memory Effects The first mechanism introduces memory to the nodes, in the sense that nodes remember all previous interactions with other nodes. This idea was adapted from the work of Karsai, Perra, and Vespignani [KPV14], and enables the formation of strong ties (i.e., interactions that are repeated often) and weak ties (i.e., interactions that area repeated infrequently) between the nodes. This heterogeneity of tie strengths is an important role for processes that take place in many real world networks. For instance, they showed in their original paper that strong ties are able to slow down the spreading of rumors in networks of social interactions (e.g., a network of phone calls). This counter intuitive result can be explained by the observation that most activity is concentrated and contained in strongly tied groups, which inhibits a fast spreading of the rumor into other parts of the network.

The memory of a node is represented by a weighted egocentric network that includes all other nodes, which were already part of one or more

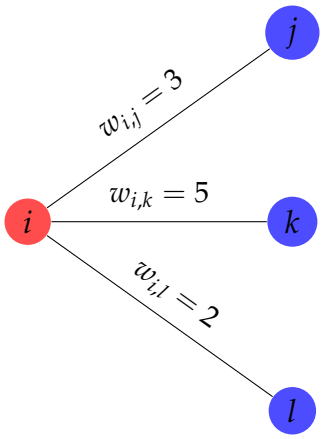


Figure 3.1.: Egocentric network of the node v_i (red node) and its neighbors v_j, v_k, and v_l (blue nodes). The link-reinforcement increment in this example is set to $\delta = 1$. This means that the weights exactly correspond to the number of previous interactions between the nodes.

interactions in the past. These previous communication partners are also called neighbors. The weight represents the number of previous interactions scaled by the link-reinforcement constant δ. More specifically, the weight is set to 1 for the first contact, and incremented by δ for every additional contact. Figure 3.1 shows an exemplary egocentric network of the node v_i and its neighbors.

A node's choice between forming a new tie and reinforcing an existing one depends on the number of its neighbors. This corresponds to the observations in social interaction dynamics, where actors tend to communicate almost exclusively within their social cycle, which has a limited size, due to cognitive capacities of the actors [Dun92]. Let $k_i = d(v_i)$ be the degree of the node v_i in its egocentric network. The probability for a node to form a new tie is given by $p(k_i) = {}^{c}/_{(k_i+c)}$, and the probability to reinforce an already established tie is $\bar{p}(k_i) = 1 - p(k_i) = {}^{k_i}/_{(k_i+c)}$, where the constant c determines the memory strength of an actor (cf. figure 3.2 for details). Therefore, the probability for the formation of a new tie decays very fast with increasing size of the egocentric network. The memory strength constant is fixed to $c = 1$ in the context of our work.

If an active node decides to reinforce an existing tie, the neighbor is selected at random with respect to its tie strength (i.e., the number of previous interactions). Therefore, the probability for node v_j to be selected as communication partner by node v_i is given by $p_{i,j} = {}^{w_{i,j}}/_{\sum_{k \in N(v_i)} w_{i,k}}$, where $w_{i,j}$ denotes the tie strength between node v_i and v_j in the egocentric network of v_i. This reinforcement process allows for the introduction of dependencies between successive interactions of node pairs, and replaces the approach of the original activity-driven framework in which communication partners are selected uniformly at random.

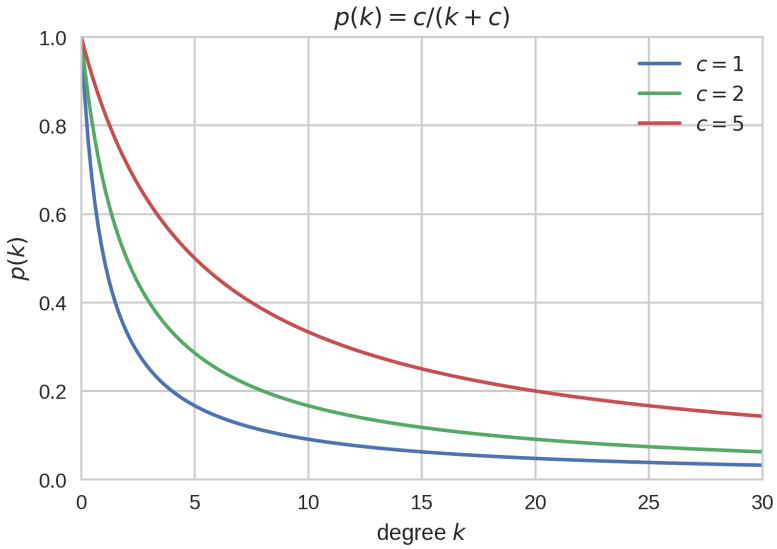


Figure 3.2.: Plots of the function that determines the probability for the formation of a new tie based on the degree of a node $p(k)$ for different values of the memory strength c. This constant can be used to model different types of users. Larger values may correspond to social explorers, that are more prone to form new ties, and smaller values are related to social keepers, which communicate almost exclusively to peers in their social circle [Mir+13].

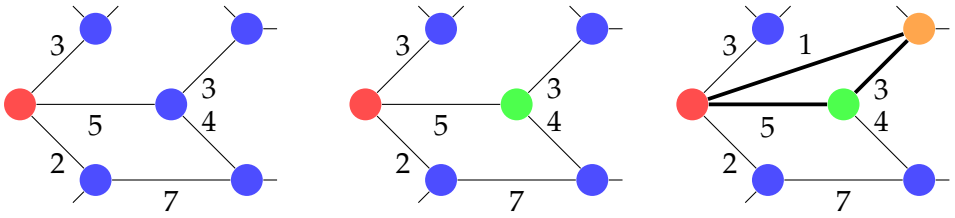
Closure Processes The second mechanism introduces two different closure processes to the model. The first one, *cyclic closure*, assures the formation of triangles (i.e., cliques between three nodes), which were linked to the formation of community structures networks by Bianconi et al. [Bia+14].

They showed that this mechanism is sufficient to generate networks with complex topological structures (e.g., long-tailed degree distributions), in which the strength of communities depends on the cyclic closure probability. If a node wants to form a new tie, it tries to perform a cyclic closure with probability p_Δ, by interacting with a randomly selected neighbor of a neighbor.

The second closure process, *focal closure*, tries to emulate the social dynamic that users tend to form ties with other users that are similar to them (e.g., if they share common interests). This process is performed whenever a new tie should be created with a probability of $1 - p_\Delta$, or if there are no suitable candidates for a cyclical closure available. This is, for instance, the case if a node becomes active for the first time. The weight of a new tie is always initialized to 1, regardless of the type of closure that was used to establish the tie.

The actual implementation of these two closure mechanisms was adapted from the work of Kumpula et al. [Kum+07], who used the same closure mechanisms to study the formation of community structures in weighted static networks. They model cyclic closure as a biased local search (cf. figure 3.3 for details) and focal closure as an unbiased global search, which means selecting a new node uniformly at random from the entire network. Furthermore, they introduced a node deletion mechanism, which was adapted in this model as well.

Node Deletion Mechanism In the activity-driven framework, nodes live forever and are, therefore, forever part of the network. However, in this extended version of the model, nodes have an intrinsic probability p_d to be removed in every time step, which is the same for every node in the network. This ensures that the network can reach a stable state, in which the structural characteristics (e.g., the community structures) become invariant in time. Every time a node is removed from the network, a new one joins to keep the size of the network constant. The deletion probability of nodes determines how fast the network reaches its equilibrium. A small value for p_d allows for nodes to stay a long time in the network and even nodes



(a) (b) (c)

Figure 3.3.: This is an illustration of the cyclic closure mechanism of the model. The network depicted in these figures is part of the union of all egocentric networks (i.e., the integrated network). (a) shows the active node in red. In the first step, this node has to select one of his neighbors. This is done at random with respect to the tie strengths. Therefore, the probabilities for the three neighbors to be selected are $3/10$, $5/10$, and $2/10$ respectively. In this example the neighbor with the highest probability was selected, which is depicted in (b) (green node). Since the selected neighbor has neighbors himself that do not share a link with the active node yet, the cyclic closure can be completed. This is done once more by selecting one of the candidates at random with respect to the weight of the ties and creating a new tie with unit strength with probability p_Δ. (c) shows they selected node (orange node) and the newly formed triangle in the network.

with relatively small activity potential can become fully integrated in the community structures. Therefore, it takes longer to reach the time invariant state if the low activity nodes are not removed fast enough.

The expected time that a node will be part of the network can be determined by viewing a node's lifetime as a simple Bernoulli process. In each iteration a biased coin is tossed for every node. The outcome of this Bernoulli random experiment determines if the node stays in the network, or is replaced in the next round. The probability for a node to be deleted after exactly x iterations is $\mathbb{P}[x] = p_d(1-p_d)^{x-1} = p_d\bar{p}_d^{x-1}$. Hence, the expected value for the lifetime of a node is given by $\mathbb{E}[x] = \sum_{x=1}^{\infty} x\mathbb{P}[x] = \sum_{x=1}^{\infty} xp_d\bar{p}_d^{x-1} = p_d\sum_{x=1}^{\infty} x\bar{p}_d^{x-1}$. This sum is related to the sum of the geometric series $\sum_{x=0}^{\infty} r^x = \frac{1}{1-r}$, for $|r| < 1$, by being its first derivative[1]. Therefore, the expected value for the

[1]The first derivative of the sum of the geometric series is $\frac{\mathrm{d}\sum_{x=0}^{\infty} r^x}{\mathrm{d}r} = \sum_{x=0}^{\infty}\frac{\mathrm{d}r^x}{\mathrm{d}r} = \sum_{x=1}^{\infty} xr^{x-1} = \frac{\mathrm{d}\frac{1}{1-r}}{\mathrm{d}r} = \frac{1}{(1-r)^2}$, for $|r| < 1$.

lifetime of a node is $\mathbb{E}[x] = p_d/(1-\bar{p}_d)^2 = 1/p_d$. This means that, for example, nodes with a deletion probability of $p_d = 5\text{e}{-5}$ will be on average deleted after 20,000 iterations.

3.1.2. Properties

The properties of this model are examined by analyzing an extended version of the integrated network. This is very similar to the basic framework, in which the temporal network is represented as a sequence of graphs. These graphs are denoted as instantaneous networks, and the union of these networks up to a time step T is called the integrated network. This is also true for this extension, however, the links in the integrated network have an additional weight assigned to them, which corresponds to the tie strength in the egocentric networks of the nodes. Another equivalent way to define the integrated network is the union of all egocentric networks up to some time step T.

These newly introduced mechanisms have interesting effects on how the structures in the integrated network evolve over time. In the beginning, after nodes formed their first ties, they start to close triangles and reinforce the ties in their egocentric network. This means that strong community structures are formed early in the process. However, after a while more and more weak ties are introduced and fewer triangles are closed, so that the strength of the communities declines and the network reaches its equilibrium state. As mentioned earlier, the node deletion probability can be used to control the time until the network converges, but it can also be used to tune the strength of the communities and the average degree of the network. A smaller value for p_d decreased the average local clustering coefficient and increases the average degree.

The cyclic closure probability p_Δ and the reinforcement increment δ control the formation of communities as well (see figure 3.4). Furthermore, like the node deletion probability, the two parameter have an effect on the average degree of the converged network. Higher values for the cyclic

closure probability or the tie reinforcement increment result in a smaller average degree. However, the two parameter do not influence the actual (heterogeneous) distribution of the degrees in a significant way. The tie strengths, which are power-law distributed, are not affected by p_Δ, and larger values for δ only influence the length of the tail of the distribution.

Another characteristic of this model is the larger impact of the cyclic closure probability on the emerging community structures compared to the tie reinforcement increment. This is true since p_Δ affects the number of triangles directly, whereas δ is responsible for the creation of strong ties, which increases the bias in the local search, and only assists in the process of finding suitable nodes for the triangle formation. Additionally, the model is able to produce higher-order correlations, that are observable in real-world networks as well. For example, weight-topology correlations (i.e., stronger ties within groups) are measurable and are dependent on p_Δ and δ as well.

3.2. Peer Influence Extension

3.2.1. Idea and Incentive

So far, the activity in temporal networks was entirely determined by the activity-potential distribution. Each node is assigned an intrinsic probability to become active in each round, which is drawn from this distribution. The assignment is done only once for every node when it is created, and does not change afterwards. After that, a node can become active in an iteration either by himself or by being contacted by another active node.

The process of becoming active on one's own accord does not dependent on whether or not the node was active in previous time steps. This corresponds to a memory-less Poisson process and leads necessarily to exponentially distributed times between two consecutive activations of a node (i.e., inter-event times). The significance of complex long-tailed inter-event time distributions in human behavior was already discussed in section 2.5.

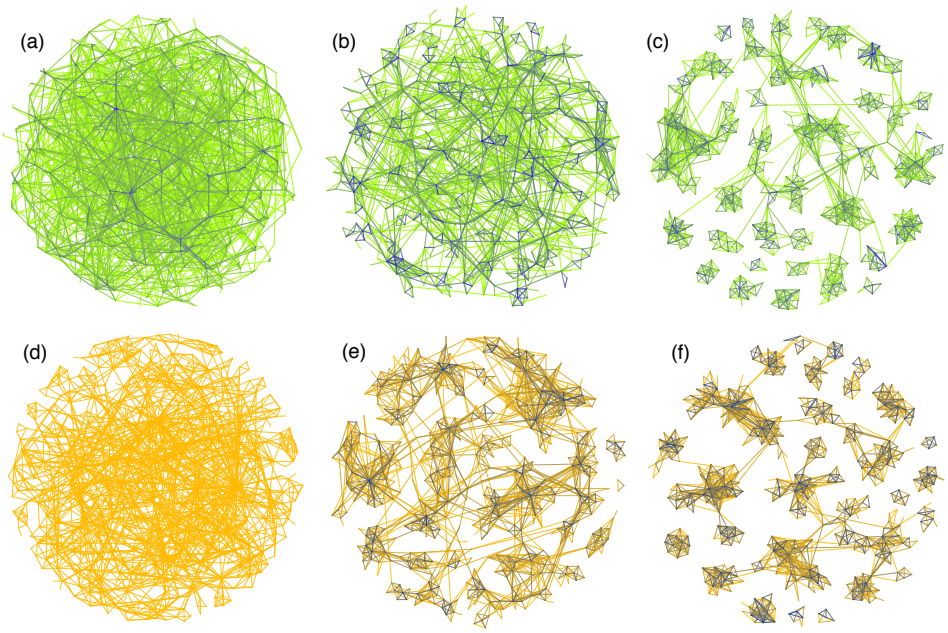


Figure 3.4.: Depiction of the influence of p_Δ and δ on the resulting community structures (image borrowed from [LSK15]). The networks in the first row (a)–(c) were generated with a fixed value for the link reinforcement increment $\delta = 1$ and varying values for the cyclic closure probability (from left to right: $p_\Delta = 0.5$, $p_\Delta = 0.9$, and $p_\Delta = 0.995$). This shows that p_Δ directly influences the strength of the communities. Furthermore, tie strength heterogeneities are observable, with strong ties within communities (darker link color) and weak ties between them (brighter link color).The second row (d)–(f) shows networks with a fixed cyclic closure probability $p_\Delta = 0.995$ and different reinforcement constants (from left to right: $\delta = 0$, $\delta = 0.5$, and $\delta = 1.5$). This shows that a high probability for the formation of triangles is not sufficient for the formation of communities. The reinforcement process, which helps to develop strong ties, is required as well and also affects the size of the communities.

However, it is evident that activations caused by other nodes are not necessarily independent of previous events, due to the memory effects introduced in the model. For example, lets assume a node with very low activity potential is part of a group of high-activity nodes with already established strong ties. The self-activation rate of the low-activity node will be quite low, with a mean value and standard deviation of the inter-event times that is equal to the inverse of its activity probability.

Nevertheless, the other nodes in the group will fairly often select the low-active node as communication partner when they become active, due the biased local search. This can, of course, alter the inter-event time distributions of nodes with a small activity potential in a significant way, compared to the distributions that are generated by the activity-driven framework, where the activation through other nodes happens completely at random. This, in this case implicit, influence that nodes have on the activity of their neighbors is an interesting effect and was a starting point for our thesis. In this section, an extension of the prior discussed underlying model with memory and closure effects is presented, which tries to model the influences of peers in the local network in a more explicit way.

The ideas for our model were heavily influenced by the work of Walk et al. [Wal+16]. They proposed a model that includes peer-influence effects and examine its impact on the global activity in collaboration networks. A more detailed description of their work is located in section 2.6. The gist of the by us proposed extension to the community-oriented activity-driven model is that a node cannot only become active on its own based on its activity potential, but also by being motivated to become active by its neighbors, that were active in the previous iteration.

Another way to look at it is that active nodes are able to influence their neighbors to become active as well in the next round. Therefore, introducing a more explicit peer influence mechanism to the model. This also means that a node now can become active in three different ways. First, it can become active by himself either due to its intrinsic fixed activity probability or due to the influence of the nodes in its egocentric network, or it can become active in a passive way by being contacted by another active node.

3.2.2. Description

The peer influence that a node v_i receives from its neighbors is denoted as p_i. Like the activity potential, p_i is a probability for an activation as well, but instead of being fixed, it may vary in each iteration based on the number of active neighbors in the last round. Therefore, a more appropriate notation is $p_i(t)$. To adjust how much peer influence a node can receive at most, an upper bound q is defined, such that $\forall t : 0 \leq p_i(t) \leq q \leq 1$. It denotes the maximal probability for an activation motivated by the neighbors of a node.

Since the peer influence probability depends on the neighbors that were active in the last round, the information of the last activations of the nodes must be stored as well. This is done by extending the egocentric networks to save the timestamp of the last activation for each node. The time of last activation of v_i is called t_i and is updated independently of the type of activation (i.e., due to self activation, or by being contacted by another active node).

Furthermore, the peer influence probability should not only be dependent on how many neighbors of a node were active in the round before, but also on the strength of the ties between them. A neighbor that shares a strong connection with a node should be more influential than, for example, neighbors that were only recently introduced to the egocentric network. This can easily be described by a weighted fraction of the last active neighbors. However, to make the model more general and adaptable the following version of the weighted fraction of active neighbors at time $t-1$ is used, that transforms the weights beforehand. Each weight in the egocentric networks of v_i is transformed and normalized by

$$w'_{i,j} = \frac{\exp(\beta w_{i,j})}{\sum_{k \in N(v_i)} \exp(\beta w_{i,k})}, \tag{3.1}$$

where β is a free parameter.

This corresponds to applying the softmax or normalized exponential function [Bis06] to each weight. The softmax function is strongly related to the Boltzmann distribution [LA87], which describes the probability for states in a physical system (e.g., particles in a magnetic field) with respect to the system's temperature and the energy of its states. Low energy states have a higher probability in the distribution, and as the temperature of the system gets close to zero, the probability for the state with the lowest energy is almost equal to one. Whereas, in a high temperature system all states are nearly equiprobable.

The free parameter β in the softmax function is called inverse temperature and replicates the behavior of physical systems. It is defined as the reciprocal value of the temperature. This softmax function is used in many applications in different fields besides physics as well. For instance, it is used in the machine learning area of reinforcement learning, where the actions of agents in some setting are selected with respect to the probabilities yielded by the softmax function. Crites and Barto [CB98] apply this method to control a group of elevators using multiple agents. It is also, for example, used in the modeling of the decision making behavior of humans in economic settings [Ray+08].

The usage of the softmax function in our model allows for different influence scenarios. For example, a value of $\beta = 0$ (this would correspond to a very high temperature) would scale every weight to the same value, which means that every active neighbor influences a node equally, regardless of the tie strength. However, it also possible to make β time-dependent (i.e., $\beta(t)$), similar to a physical system that is cooling off or heating up. For instance, for most of our experiments is the temperature at time t set to the average weight in the integrated network $G_T = \sum_{i=0}^{t} G_i$ (i.e., $\beta(t) = (\frac{1}{m} \sum_{(i,j) \in E(G_T)} w_{i,j})^{-1}$). The normalized weights $w'_{i,j}$ are also part of the extended egocentric network (c.f. figure 3.5) and must be updated at the end of every iteration. The weighted fraction of active neighbors $\alpha_i(t)$ of node v_i at time $t-1$ is then given by

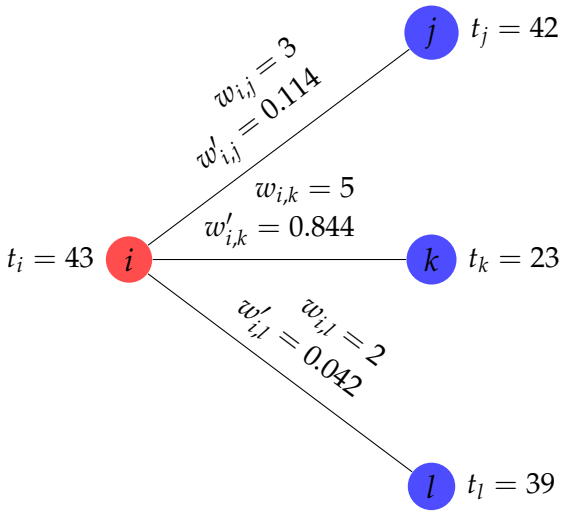


Figure 3.5.: Extended egocentric network of the node v_i (red node) and its neighbors v_j, v_k, and v_l (blue nodes). Each node stores additionally the point in time on which it was last active. For instance, node v_j was last active at $t = 42$. Furthermore, the scaled and normalized weights $w'_{i,j}$ are part of the network. For example, the scaled weight for the tie between v_i and v_j can be calculated by $w'_{i,j} = \exp(3)/\exp(3) + \exp(5) + \exp(3) = 0.114$. For the sake of simplicity is $\beta = 1$ in this example.

$$\alpha_i(t) = \frac{\sum_{j \in N(v_i)} \mathbf{1}_{\{t_j = t-1\}} \exp(\beta(t) w_{i,j})}{\sum_{j \in N(v_i)} \exp(\beta(t) w_{i,j})} = \frac{\sum_{j \in N(v_i)} \mathbf{1}_{\{t_j = t-1\}} w'_{i,j}}{\sum_{j \in N(v_i)} w'_{i,j}}, \tag{3.2}$$

where $\mathbf{1}_{\{x\}}$ is the indicator function, which yields the value 1 every time the predicate x is true, otherwise it is 0.

Next, the weighted fraction of prior active neighbors must be mapped to a peer influence probability in the range $[0, q]$ using a monotonically increasing function. One possibility would be the usage of a linear function $g(\alpha) = q\alpha$ defined for values $0 \leq \alpha \leq 1$. However, this function seems to not capture the peer influence mechanism very well. On one hand, an additional active neighbor should not be heavily influential when there is already a large portion of neighbors active. The peer influence should saturate after

some fraction of active neighbors is reached. On the other hand, the peer influence for a node should become noticeable after some threshold of active neighbors is reached.

These requirements can be satisfied by using a sigmoid function. Similar to [Wal+16], the following algebraic sigmoid function g is used to determine the peer influence for the node v_i:

$$p_i(t) = g(\alpha_i(t)) = \frac{\alpha_i(t)q}{\sqrt{\alpha_i^2(t) + \theta^2}}, \tag{3.3}$$

where the parameter q is the maximum peer influence, as discussed prior, and $\theta > 0$ denotes a critical threshold, which determines the required (weighted) fraction of active neighbors to set the peer influence probability close to its maximum. Therefore, active neighbors always affect the peer influence probability, but only after a certain point in a significant way. The satisfaction of the prior described requirements for the function can be verified by examining its first derivative

$$\frac{\mathrm{d}g}{\mathrm{d}\alpha} = \frac{q\theta^2}{\left(\theta^2 + \alpha^2\right)^{3/2}}, \tag{3.4}$$

which approaches zero very fast for values grater than q.

Sigmoid functions are usually defined for all real values, since they are often used to rescale values to the range $[-1, 1]$ or $[0, 1]$. However, in our model only input values on the unit interval are relevant and, therefore, the two parameter q and θ should be selected carefully to archive the desired peer influence behavior. For instance, too large values for θ may obstruct the peer influence mechanism critically, since the maximum influence may never be reached, even if $\alpha = 1$. Figure 3.6 shows the discussed sigmoid function with different values for the critical threshold.

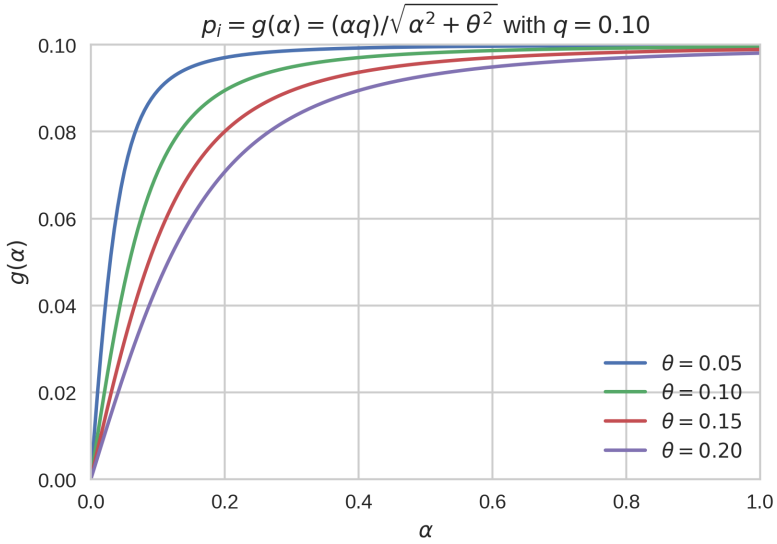


Figure 3.6.: Depiction of the sigmoid function that is used to calculate the peer influence probability p_i of a node based on its (weighted) fraction of active neighbors α for different values of the critical threshold θ. The maximum possible peer influence probability in this example is fixed to $q = 0.10$. The critical threshold determines how fast the maximum peer probability can be reached. Values in the range between 5% and 20% seem to be a sound choice, since already a small number of influential neighbors should suffice to have a notable effect on a user.

The self-activation mechanism for nodes must be adapted as well. The simple biased coin flip becomes a more sophisticated two-step random experiment. First, like in the base model, a biased coin is tossed to determine if a node becomes active with probability that corresponds to its activity potential. If this first random experiment fails, the node gets a second chance to become active by himself. This is done by calculating the node's peer influence probability and tossing another biased coin. Therefore, the total probability for a node v_i to become active at time t can be expressed by

$$\mathbb{P}[v_i \text{ becomes active}] = a_i + (1 - a_i)p_i. \tag{3.5}$$

This concludes the definition of the extension for the community-oriented activity-driven time-varying network model that introduces a new peer influence mechanism. Table 3.1 contains an overview of all model parameter and their origin. Note that many of them (e.g., Δt, η, ...) have reasonable default values assigned to them in the context of our work.

parameter	description
	activity-driven framework
n	The number of nodes in the network.
$f(x)$	The probability distribution for the activity potentials of the nodes. It yields a positive probability for values in the range of $[\varepsilon, 1]$.
ε	The lower bound of the activity potential. It should be $0 < \varepsilon \ll 1$.
Δt	The length of the time window in each iteration.
η	A rescaling factor for the activity potential to adjust the average number of active nodes in each iteration.
m	The number of contacts a node initiates, once it become active.
	community structure extension
p_Δ	The probability to form a triangle when establishing a new tie.
p_d	The probability for node to get deleted in an iteration.
δ	The constant value that is added to the tie strength when it is reinforced.
c	The memory constant, which influences the probability to form a new tie.
	peer influence extension
β	Inverse temperature parameter for the softmax rescaling of the weights. This parameter may also be time dependent.
q	The maximum possible peer influence probability that a node can receive.
θ	The critical threshold for the peer influence.

Table 3.1.: An overview of the parameter set of the proposed model.

4. Results

This chapter discusses the basic findings of the analysis of the by us proposed peer influence model. All results were obtained from synthetic networks, with a fixed size of $n = 5,000$ nodes, which were generated over $T = 75,000$ iterations. The properties that we present were obtained by averaging the results of 40 independent runs, which is necessary due to the stochastic nature of the model.

The model parameter responsible for the formation of the community structures was set to $p_\Delta = 0.90$ for the triadic closure probability, $\delta = 1$ for the link reinforcement constant, and $p_d = 5\text{e}{-5}$ for the node deletion probability for every experiment. Furthermore, the critical peer influence threshold was fixed to $\theta = 0.10$. This reflects the idea that only a relatively small number of active neighbors is sufficient to affect the activity of a node in a significant way.

The time-dependent topological properties of the integrated network, which are discussed in section 4.1, are measured only for nodes that are part of the temporal network. This means that nodes, which were removed earlier due to the node deletion mechanism do not influence the properties of the integrated network any more. Section 4.2 contains an overview of the overall network activity with respect to different levels of peer influence. The effect of the peer influence mechanism on the inter-event time distribution in the network is examined in section 4.3. Furthermore, the distributions of the node degrees and tie strengths (i.e., the link weights) derived from the integrated networks are the subject of section 4.4. All these experiments are performed for different values for the maximum peer influence probability q.

However, for the analysis performed in the last section of this chapter (section 4.5) we use a fixed value for the peer influence level and discuss how different values for β, the inverse temperature for the softmax weight rescaling, change the peer influence effects in the network. The generated temporal networks, which are used in the experiments of the first four sections of this chapter, adopt the current average tie strength as temperature for the softmax weight rescaling. Therefore, β is set to the to the reciprocal value of average weight in the integrated network in each iteration after the first one (i.e., $\beta = 1$ in the initial round to avoid division by zero, since no ties have been formed yet).

4.1. Time-dependent Integrated Network Properties

Not only the final integrated network of all 75,000 instantaneous networks and its properties are of great interest, but also how they evolve during the simulation. This time-dependent view on the network allows us to get a deeper understanding on how the model shapes the communities and the effects of the peer influence mechanism on these structures. The integrated network is build in an iterative fashion to make these observations possible. A snapshot of the integrated network is taken after the newly formed ties are included and the weights of already established links are updated in every time step. Measures like the average local clustering coefficient or the average weight of the ties are calculated for each of the 75,000 integrated network snapshots. This allows us to examine how the topology and other measures change over time.

The first, and most interesting, measure that we investigate is the average local clustering coefficient $C(t)$. Figure 4.1 depicts the development of $C(t)$ over the course of the simulation for different levels of peer influence. The graph of this function has a very distinctive pattern, which was already explained in the original work by Laurent, Saramäki, and Karsai [LSK15]. The average clustering coefficient is very small in the first few hundred

iterations, due to the sparsity of the integrated network. Almost all nodes are disconnected and the number of triangles, which have already formed is relatively small compared to the size of the network.

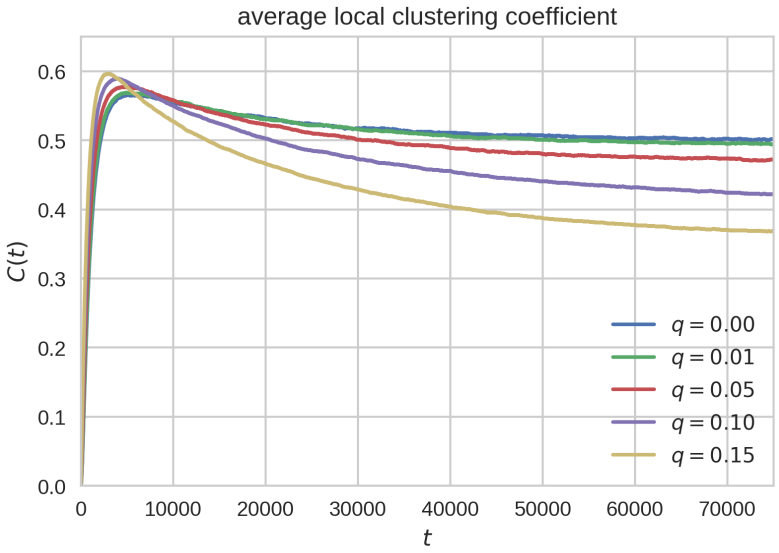


Figure 4.1.: The average local clustering coefficient C as a function of time for different maximum peer influence probabilities $q = 0, 0.01, 0.05, 0.1, 0.15$.

However, figure 4.1 also shows that the clustering coefficient grows very quickly until it reaches a maximum value after a short period of time. This rapid increase is caused by the cyclic closure mechanism of the model. Nodes that become active in this early stage first introduce some ties with selected nodes following the focal closure mechanism. This does not increase the average local clustering coefficient in a significant way, however, it establishes the egocentric networks. After the first triangles are closed, the first strong ties start to develop. These emerging strong ties amplify the biased local search of the triadic closure mechanism and result, on one hand, in more triangles, and on the other hand, in the reinforcement of already established triangles and the associated strong ties. This leads to a higher local clustering in the communities.

Nevertheless, weak ties are eventually introduced to the network by the focal closure mechanism as well. They are rarely involved in the formation

q	0.00	0.01	0.025	0.05	0.075	0.10	0.15
$t_{\max}$	5,140	4,919	4,839	5,192	4,173	4,044	3,038
$C_{\max}$	0.5659	0.5689	0.5721	0.5773	0.5822	0.5895	0.5963

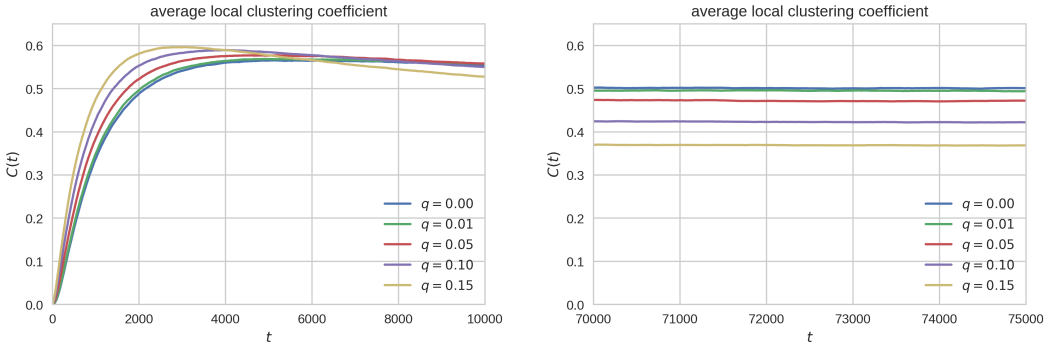
Table 4.1.: The maximum value for the local clustering coefficient $C_{\max} = \max C(t)$ and the time to reach the maximum $t_{\max} = \arg\max C(t)$, for different values of q.

of new triangles, due to the bias towards strong ties, which contributes to the decrease of the average local clustering coefficient until the network reaches its stationary state.

The by us introduced peer influence mechanism seems to influence the development of the local clustering in the beginning of the simulation significantly. Figure 4.2a depicts the time-dependent average local clustering coefficient in the initial phase of the simulation (i.e., for the first 10,000 iterations) for a range of possible values for q. It shows that the peak of C is reached faster for networks in which nodes are able to motivate their neighbors to a larger extent.

For instance, the network in which nodes are not able to influence their neighbors reaches it peak value for C after approximately 5,000 iterations, while the network with a maximum peer influence probability of $q = 0.15$ is more than 2,000 iterations faster. However, the effect only occurs for networks with $q > 0.05$ and the actual maximum value of the local clustering coefficient only increases slightly for higher levels of peer influence (cf. table 4.1 for the precise figures). Nevertheless, the proposed mechanism seems to have a positive effect on the development of the topological structures in the network by accelerating the process in the beginning.

The percentage of network activity, which is responsible for the reinforcement of ties in each iteration $r(t)$, highlights the observed behavior as well. In the beginning most activity is spent on the formation of new ties, and therefore building the topological structures of the network, but after a while more and more activity is focused on reinforcing existing ties, which leads to a drastic increase of $r(t)$. This is also emphasized by the time required to reach the point from which on more than half of the total activity is

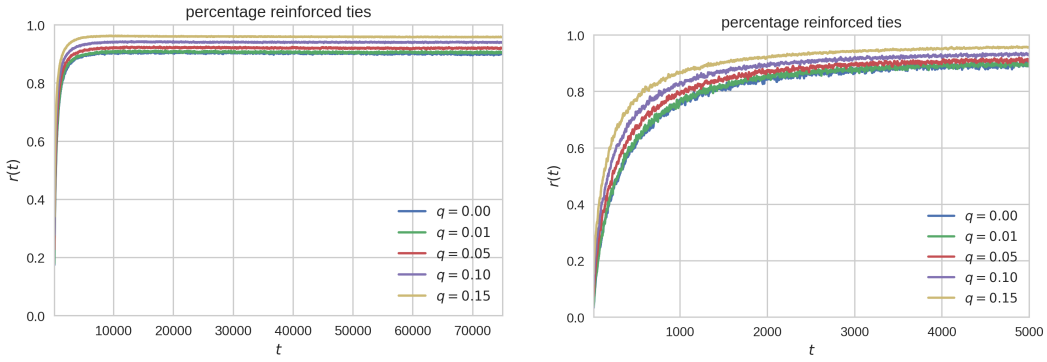


(a) (b)

Figure 4.2.: Segments in the evolution of the local clustering coefficient for different levels of peer influence. (a) shows the clustering coefficient for the first 10,000 iterations of the simulation, in which it reaches it maximum value and slowly starts to decrease. (b) depicts the stationary values for C, which can be observed in the last 5,000 iterations.

spend on reinforcement. This point is reached for the network with $q = 0.15$ after only 109 iterations. The network with no peer influence takes with 229 iterations about twice as long, indicating that peer influence may play an important role in the first few iterations that shape the topology of the network.

The share of reinforcement activity does converge to a value of over 90% for all levels of peer influence after a short period of time, which highlights the domination of the reinforcement process. Figure 4.3 shows the plots for the percentage of reinforcement activity over time in the initial phase of the simulation and over all 75,000 iterations. The plots of $r(t)$ also reveal a link between the extent of reinforcement that is happening and the degree of peer influence. In the network with no peer influence effects is the proportion of reinforced to created ties on average 0.8961. The value increases to 0.9550 in the network with a maximum peer influence probability of 15%. This can possibly be attributed to the overall increased activity within communities, due to the peer influence effects, and the associated bias towards local strong ties. With other words, the increased local activity overshadows the introduction of random links by low-activity and/or poorly integrated nodes.



(a) (b)

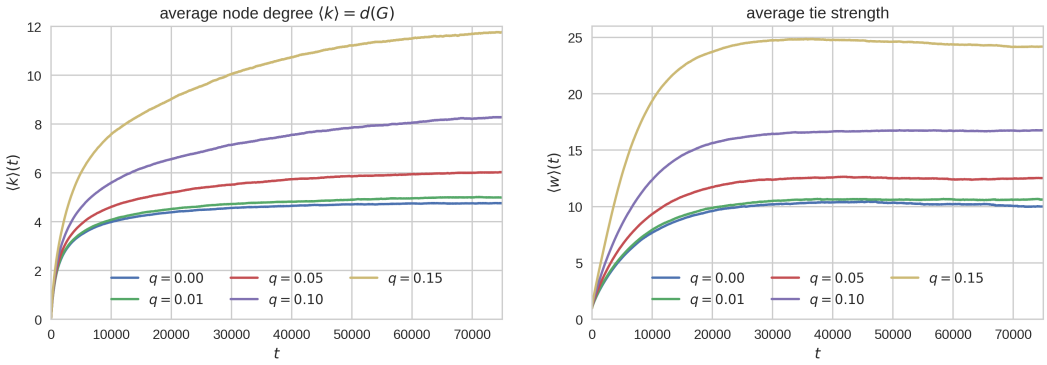
Figure 4.3.: The percentage of reinforced ties $r(t) = {}^{E_r(t)}/_{(E_r(t)+E_c(t))}$ for different levels of peer influence as a function of time, where $E_c(t)$ and $E_r(t)$ are the number of created ties and the number of reinforced ties in iteration t, respectively. (a) shows the ratio over all 75,000 iterations and (b) highlights the behavior in the beginning. Both functions were smoothed using the rolling mean method to improve the quality of the plots.

The evolution of the average local clustering coefficient does depend on the node deletion probability p_d, since low-activity nodes, which are not removed fast enough, introduce additional weak ties in the network [LSK15]. However, as clearly evident in figure 4.1, the level of peer influence does influence the clustering, and therefore the community structures of the network, as well. The more likely an activation due to peer influence gets, the smaller the stationary value for C becomes. Figure 4.2b highlights this effect well. This observation can possibly be explained in a similar way as the effect caused by the deletion probability.

However, in the case of occurring peer influence, not the decelerated removal of nodes is responsible for the smaller stationary value of C, but the overall increase in the number of contacts between nodes. The peer influence mechanism increases the activity in the network, especially in already formed communities (see section 4.2), since active nodes motivate their neighbors to become active as well. The probability for the formation of a new tie is inverse proportional to the size of a node's egocentric network. Therefore, an active node, which is already fully integrated in its community, will reinforce one of its existing ties, or at least close a triangle, with high proba-

bility. However, given enough tries, such a node will eventually introduce new weak ties following the focal closure mechanism as well. Therefore, the opportunities for the introduction of random links by active nodes increases, which leads to a smaller average local clustering in general.

Two additional measures of the integrated network, the average node degree and the average tie strength, were tracked over time for different magnitudes of peer influence as well. Figure 4.4 depicts the graphs for these two network properties as function of time. Both measures show a similar general behavior. The average degree and the average tie strength are not independent of the maximum peer influence probability in the network and do converge after the integrated network reaches its equilibrium. The stationary values of both do increase with increasing values for q by about the same order of magnitude, which is reasonable since both measures are related to each other. The tie strength of a node can be seen as its weighted degree.



(a) (b)

Figure 4.4.: Plots of (a) the average node degree $d(G_T)$ and (b) the average tie strength $\langle w \rangle$ in the network as a function of time for different maximum peer influence probabilities $q = 0, 0.01, 0.05, 0.1, 0.15$.

However, the time it takes for them to converge differs. The average degree takes longer to reach its stationary value. This can possibly be explained by the small probability for the creation of new ties after the egocentric networks have gained a certain size. Every new neighbor reduces the probability for the creation of a new tie in the future significantly. The average

tie strength does not suffer from this problem, due to the fast development of strong ties and the decreasing probability for the introduction of weak ties.

The direct effect of the peer influence mechanism on the average degree and average weight can probably be explained, similar to the effect on average local clustering coefficient, by the additional activity in the temporal network. Nodes are getting more opportunities to add additional neighbors and to strengthen their ties until they get removed. Note that the slightly different convergence behavior of the average tie strength for the high peer influence probability $q = 0.15$ cannot be explained fully at this point. It may be related to the by comparison significantly slower convergence of the average degree, but it is ultimately left open for possible future studies.

4.2. Network Activity

One obvious implication of the proposed peer influence mechanism is an increase in the activity in the network. Nodes can become active by them self not only due to their intrinsic activity potential, but also by the influence of their peers. The number of nodes that become active in iteration t entirely by them self is denoted as $E_a(t)$, and the number that becomes active motivated by others is denoted as $E_p(t)$. Therefore, the total number of activations per iteration is $E(t) = E_a(t) + E_p(t)$.

The number of peer influenced activations in a networks with $q = 0$ is trivially $E_p(t) = 0$, and the number of self-activations can be approximated by $E_a(t) \approx n\mathbb{E}[a_i]$. For example, the approximated number of activations in a network with no peer influence effects and the prior specified parameters ($\gamma = 2.7$, $\varepsilon = 0.001$, and $n = 5,000$) is $E_a(t) \approx 12.05$, which matches the observed figures. In fact, this number is roughly the same for all levels of peer influence, since the process of node self-activation due to the intrinsic activity potential is independent of the peer influence mechanism.

The total number of activations, and therefore the gain in activity due to

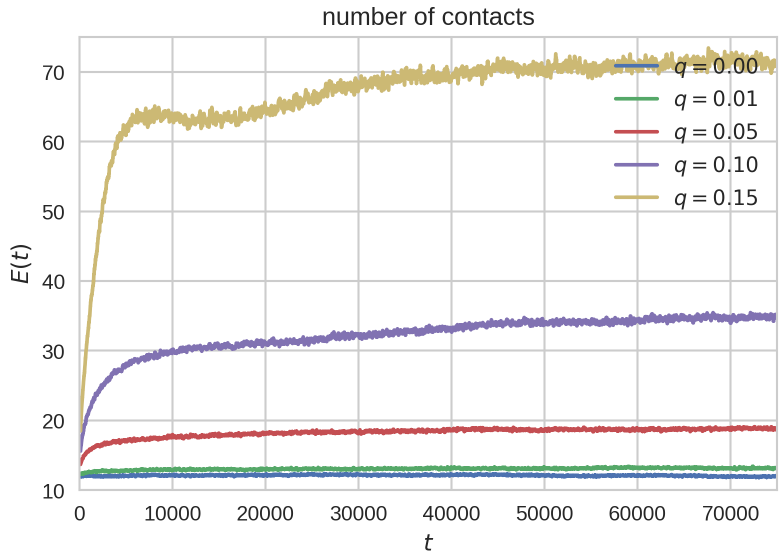
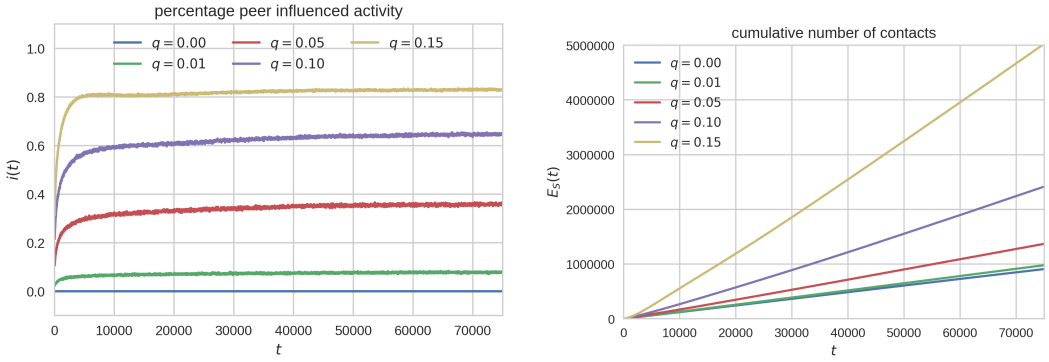


Figure 4.5.: Time-dependent number of activations $E(t)$ in the network for different levels of peer influence q. The graphs were smoothed using the rolling mean method to improve the visualization.

the peer influence, is depicted in figure 4.5. It shows that $E(t)$ reaches its stationary value quickly for smaller levels of peer influence. However, the development of the total number of contacts per iteration is more complex for networks with a higher degree of peer influence. The first phase can be described as a rapid increase in the number of activations, which stops after approximately 8,000 iterations. After that, the development of $E(t)$ starts to relax and slow down. The activity level even starts to decreases slightly in the network with a maximum peer influence probability of $q = 0.15$. Finally, the numbers slowly converge to their stationary values, which are reached after about 40,000 iterations. This is also supported by the evolution of the fraction of the peer influenced contacts per iterations $i(t)$, which shows the same general behavior (cf. figure 4.6).

The observed ramifications on the total activity in the network is possibly related to the development of the community structures and its effects on the peer influence mechanism. The stagnation (or even reduction) of the activity begins after a maximum value for the average local clustering

coefficient was reached (cf. figure 4.1), which is caused by the introduction of weak ties. These new ties are possibly responsible for the merging of communities in the network, which in turn reduces the effects of the peer influence mechanism. Inactive nodes in these newly merged communities may have a greater impact on the weighted fraction of active nodes that is required for a significant influence on nodes. This is could especially be true for weakly integrated nodes. However, a more detailed analysis has to be performed in the future to determine and verify the true effects, which are responsible for the observed behavior in the network activity for larger magnitudes of peer influence.



(a) (b)

Figure 4.6.: Different views on the ramifications of the peer influence mechanism on the activity in the network. (a) depicts the evolution of the fraction of peer influenced activity $i(t) = {}^{E_p(t)}/_{E(t)}$ over time for different levels of peer influence and (b) shows the aggregated number of contacts $E_S(t)$ over all 75,000 iterations. The $i(t)$ graphs were smoothed using the rolling mean method to improve the quality of the plot.

The effects of the peer influence mechanism on the activity within the time-varying network can be observed in other ways as well. For instance, figure 4.6b depicts the cumulative number of contacts (i.e., $E_S(t) = \sum_{i \leq t} E(i)$), which highlights the quantity of additional activity that was caused by different degrees of peer influence over time. Another example for the impact of the newly introduced mechanism is the change in the distributions of the number activations in a given time interval. There is a shift in the bulk of the distributions observable, which indicates an increased probability for larger number of activations of a node.

4.3. Inter-event Time Distributions

As already mentioned in the motivational section of this thesis (section 1.3) and while discussing different ways to model user activity (section 2.5), human activity patterns can be fairly complex. They can be described as bursts followed by longer phases of inactivity and are usually characterized by the inter-event time distribution $\varphi(\tau)$, which should reflect these requirements.

The inter-event times are defined, in the context of our model, as the time between two consecutive activations of a node. The type of activation (i.e., activation due to peer influence, activity potential, or contact by another active node) is not taken into account for the by us performed analysis. Each node v_i in the network has its own inter-event time distribution φ_i, which depends on the node's activity potential and on the influence of its peers.

However, to get a better overview on how the peer influence mechanism changes the dynamics in the network in general, the union of the inter-event time distributions of all nodes is examined. The sequence of inter-event times is determined for every node, that was active between the beginning and the end of the simulation, separately and then combined into one distribution. This distribution can be seen as a mixture of distributions [Sei11] of nodes that were at some time present in the network, i.e.,

$$\varphi(\tau) = \sum_i \pi_i \varphi_i(\tau), \tag{4.1}$$

where π_i denotes the mixing weights, which must be selected such that $\sum_i \pi_i = 1$ holds. Therefore, the mixing weights determine the relative importance of the individual distributions. For our experiments every distribution is considered equally important and is assigned the same weight, such that the summation constraint is fulfilled.

The burstiness of human behavior is difficult to describe and even more difficult to quantify. It is usually determined by certain moments of the

inter-event time distribution. For instance, the coefficient of variation [ML16] is defined as the ratio between the standard deviation σ and the mean μ of the inter-event time distribution, i.e., $c_v = {}^{\sigma}/_{\mu}$. It takes the value 0 when the events are occurring at a fixed non-random rate. The coefficient of variation is 1 for exponentially distributed inter-event times, which is the case for events that are generated by a Poisson process, and c_v can get arbitrarily large for long-tailed distributions, such as power laws.

A normalized variant of the coefficient of variation was proposed by Goh and Barabási [GB08], which is called burstiness parameter and is defined as

$$B = \frac{c_v - 1}{c_v + 1} = \frac{\sigma - \mu}{\sigma + \mu}, \tag{4.2}$$

and takes values in the range $B \in [-1, 1]$. The burstiness parameter is -1 for regularly occurring events, 0 for inter-event times that originated from a Poisson process, and 1 for a distribution that was derived from an extremely bursty sequence of events.

A nice property of this burstiness measure is that it can also be applied to mixture distributions that contain the inter-event times of, for example, people with different intrinsic activity levels. For instance, the inter-event times of power-users that use a system heavily every day can be combined with those of users that use it only irregularly and the burstiness parameter is able to capture the level of burstiness, that is present in the usage of the system, regardless. Therefore, it is also applicable for the inter-event time distributions derived from the simulations of the peer influence model.

The burstiness parameter for generated networks with no peer influence is approximately $B = 0.19$. The introduction of peer influence increases this value, however, a drastic increase can only be observed for larger values of the maximum peer influence probability q. Table 4.2 contains the mean, standard derivation and burstiness parameter of the inter-event time distribution for a maximum peer influence probability in the range $0 \leq q \leq 0.15$.

q	0.00	0.01	0.025	0.05	0.075	0.10	0.15
μ	198.71	184.59	164.26	132.80	102.43	76.28	37.23
σ	291.32	270.49	241.40	197.38	155.09	118.04	61.22
B	0.1890	0.1888	0.1902	0.1956	0.2045	0.2149	0.2437

Table 4.2.: Mean value, standard deviation and the resulting burstiness parameter of the inter-event time distribution for different levels of peer influence.

This change in B indicates that the peer influence mechanism affects the burstiness of the node activations. In a network with no peer influence are two consecutive self-activations independent of each other. Therefore, the activations happen at a certain rate that is proportional to the activity potential of a node, which leads to exponentially distributed inter-event times [MSP16]. This should results in a burstiness parameter that is close to $B = 0$. However, this is not the case for the inter-event times that were generated with the proposed model, even though peer influence was disabled (cf. table 4.2).

The observed value for B can possibly be explained by the memory effects, which allow for reoccurring interactions within groups of nodes and by the inclusion of passive activations due to other active nodes in the calculation of the inter-event times. A node with a higher intrinsic activity potential will select a node from its local group with high probability. Hence, this more active node activates less active nodes regularly, which can lead to a more bursty looking activity pattern of the other nodes, and explains (at least partially) the burstiness value of $B = 0.19$. The peer influence mechanism of this model should amplify this effect even more, since it increases the activity within communities. Furthermore, the activation of nodes possibly triggers cascading activations within the communities, which makes bursts more likely as well.

One way to examine how the peer influence mechanism changes the inter-node-activation times in the network is to inspect their distribution visually. Figure 4.7 depicts the inter-event distributions for a variety of peer influence levels. The plot shows the distribution on a log-log graph, where both axis are scaled logarithmically. This highlights the possibilities for longer inter-

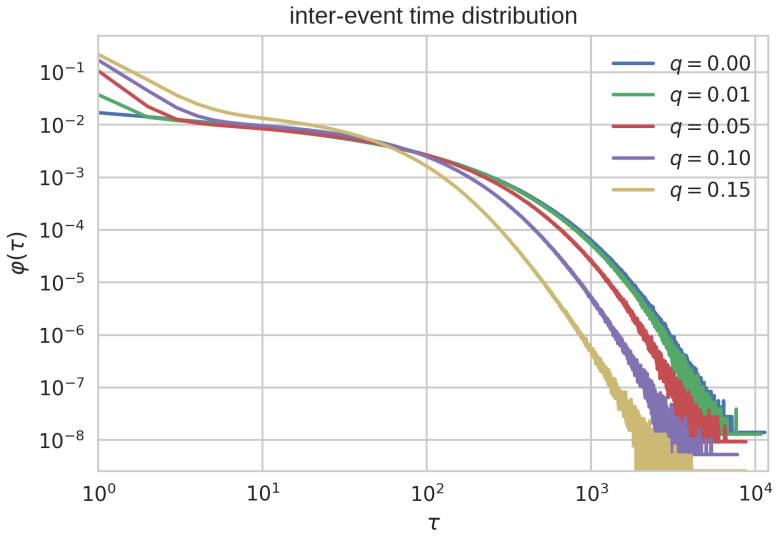


Figure 4.7.: Log-log Plot of the inter-event time distribution for different maximum peer influence probabilities $q = 0$, 0.01, 0.05, 0.1, 0.15.

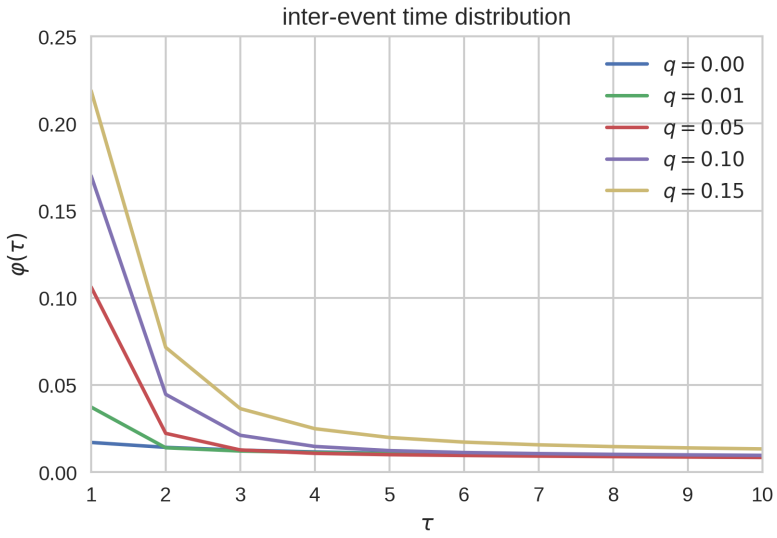


Figure 4.8.: Plot of the bulk of the inter-event time distribution for small inter-event times in the range $1 \leq \tau \leq 10$ and different levels of peer influence.

event times between two consecutive activations well. The plot also shows that the level of peer influence in the network changes the shape of the distribution. Smaller inter-event times become more likely for higher levels of peer influence, which can also be seen in the change of the distributions' mean values (cf. table 4.2).

The average time between activations decreases from about 200 in a network with no peer influence at all, to 38 in a network with $q = 0.15$. The changes in the distributions are even more noticeable for inter-event times less than ten time steps. Figure 4.8 depicts the distributions in this interval of τ. The probability for two successive activations increases drastically from less than 2% to almost 22% over the range of possible values for q. This development of the probabilities possibly explains the increase of the burstiness of the distributions, since a larger number of events within a small time frame becomes more probable due to the peer influence mechanism.

However, the tail of the inter-event time distribution changes for higher levels of peer influence as well. Large inter-event times become more and more unlikely for larger values of q, and the length of the tail decreases as well. The values for the standard derivation of the distributions reflect this behavior as well (cf. table 4.2). The standard derivation of the inter-event times in the network with high peer influence effects is only about one fifth of the standard derivation in the network without peer effects. This, of course, prevents longer intervals of inactivity to a certain extend, which is the second crucial requirement for realistic activity patterns.

4.4. Degree and Tie Strength Distributions

While section 4.1 examines the average degree and tie strength as functions of time, are the distributions of these two properties the topic of discussion in this section. Both distributions are obtained from the extended version of the integrated network, which contains all nodes that were present in the last iteration of the simulation. The degrees and tie strengths of nodes that were deleted in previous iterations are not part of the results.

The ramifications of the memory process and the closure mechanisms on the distributions were already a subject in the original work by Laurent, Saramäki, and Karsai [LSK15]. They showed that both the degrees and the tie strengths are heterogeneously distributed. Furthermore, the shape of the degree distribution is only slightly affected by the triadic closure probability p_Δ and the tie reinforcement increment δ. However, while not being affected by p_Δ, the tie strength distribution does depend on the memory process of the model. For instance, larger values for δ increase the length of the tail.

Nevertheless, a more important issue are the effects on the degree and tie strength distributions caused by the proposed peer influence mechanism. First and most important, the peer influence mechanism does not effect the heterogeneous nature of both distributions, which is an often observed property in many real world networks [AB02; KPV14]. The resulting possibilities for nodes with higher degree indicate the presence of hubs, which may play, depending on the context, various important roles in networks. Figure 4.9 depicts the right-skewed degree distributions for networks with different levels of peer influence. It shows that not only the average degree in the network is directly affected by the peer influence process, but the distribution itself as well. Nodes with small degrees become less probable, while higher degree nodes become more frequent with the increasing magnitude of peer influence. Furthermore, the tail of the distribution does not seem to be significantly affected by the process. The change in the shape of the distribution is in line with the already observed increasing average degree and can be explained by the same argument as well. The boost in node activations caused by the influence of the neighbors leads to more opportunities for the formation of new ties and consequently to larger egocentric networks.

The memory process is primarily responsible for the heterogeneous tie strength distribution and the introduction of the peer influence mechanism does, in fact, not change the distribution in a significant way. Figure 4.10 shows the distributions for different values for the maximum peer influence probability. One might expect that the increased activity within the temporal network would extend the tail of the distribution in a significant way. However, this is not the case, and a possible explanation for this effect is related to the increased average size of the egocentric networks. More

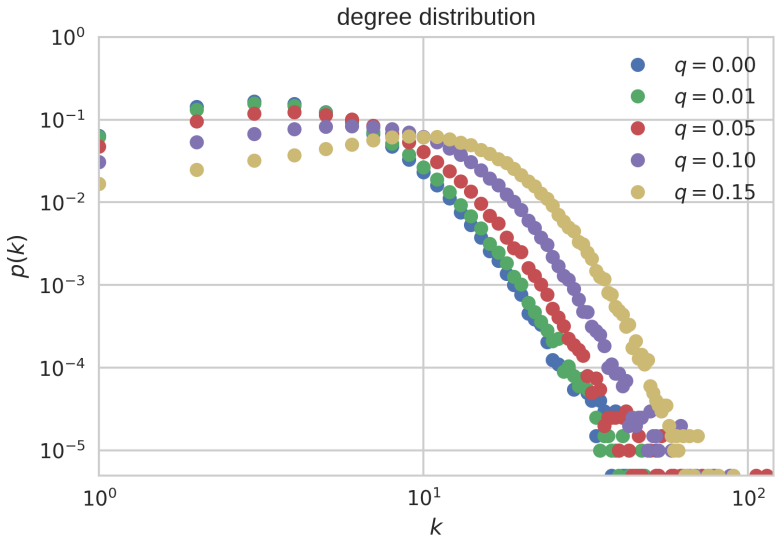


Figure 4.9.: Depiction of a degree distribution of the integrated network for different maximum peer influence probabilities $q = 0$, 0.01, 0.05, 0.1, 0.15.

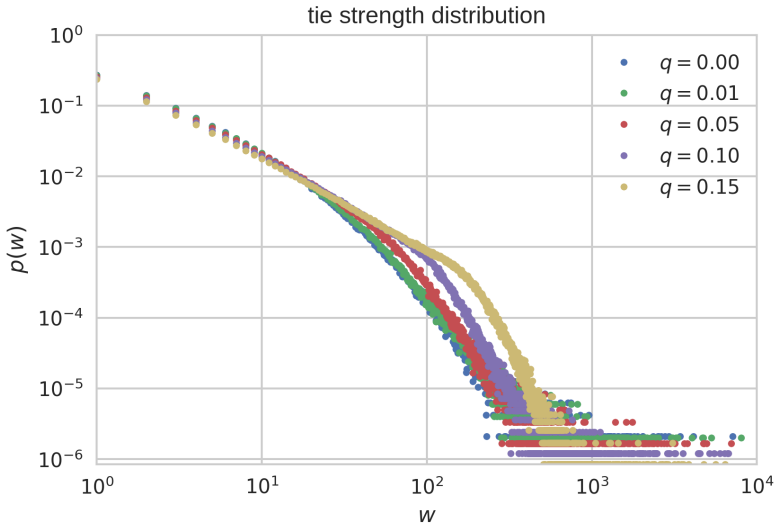


Figure 4.10.: Log-log plot of the distribution of the tie strengths (i.e., the link weights) in the integrated network for different degrees of peer influence.

neighbors imply a larger number of strong ties, which require frequent contacts to develop. Therefore, the additional activations caused by the peer influence mechanism are spent on the development of the additional strong ties and not on reinforcing the existing ones, which would presumably enlarge the tail. This would also explain the more frequent weights in the lower double-digit region, which is a consequence of the on average increased tie strength for networks with peer effects.

4.5. Tie Strength Rescaling Scenarios

Nodes are only able to influence their peers if they were active in the previous iteration. The magnitude of influence does depend on the strength of the ties between nodes. This is based on the idea that the activity of a close friend should be more effective in motivating a person to become active as well, than the activities of acquaintances.

The weights (i.e., the strength of the ties) are usually rescaled in such a way that larger weights become even larger to amplify the importance of close neighbors. This can be archived by using the softmax function, which also allows to control the weight rescaling process even more using the inverse temperature parameter β. This free parameter can vary in time as well to allow for different rescaling characteristics depending on the progress of the simulation. Furthermore, it also allows for various peer influence scenarios, in which nodes get influenced to different extents, depending on the situation.

This section highlights the ramification on the peer influence mechanism for the following three different weight rescaling strategies. Note that the results for the scenarios are obtained from synthetic networks with a fixed maximum peer influence probability of $q = 0.05$. All other model parameter are set to the same values that were already used in the other experiments.

$\beta(t) = \langle w \rangle(t)^{-1}$
The temperature for the softmax rescaling is set to the average tie strength in the snapshot of integrated network at time t. This scenario is used in all other previously discussed experiments in this chapter. Section 4.1 shows that the average tie strength does increase over time until it reaches its stationary value. Therefore, this effectively introduces a bias towards stronger ties in the early stages of the simulation that decays over time. The activity of close neighbors influences nodes to a greater extent in the beginning, which is possibly beneficial for the formation of the community structures.

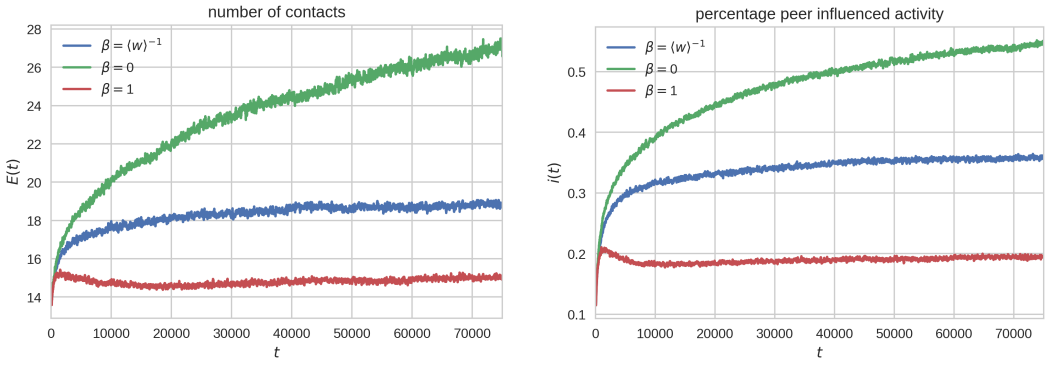
$\beta(t) = 1$
In this scenario the inverse temperature parameter is fixed to the constant value 1 for each iteration. Therefore, the bias towards strong ties is overall more prominent compared to the previous scenario, and does not change during the entire simulation. This may lead to situations in which only a very small number of neighbors do actually influence a node once they become active, since weaker ties become so insignificant that they do not exercise considerable influence anymore. Hence, only one of the highly influential neighbors is possibly enough to influence a node at the maximum possible level.

$\beta(t) = 0$
In the last scenario β is fixed to 0, which basically disables the softmax weight rescaling entirely and assigns each tie the exact same weight. The strength of the tie between any ordered pair of nodes (v_i, v_j) in the network is rescaled to $w'_{i,j} = 1/|N(v_i)|$. This allows neighbors, which share a weak tie with a node, to exercise the same degree of peer influence as strong tied neighbors. Therefore, an on average larger number of active neighbors may be required to reach the maximum possible magnitude of peer influence for a node.

It seems reasonable to examine the change in the network activity for the three different rescaling scenarios, since most of the explanations for the effects observed in the other experiments are related to it. Figure 4.11 shows

the number of interactions between nodes and the fraction of peer influenced interactions as functions of time for the three different scenarios.

Compared to the rescaling scenario that adapts to the average tie strength in the network, an overall decrease in the number of interactions is observable in the $\beta = 1$ scenario. This behavior can be attributed to an excessive bias towards strong ties. Nodes only become active by them self due to their intrinsic activity potential or possibly if one or two of their strong tied neighbors were active in the round before. This probably restricts cascading activations within groups as well, since weakly tied nodes are not able to propagate activations properly. The number of contacts and the fraction of those activations that were caused by peer influence increase rapidly in the first few hundred iterations in the $\beta = 1$ scenario, and decreases slightly after they reached a maximum value. This small decrease is not observable in the other two scenarios and can possibly be attributed to the weight heterogeneities that start to emerge at this point in time in the network, which constrains the peer influence mechanism in this scenario even more.



(a) (b)

Figure 4.11.: Depiction of the influence on the activity over time for different values of β, the inverse temperature for softmax rescaling. (a) shows the number of interactions for each iteration $E(t)$ and (b) depicts the fraction of activations that were caused by the peer influence mechanism $i(t)$. The graphs were smoothed using the rolling mean method to improve the quality of the visualization.

The scenario that disables the softmax weight rescaling using $\beta = 0$, how-

ever, does increase the number of contacts in the network in each iteration drastically. The expected larger number of active neighbors, which is required to motivate a node seems to be archived fairly often. This can possibly be attributed to the relatively low critical peer influence threshold $\theta = 0.10$, which should probably be adjusted for this rescaling scenario. Furthermore, the network activity does not convergence in this scenario. One possible explanation for the continuous increase in the number of contacts could be attributed to self-enhancing cascading activation effects going through the network. However, this requires additional investigation and is a possible topic for future studies.

5. Conclusion

In the final chapter of this thesis we not only present a review on the achieved results, but also include a short summary of the preceding chapters. Furthermore, we discuss possible limitations of the proposed peer influence model and the methods that were used for its evaluation. Finally, we conclude the thesis with suggestions for possible future studies. These suggestion include improvements for the model itself, and potential interesting experiments on synthetic and real-world networks.

5.1. Recap

We started our work with a brief motivational section, which sequentially led to the incentive for the proposal of an extension to the activity-driven time-varying network framework [Per+12a], which incorporates peer influence effects. The key idea behind the proposal is that people do not solely perform actions based on their intrinsic motivation, but also because of the influence of their peers. This idea stands in contrast to the activity-driven network model, in which nodes can become active only by them self based on their activity potential. Therefore, it provides an ideal foundation to implement a model with a peer influence mechanism on top of it.

Moreover, preliminaries on the basic topics of graph theory, various types of networks and related generative models, and time-varying networks were discussed. These are essential for the subsequent definition and evaluation of the proposed model. A detailed review on the activity-driven framework itself, the properties of the temporal networks it generates, and on the latest

related literature that either adapts or extents it, was performed as well. Furthermore, an overview on sociological peer influence studies and network models, which utilize different types of peer influence mechanisms, is also an element of our work, to highlight the rationale behind the proposed model even more.

The presence of peer influence effects requires social structures, such as tie strength heterogeneities and community structures, to exist in the network, and while the basic activity-driven framework is capable of generating rich topological structures, it is not able to provide these preconditions. To overcome this issue, an extension of the framework by Laurent, Saramäki, and Karsai [LSK15] was utilized as actual foundation for the proposed model. It introduces memory effects and closure processes to allow for the formation of communities and weight-topological correlations in the networks. The mechanisms and properties of this model were discussed in detail as well, followed by the comprehensive definition of the peer influence extension, which was heavily influenced by the work of Walk et al. [Wal+16].

The actual specification of the peer influence mechanism resolves various important issues, such as,

- how to quantify the influence of neighbors in the egocentric network of a node,
- how to determine the relative influence of individual neighbors based on the tie strengths, or
- what additional components must be included into the model to store the for the peer influence mechanism relevant information.

After the model[1] was fully specified, we evaluated it on synthetic networks. This included an examination of the effects on the topological structures of networks, which showed that the peer influence mechanism does accelerate the formation of the community structures and influences their strength. Additional investigations on the increased activity within the networks

[1] An open-source Python implementation of the proposed model can be found at `https://github.com/dumfug/PIModel`.

and its implications was conducted as well. They revealed not only the complex behavior of the total number of activations per iteration over time for larger magnitudes of peer influence, but also the impact on the activation patterns of individual nodes. The distribution of time intervals between two consecutive activations changed in a way that allows for increased burstiness, which is a property that is observable in human activity patterns. Finally, we performed an evaluation of three different scenarios that vary the relative influence of the tie strengths between nodes and their effects on the overall network activity. This revealed interesting consequences, especially in the case in which the influence of nodes was set to be independent of the actual tie strengths, which resulted in diverging levels of activity in the network.

5.2. Limitations

One shortcoming, which affected all performed experiments, is the relatively small size of the generated networks. The number of nodes in many real-world networks can be multiple orders of magnitude larger. However, to simulate temporal networks of this size efficiently, improvements in implementation of the model must be made, which minimize the additional overhead of the peer influence mechanism. Nevertheless, this would possibly help to explain some observed effects better. For example, the temporarily decrease in the network activity for higher levels of peer influence, which could possibly be explained by the merging of communities in the network, is probably more prominent in larger networks.

Simulations of the model with an increased number of nodes would also allow to compare the properties of synthetically generated networks with those of real-world data sets more easily, and therefore help to verify the validity of the proposed model. This would not only be highly interesting for topological features like clustering, but also for the inter-event time distributions. The observed changes in the distributions do not fully capture the discussed characteristics of human activity patterns, which are bursts followed by longer intervals of inactivity. While the probability for short

breaks between activations increased with additional peer influence, which matches the first requirement, declined the length of the tail. This, again, restricts the possibilities for longer intervals between bursts.

A possible explanation for this behavior is an imbalance of the peer influence mechanism. On one hand, a level of peer influence that is set too low, does not affect the activity in a significant way. On the other hand, the process becomes too dominant for higher levels, and leads to bursts within communities that are repeated over and over again. A potential solution would be to restrict the peer influence process in a way that nodes cannot be easily influenced multiple times within a short period of time (i.e., a cool-down time for the effect). However, this requires additional investigations on how these cascading peer influence effects actually work.

5.3. Future Work

The proposed peer influence model provides various opportunities for possible extensions and experiments. The following examples should give some thought-provoking impulses.

Dynamic Processes A common use case of time-varying networks is their ability to study dynamic systems [Hol15] in great detail. Two common types of these dynamic processes are epidemic models and opinion spreading models. The first one deals with the spreading of diseases. It typically consists of a set of states (e.g., susceptible, infected, recovered,...), which nodes can adopt, and transition rules between them. Such models can, for example, be used to study disease containment strategies. Opinion spreading models, on the other hand, deal with the spreading or adaption of certain concepts (e.g., opinions, ideas, products,...), which usually follow different spreading mechanisms. The by us proposed model could possibly be used as a framework to study such dynamic processes with respect to peer influence effects.

Negative Peer Influence Peer influence is always considered a positive force in the current version of the proposed model, which affects the activity level of nodes in a positive way. This type of peer influence could, for example, be observable in the context of open-source software projects, in which the activity of maintainers possibly increases the motivation of others to contribute as well. However, the activity of people in some social scenarios may lead to the opposite effect. For example, a large number of malicious contributions (e.g., due to trolls) in online communities may induce a decline in the overall interest in the website. This idea was already pursued in the context of activity dynamics by Koncar et al. [Kon+17]. Therefore, it would also be feasible to introduce negative peer effects in our model, which reduce the activity of nodes, and study its implications.

Peer Influence Mechanics The way the peer influence mechanism was implemented in the proposed model was based on the work of Walk et al. [Wal+16], and on the intuitive understanding of the concept. However, no claim to completeness and correctness for the presented model is made. There is always potential for improvement by the inclusion of new ideas, or the removal of existing elements of the model. For example, the complex softmax weight rescaling mechanism may be replaced with a simple weighted fraction. This would remove one free parameter and possibly improve the runtime performance. Another potential extension would be the introduction of a critical peer influence threshold, which depends on the size of the egocentric network of a node. This would allow for additional peer influence scenarios, like a reduced influenceability of hubs.

Appendix

Appendix A.

Power-law Probability Distribution

In this thesis we often refer to power-law probability distributions, especially in the context of activity-driven models. However, for the sake of simplicity the exact formula of the distribution is never explicitly stated. This appendix contains the derivation for the exact density, distribution function, and the mean value of the probability distribution. Additionally, it contains an example and a short description of the inverse transform sampling method, which is used in the implementation of the model. However, all this requires a more formal definition of the distribution, so let X be a continuous random variable with the probability density function $f(x) \sim x^{-\gamma}$, taking values in the range $X \in [\epsilon, 1]$, with $\gamma > 0$ and $0 < \varepsilon \leq 1$.

A.1. Probability Density Function

The probability density function (PDF) of the power-law distributed random variable X, which is defined above, is stated in a more detailed form in equation (A.1).

$$f(x) = \begin{cases} cx^{-\gamma} & \varepsilon \leq x \leq 1 \\ 0 & \text{otherwise.} \end{cases} \tag{A.1}$$

The factor c denotes a normalizing constant, which ensures that the function fulfills the properties of a probability density function (i.e., $\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x = 1$). To calculate the normalizing constant equation (A.1) must be solved for c, i.e.,

$$\begin{aligned} &\int_{-\infty}^{\infty} cx^{-\gamma}\,\mathrm{d}x = c\int_{\varepsilon}^{1} x^{-\gamma}\,\mathrm{d}x = c\,\frac{x^{1-\gamma}}{1-\gamma}\bigg|_{\varepsilon}^{1} = c\,\frac{1-\varepsilon^{1-\gamma}}{1-\gamma} = 1 \\ &\Leftrightarrow\ c = \frac{1-\gamma}{1-\varepsilon^{1-\gamma}}. \end{aligned} \tag{A.2}$$

A.2. Cumulative Distribution Function

The cumulative distribution function (CDF) is used to calculate the probability that a random variable with a probability distribution f takes a value less than x, i.e.,

$$\mathbb{P}[X \leq x] = F(x) = \int_{-\infty}^{x} f(t)\,\mathrm{d}t. \tag{A.3}$$

The derivation of the cumulative distribution function for the power-law distribution defined above is done in equation (A.4). Note that this result only holds for values of $\gamma \neq 1$.

$$F(x) = \int_{-\infty}^{x} ct^{-\gamma}\,\mathrm{d}t = c\,\frac{t^{1-\gamma}}{1-\gamma}\bigg|_{\varepsilon}^{x} = \frac{c}{1-\gamma}\left(x^{1-\gamma} - \varepsilon^{1-\gamma}\right) \tag{A.4}$$

Due to the range of possible values that can be taken by the probability density with positive probability, the CDF can be rewritten as a piece-wise function, i.e.,

$$F(x) = \begin{cases} 0 & x < \varepsilon \\ \frac{c}{1-\gamma}\left(x^{1-\gamma} - \varepsilon^{1-\gamma}\right) & \varepsilon \leq x < 1 \\ 1 & x \geq 1 \end{cases} \tag{A.5}$$

A.3. Expected Value

The expected value of the random variable X, which was defined in the beginning, is derived as follows,

$$\begin{aligned} \mathbb{E}[X] &= \int_{-\infty}^{\infty} x f(x)\,\mathrm{d}x = \int_{\varepsilon}^{1} x f(x)\,\mathrm{d}x = c\int_{\varepsilon}^{1} x x^{-\gamma}\,\mathrm{d}x \\ &= c\int_{\varepsilon}^{1} x^{-\gamma+1}\,\mathrm{d}x = c\frac{x^{2-\gamma}}{2-\gamma}\Big|_{\varepsilon}^{1} = \frac{c}{2-\gamma}\left(1-\varepsilon^{2-\gamma}\right) \end{aligned} \tag{A.6}$$

A.4. Example: $\gamma = 2.5$ and $\varepsilon = 10^{-2}$

For this example, the exponent parameter of the distribution is set to $\gamma = 2.5$ and the lower bound is fixed to $\varepsilon = 0.01$. These parameter require a normalizing constant of $c = 1/666$. Therefore, the density is $f(x) = x^{-2.5}/666$ and has an expected value of 0.027. Figure A.1 and figure A.2 show the plots for the probability distribution function and the cumulative distribution function for this example, respectively.

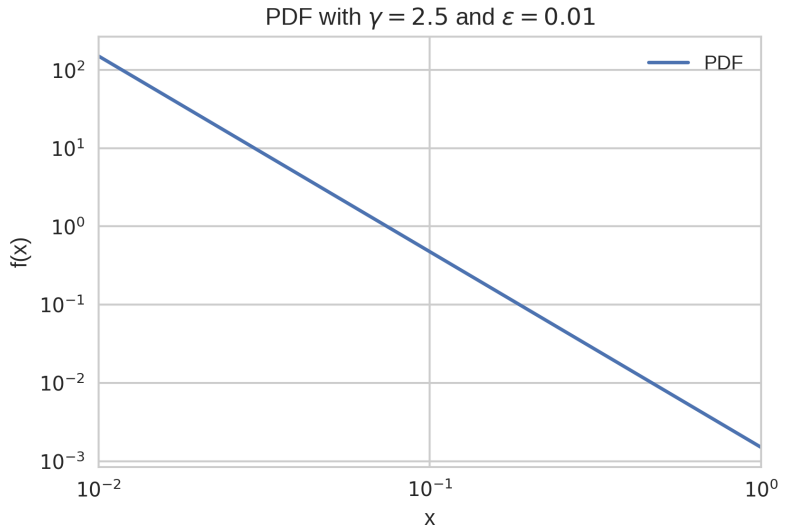


Figure A.1.: Log-log plot of the probability density function $f(x) = \frac{x^{-2.5}}{666}$ taking values in the range $[0.01, 1]$.

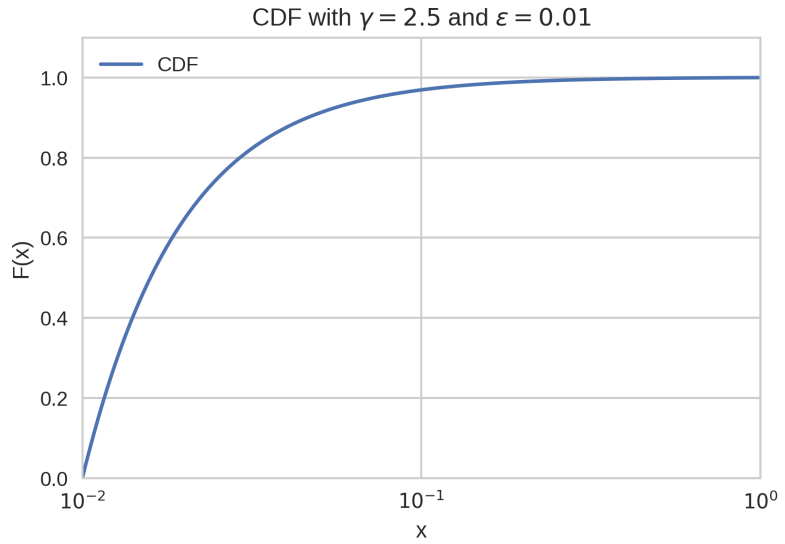


Figure A.2.: Plot of the cumulative distribution function of the power-law distribution described by the PDF $f(x) = \frac{x^{-2.5}}{666}$ taking values in the range $[0.01, 1]$.

A.5. Inverse Transform Sampling

To generate samples from the prior defined power-law distribution, the inverse transform sampling method is used in the context of this thesis. This algorithm is based on the inversion principle [Dev86]. It states that if U is an uniform random variable on the unit interval (i.e., $U \sim \mathcal{U}(0,1)$), then the random variable $Y = F^{-1}(U)$ has the probability distribution function F, where F^{-1} is the inverse distribution function. The proof of this theorem is very short (cf. equation (A.7)) and exploits the fact that $\mathbb{P}[U \leq x] = x$ for a random variable $U \sim \mathcal{U}(0,1)$.

$$\mathbb{P}[Y \leq x] = \mathbb{P}[F^{-1}(U) \leq x] = \mathbb{P}[U \leq F(x)] = F(x) \tag{A.7}$$

The actual algorithm is very short and simple as well. To draw a sample from a distribution with a cumulative distribution function F, execute the following two steps:

1. Draw a number r uniformly at random from the unit interval $[0,1]$.
2. Calculate $F^{-1}(r)$ to obtain the sample.

However, the inverse transform sampling method requires the inverse of the cumulative distribution function F^{-1}. This can, for example, be done by solving $F(x) = y$ for x. The inverse CDF for the power-law distribution is

$$F^{-1}(x) = \left(\frac{x(1-\gamma)}{c} + \varepsilon^{1-\gamma}\right)^{1/(1-\gamma)}. \tag{A.8}$$

See equation (A.9) for the derivation. Figure A.3 depicts an approximation of a power-law distribution that was generated using the inverse transform sampling algorithm.

$$\begin{aligned}
\frac{c}{1-\gamma}\left(x^{1-\gamma} - \varepsilon^{1-\gamma}\right) &= y \\
x^{1-\gamma} - \varepsilon^{1-\gamma} &= \frac{y(1-\gamma)}{c} \\
x^{1-\gamma} &= \frac{y(1-\gamma)}{c} + \varepsilon^{1-\gamma} \\
x &= \left(\frac{y(1-\gamma)}{c} + \varepsilon^{1-\gamma}\right)^{1/(1-\gamma)}
\end{aligned} \tag{A.9}$$

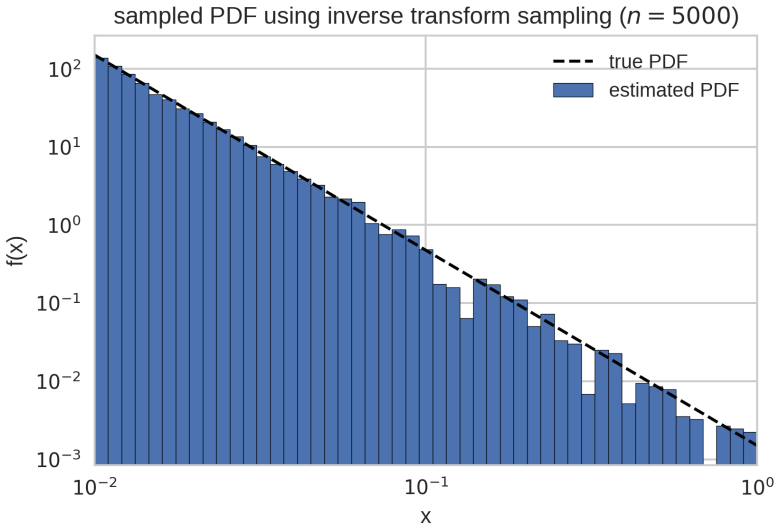


Figure A.3.: The estimated probability density function of $f(x) = \frac{x^{-2.5}}{666}$, taking values in the range $[0.01, 1]$ (i.e., the PDF from the example in appendix A.4). The approximation is archived using $n = 5,000$ samples, obtained using the inverse transform sampling method.

Bibliography

[AB02] Réka Albert and Albert-László Barabási. "Statistical mechanics of complex networks." In: *Rev. Mod. Phys.* 74 (1 Jan. 2002), pp. 47–97. DOI: 10.1103/RevModPhys.74.47 (cit. on pp. 20, 21, 23, 70).

[Ale+17] Laura Alessandretti et al. "The interplay between activity and attractiveness on random walks in time-varying networks." In: (2017). arXiv: 1701.06449 (cit. on p. 33).

[AMS09] Sinan Aral, Lev Muchnik, and Arun Sundararajan. "Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks." In: *Proceedings of the National Academy of Sciences* 106.51 (2009), pp. 21544–21549. DOI: 10.1073/pnas.0908800106 (cit. on p. 37).

[AO11] Daron Acemoglu and Asuman Ozdaglar. "Opinion Dynamics and Learning in Social Networks." In: *Dynamic Games and Applications* 1.1 (2011), pp. 3–49. DOI: 10.1007/s13235-010-0004-1 (cit. on p. 34).

[BA99] Albert-László Barabási and Réka Albert. "Emergence of Scaling in Random Networks." In: *Science* 286.5439 (1999), pp. 509–512. DOI: 10.1126/science.286.5439.509 (cit. on p. 22).

[Bag08] Ganesh Bagler. "Analysis of the airport network of India as a complex weighted network." In: *Physica A: Statistical Mechanics and its Applications* 387.12 (2008), pp. 2972–2980. DOI: 10.1016/j.physa.2008.01.077 (cit. on p. 3).

[Bar05] Albert-Laszlo Barabási. "The origin of bursts and heavy tails in human dynamics." In: *Nature* 435.7039 (2005), pp. 207–211. DOI: 10.1038/nature03459 (cit. on pp. 4, 25).

[Bar16] Albert-László Barabási. *Network Science*. Cambridge University Press, 2016. ISBN: 978-1107076266 (cit. on pp. 9, 11, 14).

[Bia+14] Ginestra Bianconi et al. "Triadic closure as a basic generating mechanism of communities in complex networks." In: *Phys. Rev. E* 90 (4 Oct. 2014), p. 042806. DOI: 10.1103/PhysRevE.90.042806 (cit. on p. 41).

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006. ISBN: 978-0387310732 (cit. on p. 49).

[Boc+06] S. Boccaletti et al. "Complex networks: Structure and dynamics." In: *Physics Reports* 424.4–5 (2006), pp. 175–308. DOI: 10.1016/j.physrep.2005.10.009 (cit. on p. 10).

[Cal+06] Guido Caldarelli et al. "Preferential attachment in the growth of social networks: the case of Wikipedia." In: *Physics Reports* (Mar. 2006). DOI: 10.1103/PhysRevE.74.036116 (cit. on p. 21).

[CB98] Robert H. Crites and Andrew G. Barto. "Elevator Group Control Using Multiple Reinforcement Learning Agents." In: *Machine Learning* 33.2 (1998), pp. 235–262. DOI: 10.1023/A:1007518724497 (cit. on p. 49).

[CF08] Nicholas A. Christakis and James H. Fowler. "The Collective Dynamics of Smoking in a Large Social Network." In: *New England Journal of Medicine* 358.21 (2008), pp. 2249–2258. DOI: 10.1056/NEJMsa0706154 (cit. on p. 33).

[CH97] Bruce Christianson and William S. Harbison. "Why isn't trust transitive?" In: *Security Protocols: International Workshop Cambridge*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 171–176. DOI: 10.1007/3-540-62494-5_16 (cit. on p. 13).

[CLM04] Paolo Crucitti, Vito Latora, and Massimo Marchiori. "Model for cascading failures in complex networks." In: *Phys. Rev. E* 69 (4 Apr. 2004), p. 045104. DOI: 10.1103/PhysRevE.69.045104 (cit. on p. 21).

[Coo12] William J. Cook. *In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation*. Princeton University Press, 2012. ISBN: 978-0691152707 (cit. on pp. 1, 2).

[Cor+09] Thomas H. Cormen et al. *Introduction to Algorithms, Third Edition*. 3rd. The MIT Press, 2009. ISBN: 978-0262033848 (cit. on p. 8).

[CSN09] Aaron Clauset, Cosma Rohilla Shalizi, and Mark Newman. "Power-Law Distributions in Empirical Data." In: *SIAM Review* 51.4 (2009), pp. 661–703. DOI: 10.1137/070710111 (cit. on p. 29).

[Dan+13] Cristian Danescu-Niculescu-Mizil et al. "A Computational Approach to Politeness with Application to Social Factors." In: (2013). arXiv: 1306.6078 (cit. on p. 3).

[Dev86] Luc Devroye. "Non-Uniform Random Variate Generation." In: New York, NY, USA: Springer-Verlag, 1986. Chap. General Principles in Random Variate Generation. ISBN: 978-1461386452 (cit. on p. 86).

[Die12] Reinhard Diestel. *Graph Theory, 4th Edition*. Vol. 173. Graduate texts in mathematics. Springer-Verlag, 2012. ISBN: 978-3642142789 (cit. on p. 7).

[Dun92] Robin Dunbar. "Neocortex size as a constraint on group size in primates." In: *Journal of Human Evolution* 22.6 (1992), pp. 469–493. DOI: 10.1016/0047-2484(92)90081-J (cit. on p. 40).

[EL03] Eli Eisenberg and Erez Y. Levanon. "Preferential Attachment in the Protein Network Evolution." In: *Phys. Rev. Lett.* 91 (13 Sept. 2003), p. 138701. DOI: 10.1103/PhysRevLett.91.138701 (cit. on p. 21).

[ER59] Paul Erdős and Alfréd Rényi. "On Random Graphs I." In: *Publicationes Mathematicae (Debrecen)* 6 (1959 1959), pp. 290–297 (cit. on p. 18).

[EV13] Ernesto Estrada and Eusebio Vargas-Estrada. "How peer pressure shapes consensus, leadership, and innovations in social groups." In: *Scientific Reports* 3 (Oct. 2013). DOI: 10.1038/srep02905 (cit. on p. 34).

[Fer+15] Alceu Ferraz Costa et al. "RSC: Mining and Modeling Temporal Activity in Social Media." In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '15. New York, NY, USA: ACM, 2015, pp. 269–278.

ISBN: 978-1450336642. DOI: 10.1145/2783258.2783294 (cit. on p. 26).

[For10] Santo Fortunato. "Community detection in graphs." In: *Physics Reports* 486.3–5 (2010), pp. 75–174. DOI: 10.1016/j.physrep.2009.11.002 (cit. on p. 14).

[GB08] Kwang-Il Goh and Albert-Laszlo Barabási. "Burstiness and memory in complex systems." In: *EPL (Europhysics Letters)* 81.4 (2008), p. 48002. DOI: 10.1209/0295-5075/81/48002 (cit. on p. 66).

[GN02] Michelle Girvan and Mark Newman. "Community structure in social and biological networks." In: *Proceedings of the national academy of sciences* 99.12 (2002), pp. 7821–7826. DOI: 10.1073/pnas.122653799 (cit. on p. 13).

[Gra73] Mark Granovetter. "The Strength of Weak Ties." In: *The American Journal of Sociology* 78.6 (1973), pp. 1360–1380. DOI: 10.1086/225469 (cit. on p. 13).

[Has+16] Ilire Hasani-Mavriqi et al. "The influence of social status and network structure on consensus building in collaboration networks." In: *Social Network Analysis and Mining* 6.1 (2016), p. 80. DOI: 10.1007/s13278-016-0389-y (cit. on p. 3).

[Hol13] Petter Holme. "Epidemiologically Optimal Static Networks from Temporal Network Data." In: *PLoS Comput Biol* 9.7 (July 2013), pp. 1–10. DOI: 10.1371/journal.pcbi.1003142 (cit. on pp. 16, 28).

[Hol15] Petter Holme. "Modern temporal network theory: a colloquium." In: *The European Physical Journal B* 88.9 (2015), p. 234. DOI: 10.1140/epjb/e2015-60657-4 (cit. on pp. 4, 15, 79).

[HS12] Petter Holme and Jari Saramäki. "Temporal networks." In: *Physics Reports* 519.3 (2012), pp. 97–125. DOI: 10.1016/j.physrep.2012.03.001 (cit. on pp. 4, 15).

[Hua+14] Grace C. Huang et al. "Peer Influences: The Impact of Online and Offline Friendship Networks on Adolescent Smoking and Alcohol Use." In: *Journal of Adolescent Health* 54.5 (2014), pp. 508–514. DOI: 10.1016/j.jadohealth.2013.07.001 (cit. on p. 33).

[Inc17] Alexa Internet Inc. *The top 500 sites on the web*. 2017. URL: http://www.alexa.com/topsites (visited on 02/11/2017) (cit. on p. 2).

[Jo+12] Hang-Hyun Jo et al. "Circadian pattern and burstiness in mobile phone communication." In: *New Journal of Physics* 14.1 (2012), p. 013055. DOI: 10.1088/1367-2630/14/1/013055 (cit. on p. 28).

[Kon+17] Philipp Koncar et al. "Exploring the Impact of Trolls on Activity Dynamics in Real-World Collaboration Networks." In: *Proceedings of the 7th Temporal Web Analytics Workshop*. Perth, Australia, Apr. 2017. DOI: 10.1145/3041021.3051116 (cit. on p. 80).

[KPV14] Márton Karsai, Nicola Perra, and Alessandro Vespignani. "Time varying networks and the weakness of strong ties." In: *Scientific Reports* 4 (Feb. 2014). DOI: 10.1038/srep04001 (cit. on pp. 3, 30, 39, 70).

[Kra+08] Hanna Krasnova et al. "Why participate in an online social network: An empirical analysis." In: *ECIS 2008 Proceedings* (2008). DOI: 10.7892/boris.47467 (cit. on p. 33).

[Kum+07] Jussi M. Kumpula et al. "Emergence of Communities in Weighted Networks." In: *Phys. Rev. Lett.* 99 (22 Nov. 2007), p. 228701. DOI: 10.1103/PhysRevLett.99.228701 (cit. on pp. 3, 42).

[LA87] Peter J. M. van Laarhoven and Emile H. L. Aarts. "Simulated annealing." In: *Simulated Annealing: Theory and Applications*. Dordrecht: Springer Netherlands, 1987, pp. 7–15. ISBN: 978-94-015-7744-1. DOI: 10.1007/978-94-015-7744-1_2 (cit. on p. 49).

[LH08] Jure Leskovec and Eric Horvitz. "Planetary-scale Views on a Large Instant-messaging Network." In: *Proceedings of the 17th International Conference on World Wide Web*. WWW '08. Beijing, China: ACM, 2008, pp. 915–924. ISBN: 978-1605580852. DOI: 10.1145/1367497.1367620 (cit. on p. 14).

[Liu+14] Suyu Liu et al. "Controlling Contagion Processes in Activity Driven Networks." In: *Phys. Rev. Lett.* 112 (11 Mar. 2014), p. 118702. DOI: 10.1103/PhysRevLett.112.118702 (cit. on p. 31).

[LSK15] Guillaume Laurent, Jari Saramäki, and Márton Karsai. "From calls to communities: a model for time-varying social networks." In: *The European Physical Journal B* 88.11 (2015), p. 301. DOI: 10.1140/epjb/e2015-60481-x (cit. on pp. 3, 5, 32, 38, 46, 56, 60, 70, 77).

[LTD13] Renaud Lambiotte, Lionel Tabourier, and Jean-Charles Delvenne. "Burstiness and spreading on temporal networks." In: *The European Physical Journal B* 86.7 (2013), p. 320. DOI: 10.1140/epjb/e2013-40456-9 (cit. on p. 4).

[Mal+08] R. Dean Malmgren et al. "A Poissonian Explanation for Heavy Tails in e-mail Communication." In: *Proceedings of the National Academy of Sciences of the United States of America* 105.47 (2008), pp. 18153–18158. DOI: 10.1073/pnas.0800332105 (cit. on p. 25).

[Mas15] Naoki Masuda. "Opinion control in complex networks." In: *New Journal of Physics* 17.3 (2015), p. 033031. DOI: 10.1088/1367-2630/17/3/033031 (cit. on p. 34).

[Mir+13] Giovanna Miritello et al. "Limited communication capacity unveils strategies for human interaction." In: *Scientific reports* 3 (2013). DOI: 10.1038/srep01950 (cit. on p. 41).

[Mis+15] Dina Mistry et al. "Committed activists and the reshaping of status-quo social consensus." In: *Phys. Rev. E* 92 (4 Oct. 2015), p. 042805. DOI: 10.1103/PhysRevE.92.042805 (cit. on p. 32).

[ML16] Naoki Masuda and Renaud Lambiotte. *A Guide to Temporal Networks*. World Scientific Publishing Europe Ltd., 2016. ISBN: 978-0199206650 (cit. on pp. 25, 66).

[MSP15] Antoine Moinet, Michele Starnini, and Romualdo Pastor-Satorras. "Burstiness and Aging in Social Temporal Networks." In: *Phys. Rev. Lett.* 114 (10 Mar. 2015), p. 108701. DOI: 10.1103/PhysRevLett.114.108701 (cit. on pp. 5, 31, 32).

[MSP16] Antoine Moinet, Michele Starnini, and Romualdo Pastor-Satorras. "Aging and percolation dynamics in a Non-Poissonian temporal network model." In: *Phys. Rev. E* 94 (2 Aug. 2016), p. 022316. DOI: 10.1103/PhysRevE.94.022316 (cit. on pp. 5, 29, 32, 67).

[Mur+15] Yohsuke Murase et al. "Modeling the Role of Relationship Fading and Breakup in Social Network Formation." In: *PLOS ONE* 10.7 (July 2015), pp. 1–14. DOI: 10.1371/journal.pone.0133005 (cit. on p. 3).

[New10] Mark Newman. *Networks: An Introduction*. New York, NY, USA: Oxford University Press, Inc., 2010. ISBN: 978-1786341143 (cit. on pp. 2, 10, 11, 14, 18, 28).

[Onn+07] Jukka-Pekka Onnela et al. "Structure and tie strengths in mobile communication networks." In: *Proceedings of the National Academy of Sciences* 104.18 (2007), pp. 7332–7336. DOI: 10.1073/pnas.0610245104 (cit. on p. 3).

[Pao11] Teo Paoletti. *Leonard Euler's Solution to the Konigsberg Bridge Problem*. 2011. URL: http://www.maa.org/press/periodicals/convergence/leonard-eulers-solution-to-the-konigsberg-bridge-problem (visited on 02/01/2017) (cit. on p. 1).

[Per+12a] Nicola Perra et al. "Activity driven modeling of time varying networks." In: *Scientific reports* 2 (2012), p. 469. DOI: 10.1038/srep00469 (cit. on pp. 4, 5, 28, 38, 76).

[Per+12b] Nicola Perra et al. "Random Walks and Search in Time-Varying Networks." In: *Phys. Rev. Lett.* 109 (23 Dec. 2012), p. 238701. DOI: 10.1103/PhysRevLett.109.238701 (cit. on p. 31).

[PTR05] Lisa M. Powell, John A. Tauras, and Hana Ross. "The importance of peer effects, cigarette prices and tobacco control policies for youth smoking behavior." In: *Journal of Health Economics* 24.5 (2005), pp. 950–968. DOI: 10.1016/j.jhealeco.2005.02.002 (cit. on p. 33).

[Rag+13] Vasanthan Raghavan et al. "Coupled hidden markov models for user activity in social networks." In: *2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*. July 2013, pp. 1–6. DOI: 10.1109/ICMEW.2013.6618397 (cit. on p. 26).

[Ray+08] Debajyoti Ray et al. "Bayesian Model of Behaviour in Economic Games." In: *Advances in Neural Information Processing Systems 21*. 2008, pp. 1345–1352 (cit. on p. 49).

[RFP14] Alessandro Rizzo, Mattia Frasca, and Maurizio Porfiri. "Effect of individual behavior on epidemic spreading in activity-driven networks." In: *Phys. Rev. E* 90 (4 Oct. 2014), p. 042801. DOI: 10.1103/PhysRevE.90.042801 (cit. on pp. 5, 31).

[RH11] Fergal Reid and Neil Hurley. "Diffusion in Networks With Overlapping Community Structure." In: (2011). arXiv: 1105.5849 (cit. on p. 22).

[RPP16] Alessandro Rizzo, Biagio Pedalino, and Maurizio Porfiri. "A network model for Ebola spreading." In: *Journal of Theoretical Biology* 394 (2016), pp. 212–222. DOI: 10.1016/j.jtbi.2016.01.015 (cit. on pp. 5, 31).

[Sch09] Sebastian Schnettler. "A small world on feet of clay? A comparison of empirical small-world studies against best-practice criteria." In: *Social Networks* 31.3 (2009), pp. 179–189. DOI: 10.1016/j.socnet.2008.12.005 (cit. on p. 14).

[Sei11] Wilfried Seidel. "Mixture Models." In: *International Encyclopedia of Statistical Science*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 827–829. ISBN: 978-3-642-04898-2. DOI: 10.1007/978-3-642-04898-2_368 (cit. on p. 65).

[SF84] Lisa K Carden Smith and Susan A Fowler. "Positive peer pressure: The effects of peer monitoring on children's disruptive behavior." In: *Journal of applied behavior analysis* 17.2 (1984), pp. 213–227. DOI: 10.1901/jaba.1984.17-213 (cit. on p. 33).

[Sim+01] Bruce Simons-Morton et al. "Peer and Parent Influences on Smoking and Drinking among Early Adolescents." In: *Health Education & Behavior* 28.1 (2001), pp. 95–107. DOI: 10.1177/109019810102800109 (cit. on p. 33).

[SKK15] Albert Sunny, Bhushan Kotnis, and Joy Kuri. "Dynamics of history-dependent epidemics in temporal networks." In: *Phys. Rev. E* 92 (2 Aug. 2015), p. 022811. DOI: 10.1103/PhysRevE.92.022811 (cit. on p. 32).

[SP13] Michele Starnini and Romualdo Pastor-Satorras. "Topological properties of a time-integrated activity-driven network." In: *Physical Review E* 87.6 (June 2013). DOI: 10.1103/physreve.87.062807 (cit. on p. 30).

[SSL16] Vedran Sekara, Arkadiusz Stopczynski, and Sune Lehmann. "Fundamental structures of dynamic social networks." In: *Proceedings of the national academy of sciences* 113.36 (2016), pp. 9977–9982. DOI: 10.1073/pnas.1602803113 (cit. on p. 3).

[TM69] Jeffrey Travers and Stanley Milgram. "An Experimental Study of the Small World Problem." In: *Sociometry* 32.4 (1969), pp. 425–443. DOI: 10.2307/2786545 (cit. on p. 14).

[TS92] K. Thulasiraman and M. N. S. Swamy. *Basic Concepts, in Graphs: Theory and Algorithms*. First. John Wiley & Sons, Inc., 1992. ISBN: 978-0471513568 (cit. on p. 7).

[Váz+06] Alexei Vázquez et al. "Modeling bursts and heavy tails in human dynamics." In: *Phys. Rev. E* 73 (3 Mar. 2006), p. 036127. DOI: 10.1103/PhysRevE.73.036127 (cit. on pp. 25, 27, 28).

[Wal+16] Simon Walk et al. "Activity Dynamics in Collaboration Networks." In: *ACM Trans. Web* 10.2 (May 2016), 11:1–11:32. DOI: 10.1145/2873060 (cit. on pp. 3, 5, 35, 47, 51, 77, 80).

[Wan+03] Yang Wang et al. "Epidemic spreading in real networks: an eigenvalue viewpoint." In: *22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings.* IEEE. Oct. 2003, pp. 25–34. DOI: 10.1109/RELDIS.2003.1238052 (cit. on p. 21).

[Wan+16] Jian Wang et al. "AST: Activity-Security-Trust driven modeling of time varying networks." In: *Scientific reports* 6 (2016), p. 21352. DOI: 10.1038/srep21352 (cit. on p. 32).

[Wat04] Duncan J Watts. *Six Degrees: The Science of a Connected Age*. W. W. Norton, 2004. ISBN: 978-0393076127 (cit. on p. 23).

[Wat99] Duncan J. Watts. *Small worlds: the dynamics of networks between order and randomness*. Princeton University Press, 1999. ISBN: 978-0691117041 (cit. on p. 14).

[Wel+01] Barry Wellman et al. "Does the Internet increase, decrease, or supplement social capital? Social networks, participation, and community commitment." In: *American behavioral scientist* 45.3 (2001), pp. 436–455. DOI: 10.1177/00027640121957286 (cit. on p. 2).

[WF94] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*. Vol. 8. Cambridge university press, 1994. ISBN: 978-0521387071 (cit. on pp. 2, 12).

[WS98] Duncan J Watts and Steven H Strogatz. "Collective dynamics of 'small-world' networks." In: *nature* 393.6684 (1998), pp. 440–442. DOI: 10.1038/30918 (cit. on pp. 2, 9, 15, 23, 24).

[WW99] Yufei Wang and Zheng Wang. "Explicit routing algorithms for internet traffic engineering." In: *Computer Communications and Networks, 1999. Proceedings. Eight International Conference on*. IEEE. 1999, pp. 582–588. DOI: 10.1109/ICCCN.1999.805577 (cit. on p. 2).

[Zac77] W.W. Zachary. "An information flow model for conflict and fission in small groups." In: *Journal of Anthropological Research* 33 (1977), pp. 452–473. DOI: 10.1086/jar.33.4.3629752 (cit. on p. 12).