UNIVERSITY OF AGDER

Visibility Graph Analysis of Real-Life and GARCH-Simulated Financial Time-Series

Espen Segberg & Sindre Skoglund

01.06.2017

Supervisor

Jochen Jungeilges

Master Thesis

University of Agder 2017

Faculty of Business and Law at UiA

Abstract

In this thesis, we will use the natural and horizontal visibility graph algorithms to look at financial time-series. We will apply the two algorithms to time-series generated from thirteen sets of simulations with the help of GARCH-processes, as well as ten subsets of daily data from S&P 500, both in its raw form, and in terms of returns. The five network statistics mean degree, average shortest path length, assortativity, local transitivity and global transitivity will be used in an attempt to see whether the visibility graph algorithms are in fact able to differentiate between time-series of different structures. This thesis will contribute to the literature with a more extensive investigation of the visibility graph's behavior when applied to financial time-series than we have yet to see. The results obtained in this thesis will show that especially the natural visibility graph has the ability to differentiate between different time-series, and that it is an analysis tool that deserves more research in the future.

Table of content

List of tables

List of figures

1. Introduction

Time-series analysis of financial processes is undoubtedly an important aspect in the understanding of financial data. By performing a time-series analysis on the data, one can obtain important statistics, estimated parameters and other characteristics that can later be used to understand, predict and forecast the behavior of the data.

As a variety of time-series analysis tools has been popular for a good amount of time, the research has obviously come a long way in this field. However, most methods of time-series analysis have some prerequisite of the data's structure, like for example stationarity. When time-series data are stationary, it shows constant mean and variance through time, but this is normally not seen in financial time-series, and transformation of the data is necessary in order to make it compatible with the analysis-tool to be used. Some examples of ways to transform data are to use returns instead of prices or to use log transformations of the data. While these transformations often are effective, in that it saves the essence of the data, while allowing you to reliably analyze it, some information of the original time-series will also inevitably be lost.

As a reason, it is important and interesting to still investigate new methods of time-series analysis that can avoid these transformations, or possibly rely on alternative structures. One form of alternative analysis that has grown in popularity over the last decade is complex network theory. For it to be used effectively on a time-series, the time-series will have to be plotted into a network with the help of an algorithm. Consequently, some information will be lost here, as in the case for classical linear time-series analysis, but depending on the algorithm chosen, theory suggests it should be possible to avoid transformations that remove the non-linear properties of the data.

This paper will cover analysis of time-series by complex network theory, and the two algorithm we have chosen to utilize are one who was first introduced by Lacasa, Luque, Ballesteros, Luque, and Nuño (2008) called the visibility graph, and its successor, the horizontal visibility graph (Luque, Lacasa, Ballesteros, & Luque, 2009). These algorithms will be discussed thoroughly in later sections. However, it is worth noting that they are straightforward geometric algorithms, making it suitable for virtually every time-series you can imagine.

Although the two visibility algorithms have been around for about a decade, there are still question marks regarding their effectiveness as an analytical tool, especially from a financial point of view. In an effort to get a better base of knowledge surrounding this tool, we will further investigate its properties.

In a recent paper by Zhang, Zou, Zhou, Gao, and Guan (2017) it was studied how visibility graphs and the statistics gathered from it would behave when applied to autoregressive processes of form AR(1) and AR(2). Several different parameters and combinations of parameters were used for the AR(1) and AR(2)-process, both positive and negative. Motivated by the mentioned study, we will be performing similar tests, but rather than studying autoregressive processes, we will be looking towards a better financial interpretation. Consequently, generalized autoregressive conditional heteroscedasticity-processes, more famously known as GARCH will be used.

The GARCH-model was first introduced by Bollerslev (1986) as an extension of the autoregressive conditional heteroscedasticity-model (ARCH) proposed by Engle (1982). Ever since, GARCH has been an extremely popular tool in financial forecasting and time-series modeling, and the most popular type, GARCH(1,1) has been said to describe the volatility dynamics of almost any financial time-series (Engle, 2004). What makes GARCH-processes very interesting and well suited for what we want to do, is that rather than assuming that the error of variance always is constant, it models a process where the variance changes with time.

First we will use GARCH(1,1) to estimate the parameters from a time-series consisting of daily return-data from the S&P 500-index. These estimated parameters will be used as a basis-point in our simulations, but we will also vary them, and in total perform simulation of thirteen different parameter-combinations. We have chosen to investigate six different combinations that we believe will cover the parameter space in an adequate way, and will perform seven simulations where one or both of the two parameters are set to zero. In addition, we will look at natural visibility graphs and horizontal visibility graphs on 10-year subsets of the S&P 500-index. This will be done for both the price-series as well as the return-series.

Similarly to the mentioned study from Zhang et al. (2017), we will keep our focus on five different network statistics. These are degree distribution, average shortest path length, assortativity by degree, global transitivity and local transitivity, and will all be described further in later sections. These are statistics that are common of use in network theory, and should give us a good idea about the properties of the networks created from the different time-series. In general, there are limited information and knowledge in the literature about how natural visibility graphs and horizontal visibility graphs behave when applied to financial time-series.

As we will see in the Section 2, most of the research in the literature has had a focus on what kind of networks the visibility algorithms are creating, while few has investigated what the networks statistics actually look like under controlled conditions. We believe our focus on how the network statistics differ

between different time-series, and whether these network statistics actually can be used to describe a financial time-series, clearly separate itself from previous studies within this field. This also includes the mentioned study by Zhang et al. (2017) that we have used as motivation for this thesis, considering how they looked at whether their network statistics was following similar patterns as their auto correlation function, and we, as mentioned have a slightly different approach where we use them to differentiate between different types of time-series.

With this study, we will take a significant step towards describing the behavior of visibility graph algorithms in a financial context, something we deem as an important part in the process of eventually evaluating the effectiveness of this analytical tool. Our combination of data sets, the different network statistics and both the natural and horizontal visibility graph should provide us with information that can later be of value when investigating most financial time-series.

Going from here, this thesis will be structured in the following way. A literature-review mainly concerning network theory and applications for the visibility graph will be given in Section 3. In Section 4 we will give a review of the methodology used, and will describe the processes we have conducted to obtain our results. Section 5 will give information on the data-sets used, as well as the sources they are obtained from, and we will then present our results for both simulation-study and the study of S&P 500 in section 6. Finally, the thesis will be completed with a discussion in Section 7 and a conclusion in section 8.

2. Literature review

As we briefly discussed in the introduction, the motivation behind this study is to investigate the possible success of an analytical tool that can be applied to data that show non-linear properties. Non-linear properties are something we often see in financial data, but are something most analytical tools are unable to operate with, due to pre-requisites in the data's structure. Brooks (2008, p. 380) states that it is especially three main sources of non-linearity that we often observe in financial data. The first is called leptokurtosis. This term refers to a distribution with fatter tails and a higher kurtosis than what would be the case in a traditional Gaussian distribution. Leptokurtosis is also commonly known as fat-tailedness, but we will stick to the former way of describing it in this paper.

Secondly, we often observe what is called volatility clustering. As explained by Mandelbrot (1963) this is a tendency often seen in financial data, where the volatility of a time-series varies in bursts, rather than staying constant throughout the whole time-series. Easily told, it is the tendency of small changes being followed by small changes either positive or negative and for large changes to be followed by large changes either positive or negative.

The third and final non-linear effect mentioned by Brooks (2008, p. 380) is something called leverage effects, which is a tendency seen in financial data where the volatility rises more after a big price fall than after a similar price rise.

As we mentioned in the introduction, we will utilize network theory in this thesis. A network can be described as a simplified representation that reduces a system to an abstract structure (Newman, 2010, p. 2). A network is a collection of points that are joined together in pairs by lines in-between them. These points will be referred to as nodes by us, but are also often referred to as vertices in the literature. The lines between the nodes are referred to as edges. The nodes and edges can be labeled with several types of additional information of the connectivity, and the possibilities are too many for us to cover, but there are two types structures that are important to note in regards to the visibility graph. A network from a visibility graph algorithm is always connected and undirected. (Lacasa et al., 2008). What this implies will be discussed later in the thesis.

Although the usage of complex network theory in time-series analysis is relatively new, network science stretches as far back as the mid-18th century, and possibly even further. The study of large complex networks started in the 1950s, and Erdös and Rényi (1959), (1960) took a substantial step towards the network modeling we see today, by introducing what is known as random graphs. Here it is assumed that

any given two nodes in a network have a certain probability of being connected to each other, and that the amount of total edges in the network consequently were dependent on this probability and the amount of nodes in the network. However, with the improvement of computers, it has later been shown that real-life networks rarely follow a complete random distribution of its edges, and two seminal complex network models has later been proposed. Watts and Strogatz (1998) introduced a network model called small-world networks. They showed with this model that networks can be highly clustered like for example regular lattices, yet have small path length between any two nodes, like we often observe in random graphs.

A second highly popular model is the one of scale-free networks, proposed by Barabási and Albert (1999). The authors demonstrate that a lot of real-life networks follow what is called scale-free properties, meaning that the networks degree distribution follows a power-law $P(k) \sim k^{-y}$. Especially two main mechanisms have been said to cause large networks to convert into scale-free networks. The first one is that these networks are continuously expanding through addition of new nodes in the network. The second is that new nodes in a network tend to connect itself with existing nodes of a high degree. This mechanism is often referred to as preferential attachment in the literature.

The process of mapping a time series into a network has been widely discussed with the increased popularity of complex network theory since the beginning of this century. The process of transforming a time series into a complex network is interesting because of the possibility that it can provide us with information that are not available from classic linear time-series analysis, and especially in cases where information gets lost due to transformation of data, as we discussed in the introduction. Several methods of plotting a time-series into a network have been proposed in recent years, showing the growing interest in the topic. Although most of these methods will not be utilized in this paper, it is worth giving a brief review of the literature at hand.

Zhang and Small (2006) developed a method of constructing complex networks from pseudo-periodic time series. Each cycle in the time series represents a node in the network, and whether nodes were connected or not was determined by the strength of temporal correlation between different cycles.

Yang and Yang (2008) introduced an algorithm that constructed a complex network from the time series' correlation matrix. A correlation coefficient is found for each element C_{ij} in the time-series, and connections in the network were based off this correlation coefficient. A critical value C_r has to be chosen as a separating point between nodes that are connected and those who are not.

Xu, Zhang, and Small (2008) embedded a time-series into an appropriate phase-space following the methods of Packard, Crutchfield, Farmer, and Shaw (1980), and then using each phase-space point as a node to create a network.

Lacasa et al. (2008) were the first to introduce the visibility graph algorithm, hereby referred to as the natural visibility algorithm to avoid confusion. In short, it is an algorithm for translating a time-series into a complex network. Every observation in the time series will be treated as a node in a network, and connection between nodes will be realized if no intermediate nodes intervene the visibility between them. An intermediate node will intervene the visibility if it lays as high or higher than the visibility-line created between the nodes at time t. The algorithm and its properties will be more thoroughly covered in Section 3. What most importantly separates the natural visibility graph from the other methods of plotting time-series into networks, is its straight-forward geometric interpretation. Its easy algorithm makes it theoretically possible to apply this technique to virtually any time-series you can imagine. However, whether the algorithm will be effective on all sorts of time-series is still something that needs to be investigated, and it is our understanding that the most successful implementation has been on time-series with constant mean.

Several alternative ways to use the visibility graph algorithm has later been proposed. Luque et al. (2009) followed up the natural visibility graph with an algorithm for the horizontal visibility graph. Here, two nodes x_i and x_j can only be connected if no intermediate node is higher than the two nodes. In other words, this is an even simpler algorithm than the natural visibility graph, and the horizontal visibility graph will always be a subgraph of the natural visibility graph.

We will utilize the natural and horizontal visibility graph algorithm in this paper, and they have both previously been successfully applied in a variety of fields. Some of these fields include hurricane data (Elsner, Jagger, & Fogarty, 2009), turbulence (Liu, Zhou, & Yuan, 2010), solar flares (Yu, Anh, & Eastes, 2012), seismology (Telesca & Lovallo, 2012), wind speeds (Pierini, Lovallo, & Telesca, 2012) and human heartbeat dynamics (Shao, 2010), to name a few.

Although the natural and horizontal visibility graph have not been extensively used in the financial world, some research has in recent years been done by using this method on economic and financial time-series. Long (2013) analyzed gold fixings from the London Bullion Market Association from 1971-2012 using the natural visibility graph algorithm. He found that both the return-series and price-series are fractal series with long-term correlation. By analyzing the relationship between the Hurst exponent and the degree

distribution, it was shown that the logarithmic price series is of fractional Brownian motion and the logarithmic return series agrees with fractal Gaussian noise. Furthermore, it was showed that the gold price-series is a small-world network with hierarchical characteristics, which means that the gold price fluctuates in form of waves, in agreement with Elliot's wave theory. Similarly, Sun, Wang, and Gao (2016) investigated time-series on the North American gas price market, and found the network generated with the natural visibility graph to have both small-world and scale-free properties. The scale-free properties means that the impact on price fluctuations from the many low degree nodes are small, while the few high degree nodes having a great influence on the price fluctuation. The small-world characteristics means that the impact range of hubs is large or price fluctuations of a long period leads to hubs. High degree nodes often correspond to significant economic or geopolitical events, such as the Iraq war 2003, and is found to have a huge impact on the neighboring nodes in the network.

Ni, Jiang, and Zhou (2009) applied the natural visibility graph to study the degree distributions extracted from fractional Brownian motions and multifractal random walks. They found that the degree distributions exhibit power-law behavior and that the power-law exponent is a linear function of the Hurst exponent inherent in the series. In addition, they showed that temporal correlation of the time-series is what mainly determines the degree distribution.

Yang, Wang, Yang, and Mang (2009) applied the natural visibility graph to study six different exchange rate-series. First off all, they found that all the series they studied converted into scale-free networks with a hierarchical structure. They also found that the visibility graph can be used to find reliable values for the Hurst exponent in time-series that are of the fractional Brownian motion type. This finding is supported by Lacasa, Luque, Luque, and Nuño (2009), who proved that a fractional Brownian motion maps into a scale-free visibility graph and that the power-law degree distribution depends linearly on the Hurst exponent.

Economic time series have also been studied using the natural visibility graph. Wang, Li, and Wang (2012) did a visibility graph analysis on four different Chinese macroeconomic time-series. The industry series converted into exponential networks and the GDP series converted into a scale-free network. All four networks also showed small-world properties, characterized by their high clustering coefficient and small average shortest path length. The observations in the series with a high degree was found to usually reflect turning points in the business activity. Wang et al. (2012) argued that the use of the natural visibility graph offers a new way to understand the dynamic processes of economic development.

Zhuang, Small, and Feng (2014) used the natural visibility graph to study the segmentation of developed financial markets. Time-series from industry sector Indices for the world and 22 developed markets from 1973-2012 were studied. They found a clear trend of market integration in the time series. By dividing the full time-series into smaller 10-year periods, they were able to locate the historical incidents leading to increased integration and in some cases leading to a segmentation of the market. The use of the natural visibility graph allowed them to find large cycles and noise in the series by detecting different communities. The window from 2003-2012 was a relatively steady period when it comes to segmentation. They found that there were a few incidents early in the period that increased the market integration, but later in the period, some incidents led to market segmentation. The natural visibility graph allows them to locate incidents that drive either integration or segmentation in a market.

As we mentioned earlier in the literature review, our focus will revolve around applications of the natural and horizontal visibility graph to financial time series. A more thorough look at how we plot time-series into networks with the two algorithms will be given in the next section, together with methodology surrounding GARCH, the network statistics utilized and the simulation strategy we have used.

3. Methodology

3.1 GARCH-simulation

As previously mentioned, we will be simulating data for this thesis with the help of a generalized autoregressive heteroscedasticity-process (GARCH) as proposed by Bollerslev (1986), which is an extension of the autoregressive conditional heteroscedasticity-model (ARCH) proposed by Engle (1982).

The ARCH-model was developed as a solution to two problems often seen when analyzing financial data. The first being heteroscedasticity, meaning that the error of variance is not constant. A classical linear regression model can be written as

$$Y = \gamma_0 + \gamma_1 X_1 + \gamma_2 X_2 + \cdots + \gamma_q X_q + e, \;\; e \sim N(0, \sigma^2) \tag{1}$$

In this model homoscedasticity is assumed, and failure to satisfy this assumption will likely mean that estimates of standard error will be wrong.

The second reason for the development of the ARCH-model was the fact that financial data has a tendency to exhibit a volatility clustering (Mandelbrot, 1963), which we discussed in Section 2.

What separates an ARCH model from a classical linear regression model like Equation (1), is while this often can be expressed the error distributed as $e_t \sim N(0, \sigma^2)$, the ARCH-model will have a variance of error which is conditional with time, giving us $e_t \sim N(0, \sigma_t^2)$. Instead of the variance of error being constant, we can regress the conditional variance σ_t^2 of the error term on the immediately previous values of the squared errors. The process of finding σ_t^2 can be written in the following way

$$\sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \alpha_2 e_{t-2}^2 + \cdots + \alpha_q e_{t-q}^2 \tag{2}$$

Equation (2) is describing an autoregressive model of length AR(q), which would result in the ARCH-model being of form ARCH(q) as well. Basically any integer above zero can be used for q in the model, but a regression based off one previous value is normally more than sufficient (Engle, 2004). It is also worth noting that conditional variance, because of it being a squared number, has to be positive. For this condition to always be satisfied, some constraints are needed for the coefficients in the AR-model. It is common to set $\alpha_i \geq 0, i > 0$ for all values of i larger than zero, although this is actually stricter than what is absolutely needed (Brooks, 2008, p. 389). α_0 in the model is a constant that has to be larger than zero, but at the same time mostly are estimated to be very close to zero.

An ARCH(1)-model can for example be expressed in the following manner:

$$Y = \gamma_0 + \gamma_1 X_1 + \gamma_2 X_2 + e_t, \qquad e_t \sim N(0, \sigma_t^2) \tag{3}$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 \tag{4}$$

However, the ARCH-model is rarely used in financial modelling anymore, due to some limitations which can be accounted for by some of the many models made to succeed the ARCH-model (Brooks, 2008, p. 391). One of these models are the previously mentioned GARCH-model proposed by Bollerslev (1986). Where GARCH separates itself from ARCH is in that it estimates the conditional variance of error based of both the autoregressive part used in ARCH and a lag of previous conditional variances. Thus, we get

$$\sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \cdots + \alpha_q e_{t-q}^2 + \beta_1 \sigma_{t-1}^2 + \cdots + \beta_q \sigma_{t-q}^2 \tag{5}$$

And consequently for a GARCH(1,1) the conditional variance is forecasted as

$$\sigma_t^2 = \alpha_0 + \alpha_1 e_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \tag{6}$$

Where the beta-parameters are the ones forecasting a future conditional variance based on previous observations. Where ARCH can be said to be closely related to an AR-process due to its auto-regression of the variance errors, GARCH is more related to an autoregressive moving average-process (ARMA), since a moving average-process of the lagged conditional variances are added to the model.

(Brooks, 2008, pp. 393-394) argues that the GARCH-model outperforms ARCH in several ways, including it being more parsimonious, which makes it avoid overfitting and as a consequence makes it less likely to breach the previously mentioned constraints on non-negativity in the conditional variance of error. At the same time it is said that any model of GARCH(p,q) higher than GARCH(1,1) rarely is used, since the latter is able to capture the volatility clustering in financial data in a sufficient way. This statement is supported by Engle (2004) who argues that GARCH(1,1) is considered as the workhorse among all the models that has been succeeding ARCH in the later years, and that it is able to describe almost any financial time-series.

Consequently, we will be focusing exclusively on GARCH(1,1). We mentioned earlier that it is necessary to set some constraint for negative parameters in the ARCH-model. We will carry this constraint over to the GARCH-model, and just simulate with non-negative parameters. This way, we are absolutely sure that we avoid a situation where we estimate the conditional variance to be a negative number. A second constraint comes from the relationship between α_1 and β_1. The value of these two parameters must never equal or exceed one, giving us $\alpha_1 + \beta_1 < 1$ as an unalterable constraint. When we estimate parameters from a time-series with a GARCH, the maximum log-likelihood approach will be used.

3.2 Simulation strategy

As we mentioned earlier, there will in the case of GARCH(1,1)-simulation be a parameter space in which the parameter subscripts will constrained by. In total, we will investigate six different points within this parameter space. We believe this will give us a good idea about the results we can achieve over the parameter space as a whole. In addition, we will run seven simulations where one or both of the two parameters are set to zero.

In order to create results we believe to be valid, we will run one hundred replications of each simulation. The GARCH(1,1)-simulation will obviously not generate the same time-series each time, and multiple replications are therefore needed to give validity to the estimations.

Further, we will assume that all the network statistics we estimate from our time-series should come from a normal distribution. This makes us able to use maximum likelihood estimation to estimate the network statistics that will be presented later in this thesis. In all cases, the maximum likelihood estimation will simply be the arithmetic mean of observations, formally written as

$$\bar{X} = \frac{\sum_{t=1}^{N} X_t}{N} \tag{7}$$

Where N obviously is the number of replications, in our case 100, and $\overline{X_t}$ indicates an estimation from a given simulation. At the same time, we can never be absolutely sure what the actual arithmetic mean of an unlimited amount of replications would be, and will have to create a confidence interval in which we believe the mean will lie. The confidence interval will be computed in the following way:

$$\bar{X} \pm Z_{\alpha/2} * \frac{\sigma}{\sqrt{N}} \tag{8}$$

Where σ is the standard deviation and $Z_{\alpha/2}$ is alpha half percentage point from a z-table of a standard normal distribution. We will be working with a confidence interval of 95 %, giving us α of 0.05, which is then divided by two, since we have a two-tailed distribution. This will ultimately lead to an approximate value of 1.96 for $Z_{\alpha/2}$. The confidence interval will in this situation represent our interval estimate of the mean of a given network statistic, and it is given by the estimated mean ± the confidence interval.

Lastly, it is worth mentioning that all simulations in this thesis was done with 3000 observations, and that the first 500 were discarded, in order to avoid possible start-up transients and biases.

3.3 Visibility graph algorithms

The visibility algorithm was first proposed by Lacasa et al. (2008). The algorithm is a simple and fast computational method that converts a time series into a graph consisting of nodes and edges. The graph constructed inherits several of the time series properties and theory suggests that the graph can be used to reliably study the time-series it was generated from.

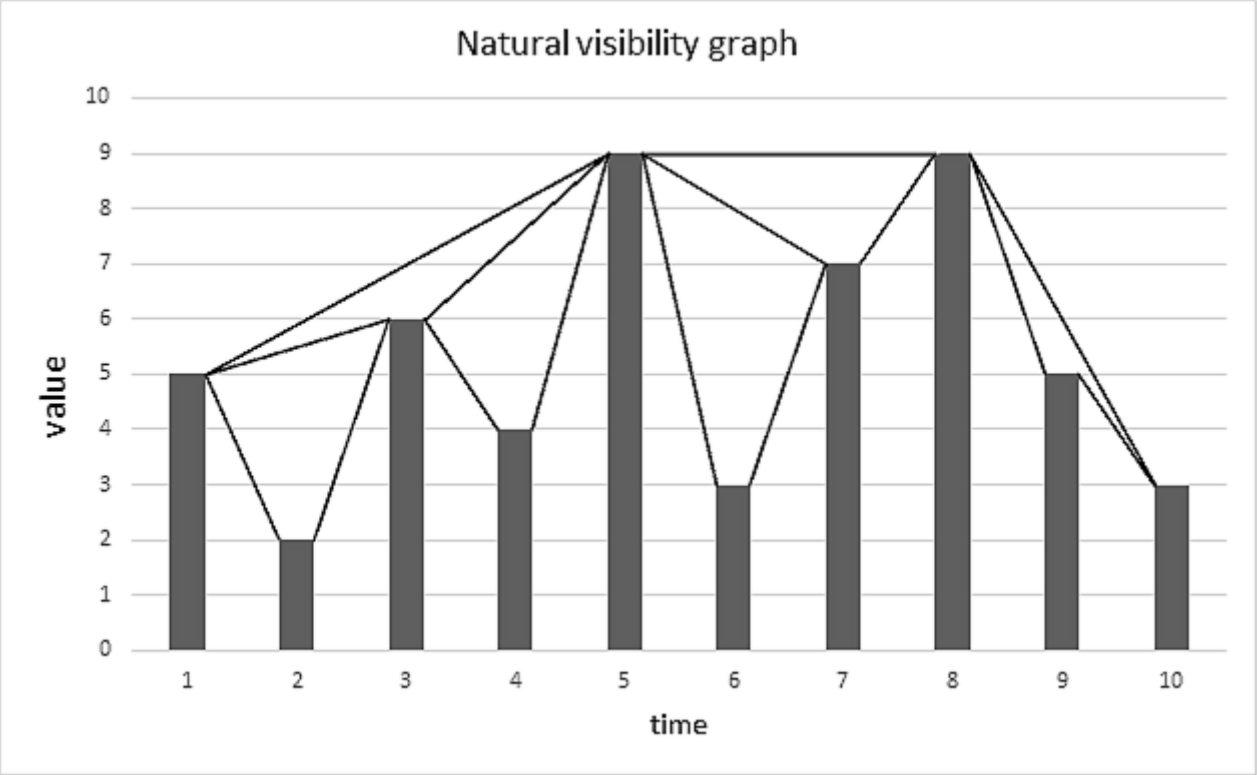

Figure 1: Example of the natural visibility algorithm applied to a time-series consisting of ten observations create by us.

The visibility algorithm produces a graph from the time series. Two nodes in the graph are connected if they have visibility to each other. This implies that if you can draw a straight line between the top of the two nodes, that does not intersect with any other intermediate observation, the visibility criterion is fulfilled. More formally, Lacasa et al. (2008) establish the following visibility criteria for the natural visibility graph: in a given time-series (t, y), where t is the time variable and y is the value of an observation, two arbitrary data values (t_a, y_a) and (t_b, y_b) will have visibility, and consequently will become two connected nodes of the associated graph, if any other data (t_c, y_c) placed between them fulfills:

$$y_c < y_b + (y_a - y_b)\frac{t_b - t_c}{t_b - t_a} \tag{9}$$

In Figure 1, we have given an example of a time-series where we have applied the natural visibility algorithm. We have visualized each observation in the time-series as a pole, where the value of the observation is the height of the pole. Lines are drawn between the poles, and these lines indicates that the visibility criterion has been fulfilled between two observations. Based on these fulfillments, we are able to create an adjacency matrix and a network. The meaning of the former will be described further down in this section. As we can see from the network on the right in Figure 2, we have turned the observations of the time-series in Figure 1 into nodes of a network, and the edges connecting the nodes are similar to the lines drawn in Figure 1.

	1	2	3	4	5	6	7	8	9	10
1	0	1	1	0	1	0	0	0	0	0
2	1	0	1	0	0	0	0	0	0	0
3	1	1	0	1	1	0	0	0	0	0
4	0	0	1	0	1	0	0	0	0	0
5	1	0	1	1	0	1	1	1	0	0
6	0	0	0	0	1	0	1	0	0	0
7	0	0	0	0	1	1	0	1	0	0
8	0	0	0	0	1	0	1	0	1	1
9	0	0	0	0	0	0	0	1	0	1
10	0	0	0	0	0	0	0	1	1	0

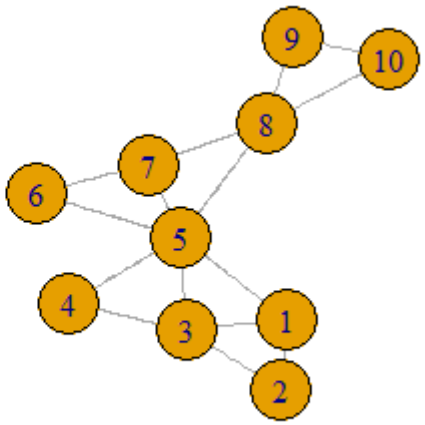

Figure 2: Adjacency matrix and graphical visualization for the network created from the time-series in Figure 1.

According to Lacasa et al. (2008) there are three main structures that will always be present in a visibility graph. First of all, it is always a connected graph. What this means, is that if you pick two arbitrary nodes from the graph, they will always be indirectly connected by at least one possible path in-between them. With this assumption, another assumption follows, which is that all nodes must have at least one other node connected to it, also known as neighbor. In our case, the visibility graph will always be connected due to the fact that any given observation in a time-series are forming a node with a connected edge to the observation it immediately succeeds, and its immediate successor. Thus, it will be guaranteed that there exists a connection between all nodes in the graph. This leads to a situation where all nodes have at least two other nodes connected to it, except for the nodes created from the first and last observation in the time-series. They have at least one edge connected to them.

Secondly, the graph will always be undirected. This means that the edges in the graph have no separation in terms of which way they are going from one node to another. Simultaneously, it will consequently not discriminate between edges coming in and going out from a node. For a network generated from a visibility

graph, this will lead to a loss of the information about time. In difference from a time-series, the graph will have no information about the sequence the observations happened in, nor whether observation t_a or t_b happened first.

At last, the visibility criterion is invariant under affine transformation of the time-series data. This means that the original time-series can be rescaled both horizontally and vertically without the graph being affected. At the same time, it will also be possible to do certain translations of the data, such as adding a linear trend, without it having any effect.

The second algorithm we will apply in this thesis, the horizontal visibility graph, as first proposed by Luque et al. (2009), is a subgraph of the natural visibility graph. Here, two observations form directly connected nodes if, and only if, no intermediate observation is higher. Formally, we can write this as;

$$x_i, x_j > x_n, \qquad i > n > j \tag{10}$$

Here x_i and x_j represent any two given nodes in a time-series, while x_n represents every intermediate observation.

In Figure 3, we have given an example of how the observations in a time-series are formed into a network with the horizontal visibility graph algorithm. The time-series used here is the same as the one in Figure 1, but the algorithms applied in the two cases are obviously different. The concept is the same, but as we said, in this case, the criterion is only fulfilled if both x_i and x_j are higher than any intermediate node x_n.

Again we have transformed the time-series into an adjacency matrix and a network, as shown in Figure 4. The main thing to note between Figure 2 and Figure 4 is that the natural visibility graph algorithm clearly creates a network with more edges than the horizontal visibility graph algorithm does.

As for the natural visibility graph algorithm, the three main structures we mentioned earlier are also present in the horizontal visibility graph algorithm. For the horizontal visibility graph, we have the possibility to discriminate between nodes going in and out. In this case, every node would send out two edges, the out-degrees, one in each horizontal direction, and the recipients would get the in-degree. However, we have not seen this method being applied in the literature, and will stick to the original horizontal visibility graph, as it was proposed by Luque et al. (2009).

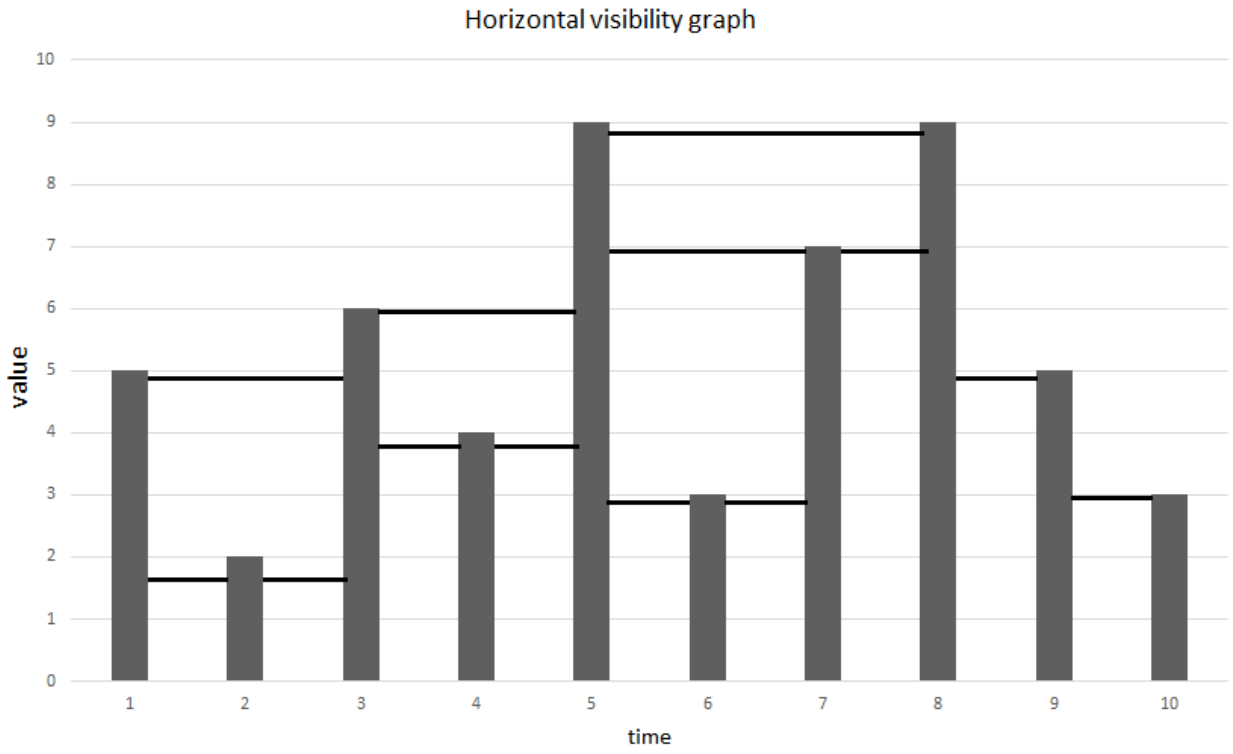

Figure 3: Example of the horizontal visibility algorithm applied to a time-series consisting of ten observations created by us.

We also mentioned earlier how every node has at least two edges connected to it, except for the first and last observation in the original time-series, which has at least one. This comes apparent in Figure 3 and 4, by looking at the node created from the 10th observation and comparing it to the other nine.

	1	2	3	4	5	6	7	8	9	10
1	0	1	1	0	0	0	0	0	0	0
2	1	0	1	0	0	0	0	0	0	0
3	1	1	0	1	1	0	0	0	0	0
4	0	0	1	0	1	0	0	0	0	0
5	0	0	1	1	0	1	1	1	0	0
6	0	0	0	0	1	0	1	0	0	0
7	0	0	0	0	1	1	0	1	0	0
8	0	0	0	0	1	0	1	0	1	0
9	0	0	0	0	0	0	0	1	0	1
10	0	0	0	0	0	0	0	0	1	0

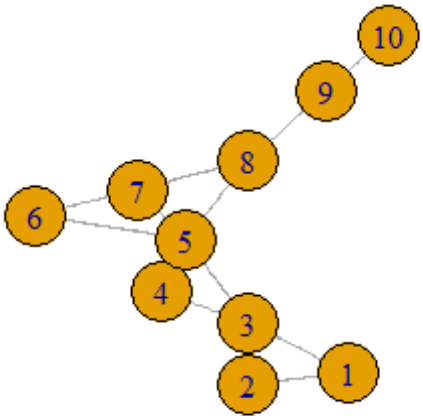

Figure 4: Adjacency matrix and graphical visualization for the network created from the time-series in Figure 3.

It is important to note that for both the natural visibility graph and the horizontal visibility graph, the length of the original time-series is important in evaluating the results obtained. As argued by Zhuang et al. (2014)

boundary effects can affect the properties of the visibility graph. We will also see later in this thesis, that some of the network statistics used require consistent time-intervals when used for comparative purposes. Lastly, it can be a challenge to find the length of a time-series that provides enough information to perform a visibility graph analysis, without making it so complex it becomes impossible to analyze. Rather than trying to find the optimal number of observations, we will make our decisions based on previous literature. In Zhuang et al. (2014), it was argued for the use of ten-year intervals when working on financial data, and we will stay consistent with that.

Based on the connections we are able to obtain in the original time-series, an adjacency matrix will be created. We briefly mentioned it further up, and as we saw in both Figure 2 and 4, this matrix will consist of zeros and ones only, where one will indicate that two nodes are directly connected by an edge, while a zero-value will indicate no such connection. Any given two nodes in the network will be presented in the adjacency matrix (A) the following way:

$$A_{ij} = \begin{cases} 1 \; if \; there \; is \; a \; connection \; between \; node \; i \; and \; j \\ 0 \; otherwise. \end{cases} \tag{11}$$

The adjacency matrix will be of form n by n, where n in this case is equal to the number of observations in the original time-series. The diagonal in the adjacency matrix will always consist of zeros, since we are operating with an algorithm where nodes are not connected to themselves. At the same time, we will always have ones in the items next to the diagonal, as a result of every observation being connected to its two closest neighbors. An adjacency matrix will in the case of undirected graphs always be symmetric, meaning that it looks the same above and below the diagonal. This means that A_{ij}, and A_{ji} always will be of the same value.

In Figure 5 we display a graphical interpretation of two different adjacency matrices generated with a time-series consisting of 250 observations from S&P 500 return-series. In the graph, a connection is represented with a black dot, similarly to how it was represented by a 1 in the previous example of adjacency matrices. The one on the left is for the natural visibility graph, while the one on the right is for the horizontal visibility graph. This figure nicely shows us two important aspects about the graphs we study. First of all, we see clearly from the areas around diagonal the effect of all observations being connected with its nearest neighbors, and that observations have a tendency to connect to other observations that lay close to them. Connections observed further from the diagonal are very sparse. However, these observations are important as they often are related to the hubs of the network.

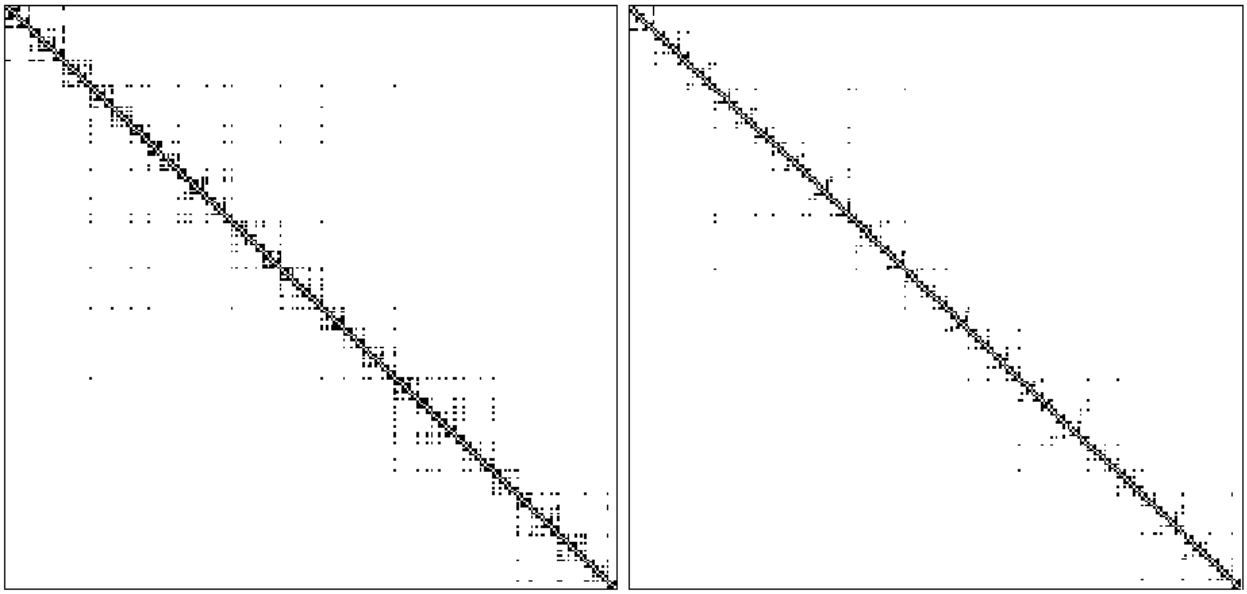

Figure 5: The adjacency matrix created from a natural visibility graph (left) and horizontal visibility graph (right), from a return series consisting of 250 observations from S&P 500.

Secondly, we see an obvious difference between the natural visibility graph and the horizontal visibility graph in regards to the amount of edges created. The connections from the natural visibility graph undoubtedly spreads more out in the adjacency matrix, and we can see that there are a lot more nodes with a high degree. This is similar to what we saw in the examples of the time-series we used as an example above.

While the adjacency matrix is a mathematical representation of the graph from a visibility graph, we can also exhibit it as a graphical figure. In Figure 6, we show how the same network as in Figure 5, but this time graphically. Again, we are showing the natural visibility graph on the left and the horizontal visibility graph on the right. As previously, we can see two main things: firstly, nodes are often connected to its nearest neighbors, and these connections create so-called neighborhoods, which are a heavy clustering of parts of the nodes in a graph. Secondly, we also here see a significant difference in the amount of edges between the two algorithms. In this visualization we have kept the nodes constant, and it helps to show how the horizontal visibility graph is a subset of the natural visibility graph. Every edge you can find in the horizontal visibility graph is also present in the natural visibility graph.

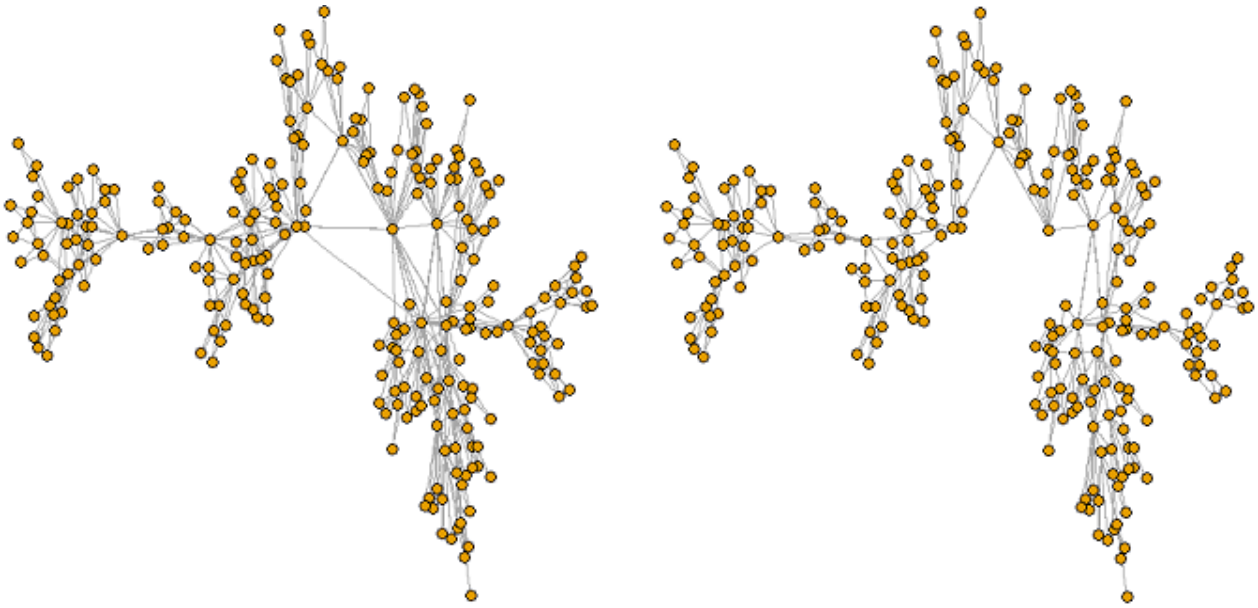

Figure 6: Natural visibility graph (left) and horizontal visibility graph (right) of a return series consisting of 250 observations from S&P 500.

Additionally, we have in Figure 7 plotted similar networks as above, this time both are generated from the S&P 500 raw price-series. This shows the significant differences we obtain by using these two algorithms for different types of processes. The natural visibility graph (left) has a substantially larger amount of edges than the case was for the return-series. Because of the time-trend we often see in raw financial data, the visibility criterion is more easily and more often fulfilled than in the case of return-series. For the horizontal visibility graph, it visually looks like we have a similar situation with more edges for the raw series than the return-series, but as we will show in Section 5 this is actually not the case. What happens is that in addition to connection to its nearest neighbors, nodes in a horizontal visibility graph often connects with nodes far from itself in time. When we look at the visibility graph in regards to raw prices, we will often get highly clustered neighborhoods, created between the top of one price-cycle and the top of the next. These cycles can easily be seen by looking at the most clustered areas of the graphs below.

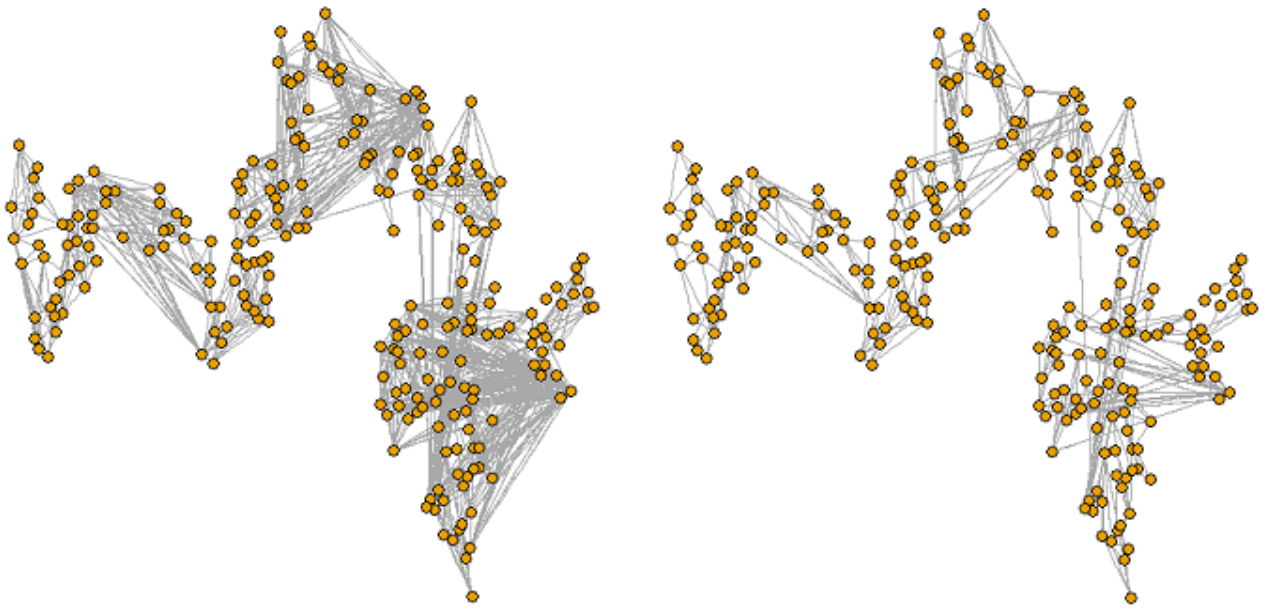

Figure 7: Natural visibility graph (left) and horizontal visibility graph (right) of a price series consisting of 250 observations from S&P 500. The time-frame is the same as in the return-series above.

3.4 Network statistics

As we briefly mentioned in previous sections, we will look at five main network statistics for complex networks. These five are degree distribution, assortativity, local and global transitivity (clustering coefficient) and average shortest path length. The networks statistics chosen were picked due to the same five network statistics being used by Zhang et al. (2017). Some of these network statistics have a fundamental standing in the complex network theory, and will be seen in virtually every study within this field, while others are of relatively less importance. In general, there exists nearly an unlimited amount of statistics than can be applied to a network, but in the context of our research problem, it seems reasonable to limit ourselves to the five network statistics mentioned above.

3.4.1 Node degree

The degree of a node in a network is the amount of other nodes it is connected to (Newman, 2010, p. 133). A connection between two nodes is called an edge, and it is the amount of edges it has connected to it that makes up a node's degree. In a directed graph, we would separate between incoming and outgoing edges, but in undirected graphs, as we will be working with, the degree is just the total number of edges connected to a node. If we denote k_i as the degree of any given node, we can express this in term of the adjacency matrix A_{ij} in the following way:

$$k_i = \sum_{j=1}^{n} A_{ij} \tag{12}$$

In a network derived from a time-series, all nodes except for those from the first and last observation will have at least two edges connected to it. Most nodes in a network tend to have a relatively small number of edges, and for a graph derived from a time-series this will mean a value not much larger than two. From time to time we see single nodes with a very large degree. These nodes are called hubs, and is often found to have a significant influence on the network's structure and statistical measures.

A node's degree can be described as a local network statistic, since it is looking at every node's degree in isolation to the rest. However, we are mainly working with global network statistics in this paper, and will use the arithmetic mean of the node degrees in our presentations.

The degree of the nodes in a graph can also be used to form the graph's degree distribution, one of the most defining statistics of a network's structure. If k_i is the degree of a given node in the network, a degree density p(k) will give the fraction of nodes in the network with degree k. Consequently, the degree distribution is given as $\mathrm{F(K)} = \mathrm{p(k < K)}$. As previously stated, a node's probability of having a low degree tends to be high, and it has been shown that p(k) becomes lower as k grows. We will show examples of this for a network generated form S&P 500 in Section 6, although degree distribution is not the most important aspect of this thesis. The degree distribution is often suggested as a possible tool for determining what type of network one is dealing with. Barabási and Albert (1999) showed that the degree distribution of large networks tends to converge to a power law, and that if the degree distribution follows a power law, the network can be described as scale-free. Lacasa and Toral (2010) stated that one can differentiate between chaotic and correlated stochastic processes with help from the horizontal visibility graph in instances where the degree distribution followed an exponential distribution.

3.4.2 Assortativity

Assortative mixing is a preference for nodes to connect with other nodes that are of similar nature in some way. In, for example, a social network this may be that people connect with people they find similar to themselves whether the similarity is of age, gender, race, income, interests or any other characteristic. One could also have disassortative mixing, which is when nodes tend to connect with nodes that are dissimilar to themselves. Assortative mixing may be examined by a variety of enumerative characteristics, but most network theorists choose to examine assortativity by node degree (Newman, 2003). Assortative mixing by degree is a special case of a scalar characteristic. It examines if nodes of a high degree tend to connect with other nodes of a high degree, and nodes of a low degree with other nodes of a low degree. The reason why this method is interesting is that a node's degree in itself is a property of the network structure.

Assortativity by degree is calculated by the assortativity coefficient r, which is the Pearson correlation coefficient of degree between pairs of linked nodes (Newman, 2002). When A_{ij} is a given observation in the adjacency matrix, k_i and k_j represents the degree of two arbitrary nodes, δ_{ij} is the ratio between nodes and edges between a pair of nodes and m is the number of edges in the network. We can write the formula for the assortativity coefficient in the following way:

$$r = \frac{\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)k_i k_j}{\sum_{ij}\left(k_i \delta_{ij} - \frac{k_i k_j}{2m}\right)k_i k_j} \tag{13}$$

The values of r lie in the interval [-1,1] where a positive value indicates assortative mixing and a negative value shows disassortative mixing. If the value is 0 the network is non-assortative or in other words random in regards to assortativity.

Newman (2002) examined multiple real world networks for the assortativity coefficient and found that while social networks normally were assortative, technical and biological networks often were disassortative. Newman (2002) also found that networks modelled according to the random graph theory of Erdos and Rényi (1960) and the scale-free model of Barabási and Albert (1999), had a coefficient of zero. The latter was deemed as a bit of a surprise, since scale-free networks often had been used to model networks with disassortative structures.

A reason for this disassortative mixing can be a function of preferential attachment that often is found in networks. New nodes have a tendency to connect to already high degree nodes and this can cause

disassortative mixing throughout the network (Barabási & Albert, 1999). Although there has been done some research on assortative and disassortative mixing in a variety of networks, little research has been done in regards to network generated from economic and financial time-series, and it will therefore be hard to conclude on whether the results found theoretically are making sense or not. However, based on how the visibility graph is generated, where the hubs have good visibility over the time-series, and consequently often create an edge to other hubs in the network, we do expect most financial time-series to give networks with a positive assortativity-coefficient.

3.4.3 Local and global transitivity

Transitivity, often also called clustering coefficient in the literature, is a statistic that measures whether a node y and another node z are connected, when we already know that both y and z are connected to a third node x. Although there are several different interpretations of how to calculate the transitivity of a network we will stick to the interpretation mentioned above, which is also used by (Newman, 2010, p. 198). These interpretations assume that the graph is undirected, as is the case for visibility graphs. Two different measures will be used, global and local transitivity.

The global transitivity refers to the transitivity for the network as a whole. A coefficient between 0 and 1 tells us about the probability of a given node and two of its connections being part of a triplet. Described in other words, what we measure is the ratio of connected edges that also form a triplet. We can write the formula for global transitivity as:

$$C = \frac{number\ of\ closed\ paths\ of\ length\ two}{number\ of\ paths\ of\ length\ two} \tag{14}$$

Another way of writing it is the following:

$$C = \frac{(number\ of\ closed\ trianges\ in\ the\ network) * 3}{number\ of\ connected\ triples} \tag{15}$$

Here a connected triple is three nodes that are connected with at least two edges. The number of triangles needs to be multiplied by three because every edge in the triangle represents a unique closing of three connected nodes, even though it is the same three nodes in all three instances.

In the case of a coefficient value of 1, we have what is called perfect transitivity. For this to be possible, every component of the network is a closed clique. Obviously, the probability of this happening in large networks are very small, to say the least. On the other hand, for a network to have the coefficient of 0, there would have to be no closed triplets of observations in the network. As for the case of a coefficient value close to 1, this is unlikely when working with networks from time-series. Although obtaining the polar-values are highly unlikely, we can find values close to them in, for example, a fractional Brownian motion type process with Hurst exponent close to either 0 or 1.

Unlike the global transitivity, the local transitivity looks at single nodes, and the probability that two of its i connected nodes are also connected.

$$C_i = \frac{numbers\ of\ paris\ of\ neighors\ of\ i\ that\ are\ connected}{number\ of\ pairs\ of\ neighbors\ of\ i} \tag{16}$$

For a lot of networks, it is shown that the local transitivity has a rough negative dependence on the degree of a node (Newman, 2010, p. 265). Nodes with high degree tend to have a low local transitivity and the other way around.

In this paper, we will similarly to Zhang et al. (2017), use the statistic mean local transitivity. This simply means that we take the arithmetic mean of each node's local transitivity and create a global network measure out of it.

Let it be said that we believe this statistic to have some clear weaknesses, but have included it to stay consistent with the mentioned study from Zhang et al. (2017). The reason why we lack faith in it, is because in its original form the local transitivity of a node is given as a percentage value. This percentage value loses its meaning when it is no longer tied up to the degree of the node it represents, as will be the case when we find the arithmetic mean. As an example, when we find the mean local transitivity, we fail to differentiate between a node with degree 10 and a 0.25 local transitivity value, and a node with degree 20 and a 0.25 local transitivity value. This is obviously problematic, since it overlooks an important part of information about the network. For a financial comparison, it is like investing money in two assets with respectively a 10 % and 20 % return, and thinking that your payoff is 15 %, without actually considering how much money you invested in each of the two assets.

3.4.4 Average shortest path length

A shortest path, also called geodesic path, is a path between two nodes such that no shorter path exists (Newman, 2010, p. 139). The average of the shortest path is used as a measure of how effectively information is transported throughout the network. A path in a network is the road along edges that connects a pair of nodes and the length of a path is the number of edges that must be passed from one node to another. In a directed network, the path must be traversed in a given direction, and in some networks there may not always be a path that exists between any given pair of nodes. However, we know by definition that the visibility algorithm gives us a network that is both connected and undirected, which means that there exists a path between all pairs of nodes and the direction of the path does not matter. There are several different types of paths, but we decide to focus on the geodesic path, which is the most relevant to the visibility algorithm.

The length of a geodesic path between two nodes equals the amount of edges that the path goes through between the pair. Thus, the average shortest path length for a connected and undirected network can be defined as followed: Consider a network G with a set of nodes N. Define the shortest path between a pair of nodes as $sp(n_i, n_j)$ and $path(n_i, n_j)$ as a path between a pair of nodes.

$$ASPL_G = \frac{\sum_{i,j}^{N} sp(n_i, n_j)}{\sum_{i,j}^{N} path(n_i, n_j)} \tag{17}$$

To get the average shortest path length we have to take the value of the shortest path length between all pair of nodes in the network and divide it by the value of all existing paths in the network. For a connected and undirected graph, as the visibility graph, $\sum_{i,j}^{N} path(n_i, n_j) = N(N-1)$ because there exists a path between any pair of nodes in the network (Mao & Zhang, 2013). Therefore, for a visibility graph we get the following equation for average shortest path length:

$$ASPL_G = \frac{\sum_{i,j}^{N} sp(n_i, n_j)}{N(N-1)} \tag{18}$$

The length of the average shortest path length will normally be dependent on especially two things. The first is obviously the amount of edges in the network. Many edges give us more sets of nodes that are connected, and where the path length is only one, which makes the average shortest path length shorter. The second phenomenon that affects the path length of a network are structural holes. Structural holes are nodes, often a hub, that controls the flow of information in a network, in that a high fraction of the

paths has to go through it. As an example, for a financial return-series plotted into a visibility graph, this will often correspond to incidents where the return is very high compared to the mean.

The value of the average shortest path length gives us some indications of what type of network we are dealing with. Random networks tend to have a higher average shortest path length than the networks mentioned below because there is no order in the pattern of the network, which usually leads to shorter distances between a pair of nodes in contrast to non-random networks. Small-world networks will typically have a low average shortest path length, $L \sim \ln(N)$ (Watts & Strogatz, 1998) and scale-free networks will typically have an even lower average shortest path length, $L \sim \ln\ln(N)$ (Cohen & Havlin, 2003).

Previously in this section, we showed some networks generated with short time-series from S&P 500. As these were just examples, we did not further discuss the underlying data used to create these network, as well as the data's source. This will obviously have to be done for the actual studies, and in Section 4 we will present the data we have used to obtain the results given in Section 5.

4. Data

4.1 S&P 500

We will apply the natural visibility graph and horizontal visibility graph algorithm on subsets of the S&P 500 portfolio. The data we have used are available from Yahoo Finance.[1] We will use the adjusted closing price, both in its raw nature, and as regular return-prices. Rather than looking at all the available data in the time-series as a whole, we have divided the data into ten subsets of 2500 observations, with a rolling start for each 1250th observation. This makes the subsets overlap each other, such that every observation of the original time-series is present in exactly two of the ten subsets.

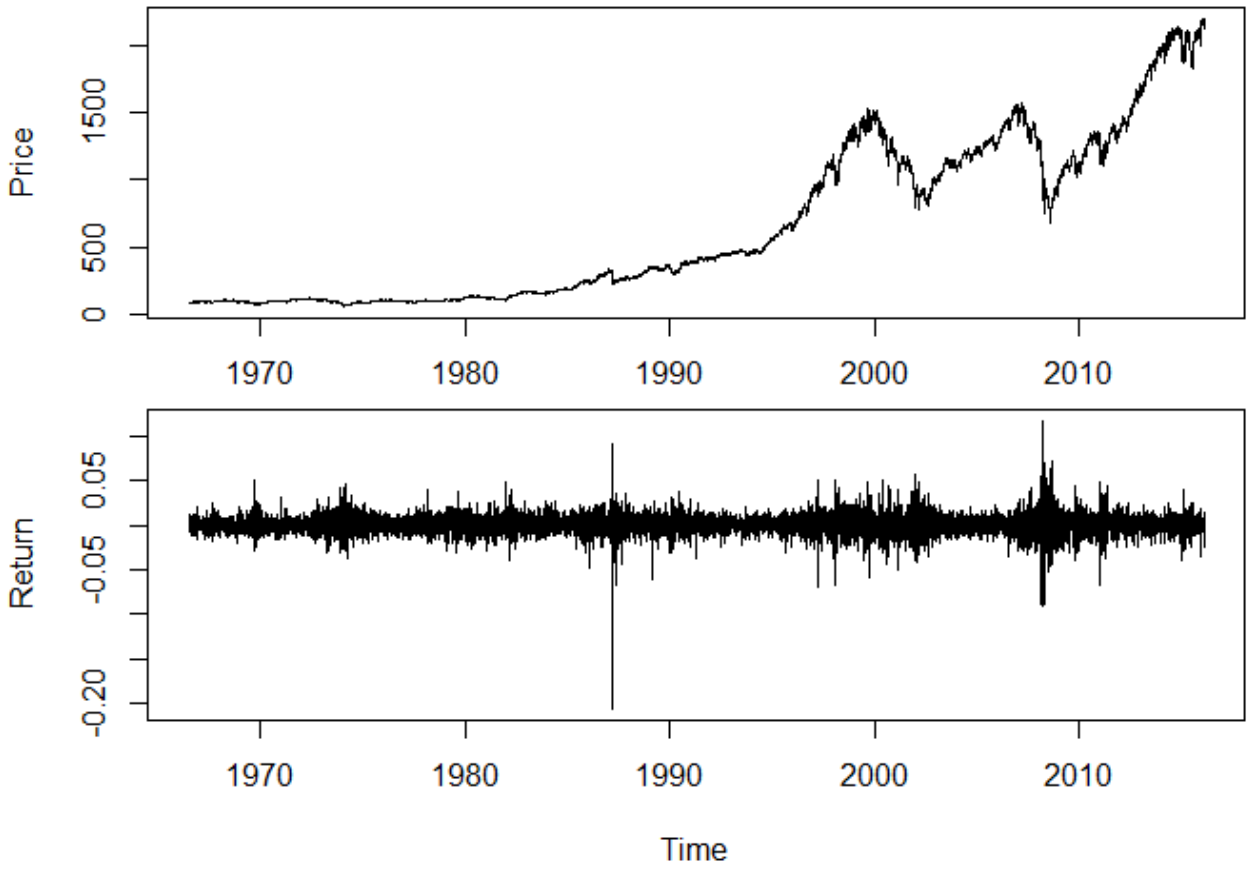

Figure 8: Prices (upper) and returns (lower) for S&P 500 from 1967 to 2016.

[1] https://finance.yahoo.com/quote/%5EGSPC/history?period1=-157770000&period2=1454194800&interval=1d&filter=history&frequency=1d

In a regular year there are normally between 250 and 252 business days, so for daily prices, 2500 observations will equal approximately 10 years of data. The last observation used is the last business day of 2016, more precisely December 30th. As some of the network-statistics used are dependent on the amount of observations in it, we find it important out of consistency purposes to keep every time-series of equal length, which is why we have chosen to use 2500 observations rather than ten years exactly.

The full time-series as well as the return-series can be seen in Figure 8. As we can see, the raw time-series has neither constant mean, nor variance. It can therefore be an interesting study to apply the visibility graph algorithms to it, and to compare the findings to those of the return-series. The return-series show signs of constant mean, but bursts of volatility clustering, especially centered around the different financial crises such as Black Monday in 1987, the dot com-bubble around year 2000, and the global financial crisis of 2007-2008. The return-series is not available from S&P 500, and needs to be calculated with the following formula

$$r_t = \frac{p_t - p_{t-1}}{p_{t-1}} \tag{19}$$

The descriptive statistics of the raw series and the return-series can be seen in Table 1 and Table 2 below. As is expected, these tables show us quantitatively the difference between the two time-series. Skewness and kurtosis is only calculated for the return-series, since the observations in the price-series obviously is nowhere near following a normal distribution. The kurtosis in Table 2 is calculated under the assumption that a value of 3 is the normal distribution, in other words, the kurtosis is not adjusted to show an excess value.

	Prices S&P 500			
	mean	sd	min	max
1962-1971	87.006	12.502	52.32	110.46
1967-1976	96.110	10.616	62.28	120.24
1972-1981	103.371	14.769	62.28	140.52
1977-1986	145.740	48.301	86.90	301.64
1982-1991	252.376	83.749	102.42	420.77
1987-1996	421.045	126.670	223.92	816.29
1992-2001	855.114	370.498	394.50	1527.46
1997-2006	1158.326	177.480	737.65	1527.46
2002-2011	1166.644	183.623	676.53	1565.15
2007-2016	1516.990	394.627	676.53	2271.72

Table 1: Descriptive statistics for the price-series of S&P 500.

	Returns S&P 500					
	skew	kurt	mean	sd	min	max
1962-1971	-0.08	11.42	0.00	0.01	-0.07	0.05
1967-1976	0.32	5.49	0.00	0.01	-0.04	0.05
1972-1981	0.17	4.51	0.00	0.01	-0.04	0.05
1977-1986	0.13	4.87	0.00	0.01	-0.05	0.05
1982-1991	-2.82	60.65	0.00	0.01	-0.20	0.09
1987-1996	-3.89	87.73	0.00	0.01	-0.20	0.09
1992-2001	-0.21	7.49	0.00	0.01	-0.07	0.05
1997-2006	0.02	5.99	0.00	0.01	-0.07	0.05
2002-2011	0.03	11.35	0.00	0.01	-0.09	0.12
2007-2016	-0.09	13.07	0.00	0.01	-0.09	0.12

Table 2: Descriptive statistics for the return-series of S&P 500.

4.2 GARCH-simulation

As discussed in Section 3, the most important portion of data in this thesis will be generated with simulation from a GARCH(1,1)-process. As previously stated, a GARCH(1,1)-process consist of three parameters, the constant α_0, a parameter α_1, performing an auto regression of the previous error-variance, and a parameter β_1, performing a moving average process on the previous observations of conditional variance. The goal for this paper is to see the effects of visibility graphs on financial time-series, and consequently, we want our simulations to be as close to real-life financial time-series as possible. In order to achieve this, we need parameters that are representative for a typical financial time-series, and have decided to estimate GARCH-parameters from the S&P 500 time-series. We will use the same time-series as we showed in section 5.3, and more precisely, we chose to estimate the parameters from the most recent of our ten subsets, the one stretching from 2007 to 2016.

A more detailed plot of the time-series we have estimated our GARCH-parameters from can be seen in figure 9. There is clear evidence of volatility clustering in this data set, as we can see several bursts of increased volatility compared to the mean. Especially the period around the 500th observation is glaring. This obviously corresponds to the global financial crisis of 2007/2008.

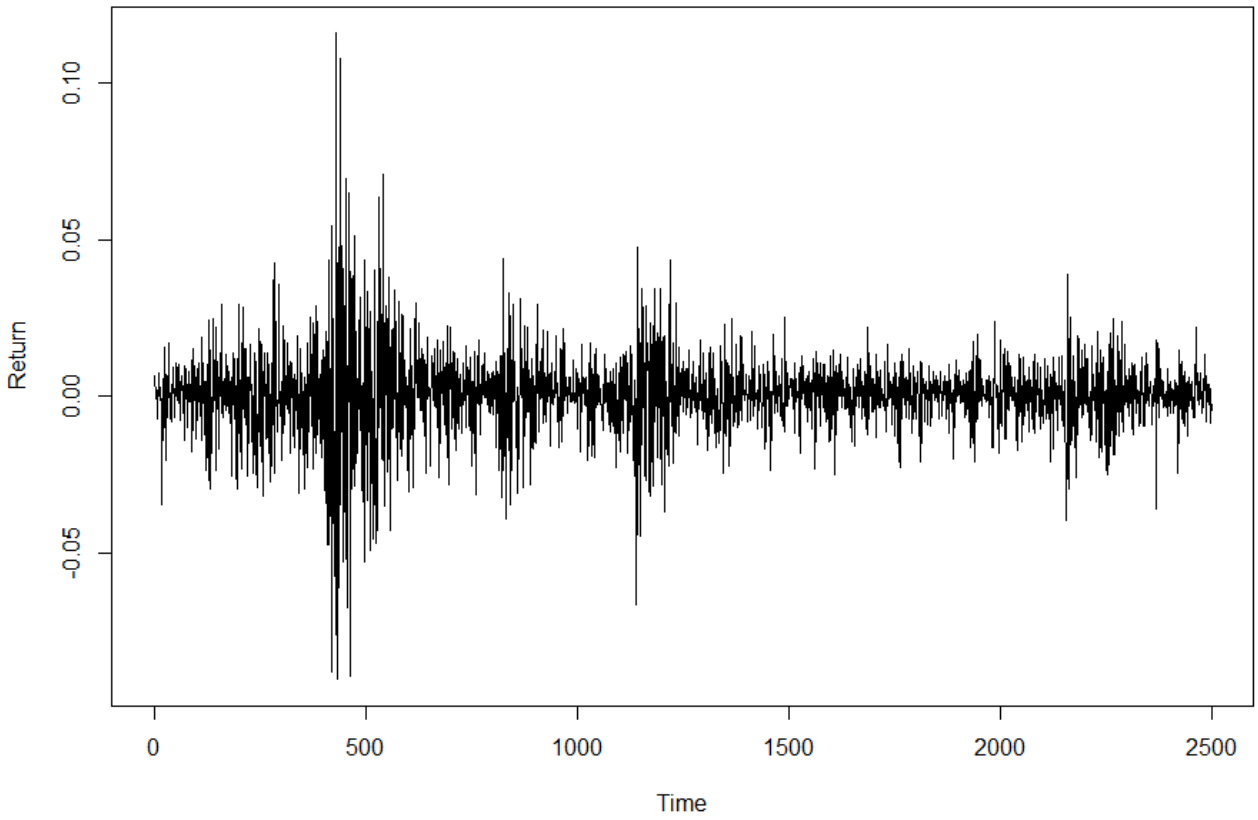

Figure 9: Time-series plot of S&P 500 returns (2007-2016)

	Test statistics			
	Estimate	Std Error	t-value	p > |t|
α_0	$2.938 * 10^{-6}$	$5.290 * 10^{-6}$	5.553	0.00
α_1	0.1169	$1.328 * 10^{-2}$	8.802	0.00
β_1	0.8612	$1.429 * 10^{-2}$	60.628	0.00

Table 3: Estimated parameters and test statistics for the return-series from S&P 500, stretching from 2007 to 2016.

The estimated parameters for the GARCH(1,1)-process can be seen in Table 3 above, together with the associated test statistics. As mentioned, these parameters will be used to simulate data, to which we will apply the visibility graph and horizontal visibility graph. In addition, we will investigate other possible time-series from GARCH within the given three-dimensional parameter space. As discussed in Section 3, there are some boundaries to the parameters. We will not consider parameters of negative value, and we always have to make sure that $\alpha_1 + \beta_1 < 1$. Lastly, we choose to keep the constant α_0 fixed throughout all our simulations. Consequently, our remaining parameters will come from a two-dimensional space bounded by the x-axis, the y-axis and the line $y = -x + 1$. In total, we have chosen to investigate six different time-series of varying parameter-combinations where α_1 and β_1 are both positive and seven simulations where one or both of the two parameters are set to zero. This gives us a simulation strategy consisting of thirteen total simulations, as portrayed by Table 4.

	α_0	α_1	β_1	$\alpha_1 + \beta_1$
Case 1: $low\ \alpha_1,\ high\ \beta_1$	$2.938 * 10^{-6}$	0.1169	0.8612	0.9781
Case 2: $low\ \alpha_1,\ medium\ \beta_1$	$2.938 * 10^{-6}$	0.1169	0.4755	0.5924
Case 3: $low\ \alpha_1,\ low\ \beta_1$	$2.938 * 10^{-6}$	0.1169	0.1169	0.2338
Case 4: $medium\ \alpha_1,\ low\ \beta_1$	$2.938 * 10^{-6}$	0.4755	0.1169	0.5924
Case 5: $high\ \alpha_1,\ low\ \beta_1$	$2.938 * 10^{-6}$	0.8612	0.1169	0.9781
Case 6: $medum\ \alpha_1,\ medium\ \beta_1$	$2.938 * 10^{-6}$	0.4755	0.4755	0.9510
Case 7: $zero\ \alpha_1,\ high\ \beta_1$	$2.938 * 10^{-6}$	0	0.1169	0.1169
Case 8: $zero\ \alpha_1,\ medium\ \beta_1$	$2.938 * 10^{-6}$	0	0.4755	0.4755
Case 9: $zero\ \alpha_1,\ low\ \beta_1$	$2.938 * 10^{-6}$	0	0.8612	0.8612
Case 10: $zero\ \alpha_1,\ zero\ \beta_1$	$2.938 * 10^{-6}$	0	0	0
Case 11: $low\ \alpha_1,\ zero\ \beta_1$	$2.938 * 10^{-6}$	0.1169	0	0.1169
Case 12: $medum\ \alpha_1,\ zero\ \beta_1$	$2.938 * 10^{-6}$	0.4755	0	0.4755
Case 13: $high\ \alpha_1,\ zero\ \beta_1$	$2.938 * 10^{-6}$	0.8612	0	0.8612

Table 4: Design points of the simulation study

As we can see from Table 4, we have decided upon a simulation strategy where we use large variations in both α_1 and β_1. This way, we are able to cover the whole parameter space in a good way, without ever leaving the boundaries. We have decided upon a strategy where we only use four different values for both of the parameters. The clear advantage in only changing one parameter at the time is that we with certainty can contribute changes in the estimated network statistics to the change in parameter-value.

The combinations given in table 4 is also shown graphically in Figure 10, together with the constraint line $y = -x + 1$. The results obtained with the data presented in this section will be given in Section 6. First for the simulation study, and then for the data from S&P 500.

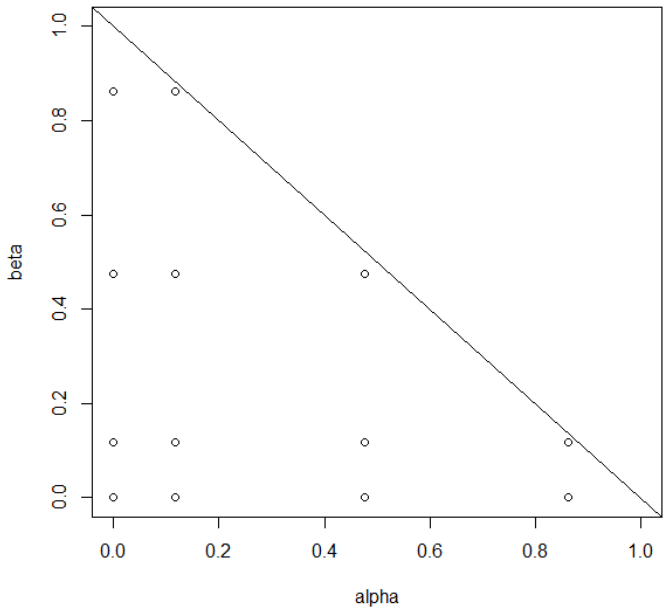

Figure 10: Graphical display of design points for the simulation study

5. Results

5.1 Simulations

Case 1	α_1 = 0.1169, β_1 = 0.8612					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.013	5.100	$1.5*10^{-6}$	0.011	0.058	-0.060
Standard deviation	0.152	1.750	$2.3*10^{-4}$	0.001	0.016	0.019
Confidence interval	±0.030	±0.034	$\pm4.6*10^{-5}$	±0.000	±0.003	±0.004
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	10.878	0.154	0.339	0.122	3.984	
Standard deviation	1.051	0.010	0.002	$1.6*10^{-4}$	0.003	
Confidence interval	±0.206	±0.002	$\pm4.7*10^{-4}$	$\pm3.2*10^{-5}$	±0.001	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.452	0.229	0.357	0.154	6.107	
Standard deviation	0.257	0.025	0.010	$5.8*10^{-4}$	0.085	
Confidence interval	±0.050	±0.005	±0.002	$\pm1.1*10^{-4}$	±0.017	

Table 5: Simulation case 1: α_1 = 0.1169 and β_1= 0.8612

Case 2	α_1 = 0.1169, β_1= 0.4755					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	0.003	3.12	$-4.0*10^{-6}$	0.003	0.010	-0.010
Standard deviation	0.043	0.15	$5.4*10^{-5}$	$5.1*10^{-5}$	0.001	0.002
Confidence interval	±0.008	±0.030	$\pm1.1*10^{-5}$	$\pm9.9*10^{-6}$	$\pm2.3*10^{-4}$	$\pm2.9*10^{-4}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.287	0.164	0.337	0.122	3.986	
Standard deviation	0.551	0.011	0.003	$1.5*10^{-4}$	0.003	
Confidence interval	±0.108	±0.002	$\pm5.0*10^{-4}$	$\pm3.1*10^{-5}$	$\pm5.4*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.688	0.194	0.376	0.154	5.947	
Standard deviation	0.211	0.015	0.006	$7.0*10^{-4}$	0.066	
Confidence interval	±0.041	±0.002	±0.001	$\pm1.4*10^{-4}$	±0.013	

Table 6: Simulation case 2: α_1 = 0.1169 and β_1= 0.4755

Case 3	α_1 = 0.1169, β_1= 0.1169					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.004	3.084	$-1.1*10^{-6}$	0.002	0.007	-0.007
Standard deviation	0.061	0.122	$3.4*10^{-5}$	$3.3*10^{-5}$	$7.2*10^{-4}$	$8.9*10^{-4}$
Confidence interval	±0012	±0.024	$\pm6.7*10^{-6}$	$\pm6.4*10^{-6}$	$\pm1.4*10^{-4}$	$\pm1.8*10^{-4}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.231	0.167	0.337	0.122	3.986	
Standard deviation	0.480	0.010	0.003	$1.9*10^{-4}$	0.003	
Confidence interval	±0.094	±0.002	$\pm5.1*10^{-4}$	$\pm3.7*10^{-5}$	$\pm5.7*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.709	0.192	0.376	0.154	5.925	
Standard deviation	0.226	0.014	0.006	$6.5*10^{-4}$	0.061	
Confidence interval	±0.044	±0.003	±0.001	$\pm1.2*10^{-4}$	±0.012	

Table 7: Simulation case 3: α_1 = 0.1169 and β_1= 0.1169

Case 4	α_1 = 0.4755, β_1= 0.1169					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	0.007	6.841	$1.2*10^{-6}$	0.003	0.017	-0.017
Standard deviation	0.315	3.754	$5.5*10^{-5}$	$1.3*10^{-4}$	0.006	0.005
Confidence interval	±0.062	±0.736	$\pm1.1*10^{-5}$	$\pm2.5*10^{-5}$	±0001	±0.001
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.830	0.137	0.338	0.122	3.985	
Standard deviation	0.582	0.011	0.002	$1.8*10^{-4}$	0.003	
Confidence interval	±0.114	±0.002	$\pm4.6*10^{-4}$	$\pm3.5*10^{-5}$	$\pm5.9*10^{-5}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.290	0.157	0.353	0.156	6.283	
Standard deviation	0.254	0.028	0.015	$6.4*10^{-4}$	0.117	
Confidence interval	±0.498	±0.006	±0.003	$\pm2.3*10^{-4}$	±0.023	

Table 8: Simulation case 4: α_1 = 0.4755 and β_1= 0.1169

Case 5	α_1 = 0.8612, β_1= 0.1169					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.180	76.966	$1.5*10^{-6}$	0.006	0.081	-0.086
Standard deviation	4.214	99.083	$1.4*10^{-4}$	0.002	0.044	0.083
Confidence interval	±0.926	±19.420	$\pm2.8*10^{-5}$	$\pm4.4*10^{-4}$	±0.009	±0.016
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	11.063	0.106	0.340	0.122	3.984	
Standard deviation	0.905	0.012	0.002	$1.7'10^{-4}$	0.003	
Confidence interval	±0.178	±0.002	$\pm4.5*10^{-4}$	$\pm3.3*10^{-5}$	$\pm6.8*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	4.635	0.041	0.283	0.158	7.195	
Standard deviation	0.346	0.040	0.041	$9.1*10^{-4}$	0.297	
Confidence interval	±0.068	±0.008	±0.008	$\pm1.7*10^{-4}$	±0.058	

Table 9: Simulation case 5: α_1 = 0.8612 and β_1= 0.1169

Case 6	α_1 = 0.4755, β_1= 0.4755					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.014	27.729	$1.7*10^{-6}$	0.007	0.072	-0.072
Standard deviation	1.476	34.195	$1.6*10^{-4}$	0.003	0.051	0.073
Confidence interval	±0.289	±6.702	$\pm3.1*10^{-5}$	$\pm5.4*10^{-4}$	±0.010	±0.014
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	11.479	0.122	0.342	0.122	3.983	
Standard deviation	0.909	0.012	0.002	$1.7*10^{-4}$	0.004	
Confidence interval	±0.178	±0.002	$\pm4.2*10^{-4}$	$\pm3.2*10^{-5}$	$\pm7.1*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	4.919	0.112	0.315	0.156	6.814	
Standard deviation	0.324	0.057	0.040	$6.7*10^{-4}$	0.266	
Confidence interval	±0.064	±0.011	±0.008	$\pm1.3*10^{-4}$	±0.052	

Table 10: Simulation case 6: α_1 = 0.4755 and β_1= 0.4755

In table 5-10 we have given a quantitative summary of the results obtained by our simulation study. Each table includes descriptive statistics and network statistics for both the horizontal and natural visibility graph. As previously noted, each simulation was replicated 100 times, and the estimates presented here are the maximum likelihood estimate of these replications. In addition, the standard deviation of all the statistics are presented, as well as a confidence interval for the arithmetic mean. An interval estimate can easily be found from the point estimate and confidence interval, as described in section 3.2.

First off all, we observe that in the case of the horizontal visibility graph, the mean degree is very close to four in all cases. This is something that is expected, as the mean degree of a network generated with a horizontal visibility graph never can exceed four, an easy calculation, given that all nodes sends out two edges, and receives an average of close to two edges.

Further, we see that the mean local transitivity returns the same or nearly the same estimate for all six of the simulations for both graphs. This solidifies our initial belief of the weakness of this statistic, but we did not expect it to return such similar values. The same thing can also be seen for the global transitivity statistic for the horizontal visibility graph, while we see bigger variations when the statistic is estimated for the natural visibility graph.

The most interesting and intriguing results from these simulations are the relationship between the kurtosis of the time-series, the average shortest path length, and the mean degree. We previously discussed that there for obvious reasons often can be a dependence between path length and mean degree, but will also show in the S&P 500-case that mean degree can vary significantly between networks with relatively similar average shortest path length. However, here, we observe that if you rank the estimated statistics of the six simulations from highest to lowest kurtosis, from highest to lowest mean degree and from lowest to highest average path length, you would get them ranked in the exact same order. In addition, we also observed that the estimated values we obtained showed signs of dependence on the parameters used in the simulations, and where in the previously shown parameter-space they were laying.

Similar tendencies were also seen in the cases for both assortativity and global transitivity, without them being as strict for these cases. For both the assortativity and the global transitivity we observed that their estimated values decreased when α_1 was increased, while we observed less change in the estimated value by changes in β_1.

In addition to Case 1-6 presented above, we simulated seven more cases with different parameter combinations (tables can be seen in the appendix). Case 7-13 were all simulations where one or both of the parameters were set to zero. To get a look at how all the simulations compare to each other, and how they depend on the GARCH-parameters used, we have in Figure 11 plotted them all in four separate graphs, with the value of α_1 on the y-axis, the value of β_1 on the x-axis, and a point-estimate as the label. From graphs generated with the natural visibility graph algorithm has been done for average shortest path length, mean degree, assortativity and transitivity.

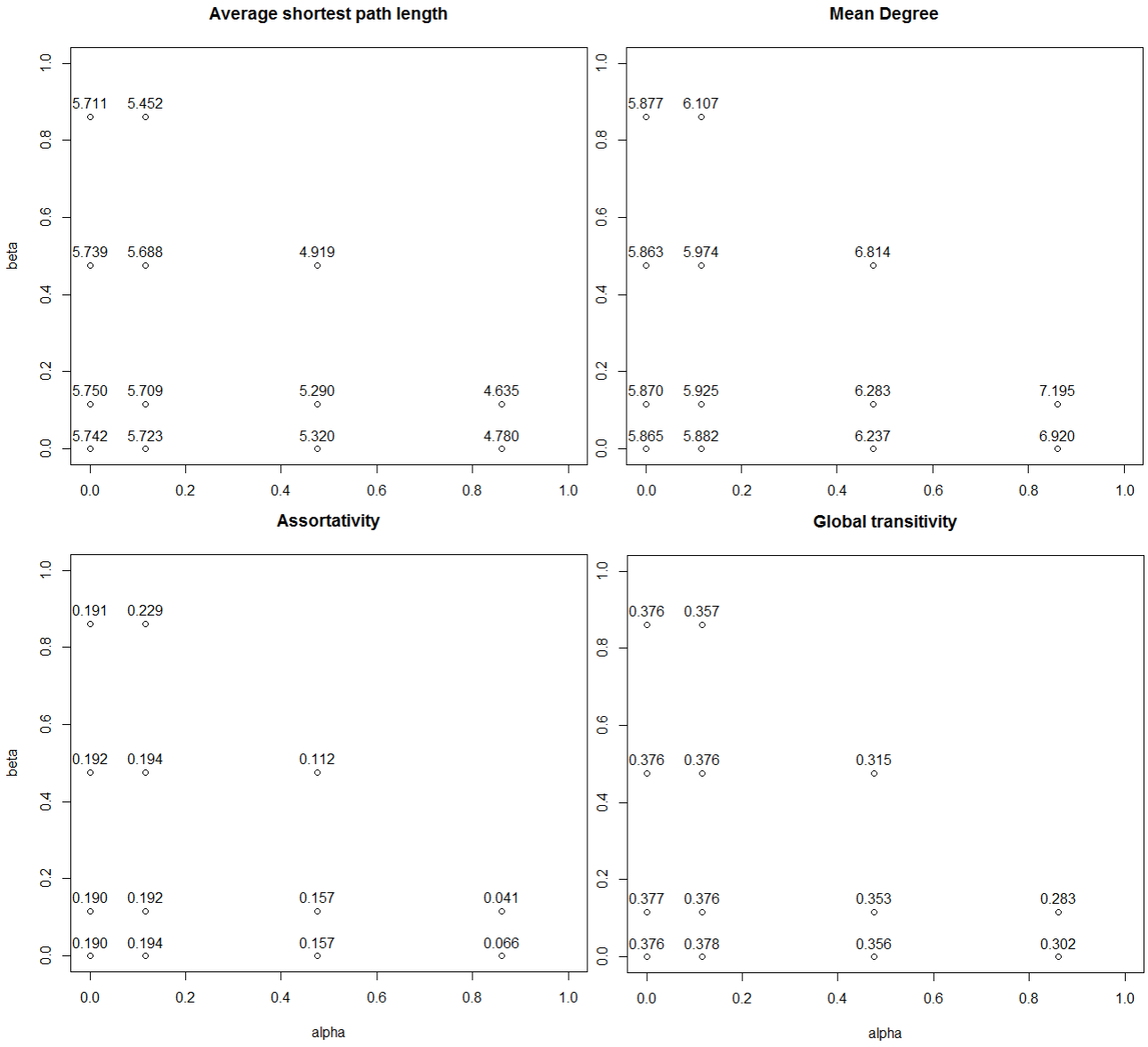

Figure 11: Plots showing the estimated value of the average shortest path length, mean degree, assortativity and global transitivity for all thirteen simulations with the natural visibility graph.

As we can see in Figure 11, the signs we saw in case 1-6 (table 5-10) of dependence between location in the parameter space and the estimated value of the different statistics is only solidified with the seven additional simulations. For average shortest path length, the graph in Figure 11 shows clear signs of a dependence between the parameters used in the simulation and the estimated network statistics obtained from these simulation cases. We see that the average shortest path length is at its longest when both parameters are set at zero, or close to zero. Moving out of the graphs origin, we see a significantly lower path length when α_1 is increased, and also some changes with increased β_1, however not as substantial. The changes in estimated average shortest path length are so large between simulations where alpha is changed that the confidence intervals are not overlapping. With that in mind, we can at a significance level of 95 % say that these estimates are in fact different.

We also see signs indicating that changes in β_1 are more substantial when α_1 is high, rather than when it is zero. When α_1 is fixed at zero and the β_1 is varied, the average shortest path length shows very small variations, and based on the point-estimate, does not showcase the same trend as we see in the rest of the parameter-space with decreased β_1 leading to increased average shortest path length. However, if we look at the interval estimates (case 7-10 in the Appendix), we see that these confidence intervals are overlapping, and we can not conclude that there is not a similar, but weaker relationship between these statistics as well. At the same time, we do not have a basis to say that these network statistics are significantly different from each other.

The same thing as we saw for average shortest path length is seen in mean degree, only with the development being opposite. Where average shortest path length returns small estimates, mean degree are high, and the other way around. This makes us believe that there in the case of GARCH-simulated time-series are strong dependences between average shortest path length and mean degree, a relationship we also suggested in Section 4.3.4.

In the cases of assortativity and global transitivity, we see something similar to what we saw in average shortest path length and mean degree above. Changes in β_1 create less significant changes in the network statistics than what changes in α_1 does. However, the changes we see in these two cases are not as clear as the ones we see in the two graphs above. With that in mind, it is possible that average shortest path length and mean degree are better statistics to use when the goal is to separate between different parameter-combinations, and following, when separating between different types of financial time-series.

In Figure 12 we have done the same drill as in Figure 11, but in this case for the horizontal visibility graph. Since mean degree and the two transitivity-statistics, as we showed earlier, produces very similar

estimates between the different simulations, we only show graphs for average shortest path length and assortativity here.

Compared to what we saw from the natural visibility graph, the horizontal visibility graph has some interesting results. As in the case for natural visibility graph, we see tendencies of a dependency between the parameter combination and the average shortest path length, but there seems to be a negative correlation between the estimation from the natural and horizontal visibility graph. The horizontal visibility graph has its lowest average shortest path length around the origin and along the x-axis. For the natural visibility graph it is the opposite way around, where these simulation cases have high average shortest path length around origin and along the x-axis.

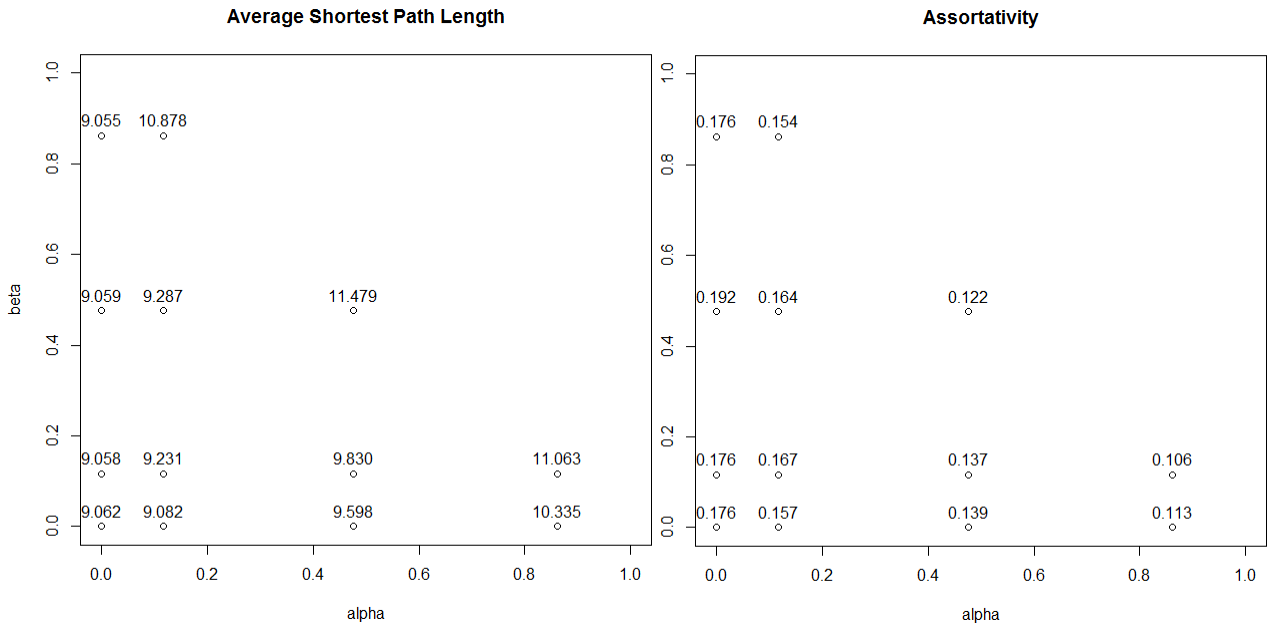

Figure 12 Plots showing the estimated value of the average shortest path length and assortativity for all thirteen simulations with the horizontal visibility graph.

For the assortativity, we see similar patterns between the natural and horizontal visibility graph. It is low when α_1 is set high, and high when it is set at a lower value. Based on our simulations we also see that the range is larger for the assortativity in the case of natural visibility graphs. Based on these findings, and how much more clear cut the findings are when natural visibility graph is used, it seems to us like the natural visibility graph has more power than the horizontal visibility graph does, when the goal is to differentiate between financial time-series.

5.2 S&P 500

As discussed in Section 4, we have applied the natural visibility graph and horizontal visibility graph algorithm on ten subsets of the S&P 500 portfolio, all consisting of 2500 observations. This is done for both the return-series and the raw price-series, and the results obtained from both will be presented in this section. We will also in this section show some examples of what the degree distribution of a time-series like this will typically look like. We will use the most present of the ten sub-sets as an example. Although we will do no further testing of the degree distribution, and will stick to mean degree as our test statistic, we believe it can be helpful to show some examples of what it looks like, in the process of understanding how the edges are distributed between nodes in a network like this. Finally, we have in this section estimated GARCH-parameters for all ten of the subsets from the S&P 500 return-series. We believe it can be an interesting study to see whether the relationships we saw between the parameters and the network statistics in the simulation case will carry over to real-life data. The estimated parameters are presented in table 11, and we will see the network statistics for the return-series in table 12.

year	α_0	α_1	β_1
1962-1971	$1.124 * 10^{-6}$	0.1671	0.8137
1967-1976	$6.991 * 10^{-7}$	0.0776	0.9121
1972-1981	$6.339 * 10^{-7}$	0.0501	0.9423
1977-1986	$8.664 * 10^{-7}$	0.0352	0.9535
1982-1991	$5.446 * 10^{-6}$	0.0926	0.8548
1987-1996	$1.745 * 10^{-6}$	0.0905	0.8915
1992-2001	$5.759 * 10^{-7}$	0.0698	0.9271
1997-2006	$7.417 * 10^{-7}$	0.0726	0.9241
2002-2011	$1.346 * 10^{-6}$	0.0836	0.9075
2007-2016	$2.938 * 10^{-6}$	0.1169	0.8612

Table 11: Estimated parameters for each sub-set of the S&P 500 return-series. All values are significant, and the rest of the test statistics can be seen in the appendix.

As we can see from the GARCH-parameters, the variations between the different subsets are far smaller here than what we were using in our simulation-study. The highest estimated α_1 is 0.1671 and the lowest estimated β_1 is 0.8137, both representing the first of the ten subsets. Consequently, this leads to less variation in the obtained network statistics than what we saw in the simulation-study. In addition, we have a third variable which is changing between the ten subsets, that we did not have in the simulation-study, namely the α_0-parameter. The effect of this variable is to us unknown, but the variations in the parameter seems to be relatively small, and as a consequence we do not think its variations are very important for the obtained results.

As we remember from the simulation-study, we saw the clearest sign of a relationship between the network statistics and parameters in the case of natural visibility graph for the average shortest path length and mean degree, and there were especially two things that had an effect on the obtained results. First, we saw that the higher the added parameter values were, the higher the network statistics became for mean degree and the lower it became for average shortest path length. We also saw that the same changes were obtained when we increased α_1 while decreasing the β_1 proportionally.

When we look at the results form S&P 500 in Table 10, the mentioned relationship for mean degree is not showing up in the same way as in the simulation-study. However, for average shortest path length, we see some very similar results. In general, we see that time-series with a high estimated α_1 often returns network with a short average shortest path length, and the other way around for low estimated α_1. The only subset that stands out as an outlier in terms of this relationship is the one stretching from 2002 to 2011, where the average shortest path length was low despite of a low estimated α_1. Exactly why this is so, is something we do not know with certainty, but our main theory is that the financial crisis in the middle of this subset is contributing to shorten the average shortest path length. The highest returns during this period are higher than what we see in most of the other subsets, and it gives these observations great visibility over the time-series.

Although we see a similar relationship in this real-life case as we did in the simulation-study, these results are obviously not as solid. In the simulations, we were comparing the mean of 100 replications for each simulation, while we here compare the network statistics from a single replication. Consequently, these results are more prone to outliers, which as we will show from the simulation-study in the appendix, can be significant. With that in mind, these relationships that we have seen in this case might just as well be spurious as something real. An alternative argument can just as well be made saying that the constant being high leads to a short average path length, and it will make sense based on the obtained results.

	S&P 500: returns							
	Horizontal visibility graph				Natural Visibility Graph			
	ASPL	Assorta-tivity	Transi-tivity	Mean degree	ASPL	Assorta-tivity	Transi-tivity	Mean degree
1962-1971	9.94	0.16	0.35	3.99	4.90	0.10	0.32	6.63
1967-1976	11.53	0.17	0.35	3.98	5.20	0.18	0.36	6.50
1972-1981	19.25	0.18	0.35	3.98	5.53	0.19	0.37	6.28
1977-1986	10.44	0.17	0.34	3.98	5.52	0.18	0.37	6.17
1982-1991	10.46	0.17	0.34	3.99	4.83	0.18	0.34	6.20
1987-1996	10.05	0.18	0.34	3.98	4.95	0.19	0.35	6.11
1992-2001	10.01	0.17	0.34	3.98	5.67	0.18	0.36	6.07
1997-2006	11.34	0.17	0.34	3.98	5.35	0.22	0.36	6.00
2002-2011	11.23	0.17	0.33	3.99	4.80	0.22	0.32	6.09
2007-2016	11.47	0.16	0.34	3.97	5.00	0.22	0.33	6.02

Table 12: Network statistics for networks created with the hoirzontal and natural visibility graph algorithm on the return-series of S&P 500. The statistics used are the same as in the simulation-study, except for mean local transitivity that were excluded due to its lack of variation between the different simulation sets. Each set is of 2500 observations

As we previously mentioned, the obtained network statistics in this study are more clustered than what we saw in the simulation case. This is likely because the fractal structure of the ten subsets are self-similar to the fractal structure of return-series as a whole, according to the theories of Mandelbrot (1997).

Further, what we see here is very similar to what we saw in the simulation-study. The average shortest path length for the horizontal visibility graph is again significantly higher than what we see for the natural visibility graph. Again, this is clearly due to the natural visibility graph having more edges in it than the horizontal visibility graph does. An interesting thing to note on the path length, is that the five portfolios with the shortest average path length also are the five portfolios with the highest kurtosis. This is a relationship we also saw in the simulation-case, and there might be a dependence there, but based on such a small sample size it might also be a spurious relationship.

The rest of the statistics for the horizontal visibility graph does not seem to show much difference between the different subsets. For mean degree and global transitivity, this is similar to what we saw in the simulation case. For assortativity we saw some larger variations, but we are not able to see similar results here. This is likely due to the closeness of the estimated parameters, indicating that the time-series are more similar here than in the simulations.

In the case of the natural visibility graph, we see signs of a growing assortativity coefficient. We mentioned that we expected a positive coefficient in Section 3.4.2, but why it is differing so much between some of the subsets, and if an increased assortativity coefficient has a financial meaning, is something we are not fully able to explain. As in the case for the horizontal visibility graph, the global transitivity seems to be

fairly constant across subsets, but here as well, a case can be made for a possible relationship with the kurtosis. The arithmetic mean of the node degree, seems to be following an opposite trend of the assortativity, where it is lower in recent years than it is for the subsets further back in time. Unlike the average shortest path length, we are not able to see relationships for the other test statistics similar to those we saw in the simulation-study.

In table 13, we can see the results obtained when both visibility graph-algorithms were applied to the raw S&P 500 price-series. We believe it is interesting to see some examples of this as well, since it obviously is a totally different process than what the return-series is. That fact can clearly be seen by comparing the test statistics between networks from the return-series and price-series. As we saw in Figure 8 in the data-section, the trend for S&P 500 has been that it is steadily increasing with time, except for some periods of decline due to different financial crises. This clearly has an effect on the average shortest path length for the horizontal visibility graph, due to a structure like this creating far less edges than what we saw for the return-series. We can also see that the mean degree goes down, a consequence of the time-series structure, where a high degree of the observations are larger than every single node preceding it. We can also see that the graphs created from this time-series are close to non-assortative and that the global transitivity actually goes up in comparison to the return-series. As we mentioned in Section 3.4.2, a random network should have an assortativity of zero, and we are laying close to this in all the subsets. However, we do not believe the graph actually is random, due to its very high average shortest path length.

	S&P 500: Prices							
	Horizontal visibility graph				Natural Visibility Graph			
	ASPL	Assorta-tivity	Transi-tivity	Mean degree	ASPL	Assorta-tivity	Transi-tivity	Mean degree
1962-1971	97,98	0.05	0.40	3.78	5.62	0.40	0.29	17.69
1967-1976	64.40	0.04	0.40	3.87	4.70	0.41	0.27	18.29
1972-1981	38.72	0.03	0.40	3.91	4.77	0.39	0.26	16.90
1977-1986	68.24	0.04	0.39	3.82	4.17	0.24	0.21	19.23
1982-1991	84.37	0.05	0.38	3.81	4.67	0.28	0.29	15.02
1987-1996	74.05	0.06	0.39	3.81	4.06	0.21	0.21	18.19
1992-2001	134.19	0.06	0.40	3.72	4.80	0.22	0.18	21.89
1997-2006	59.65	0.05	0.39	3.89	5.99	0.32	0.23	16.63
2002-2011	71.23	0.02	0.38	3.90	5.38	0.17	0.25	15.37
2007-2016	66.36	0.02	0.38	3.87	4.57	0.05	0.18	16.62

Table 13: Table 14: Network statistics for networks created with the hoirzontal and natural visibility graph algorithm on the raw price-series of S&P 500. The statistics used are the same as in the simulation-study, except for mean local transitivity that were excluded due to its lack of variation between the different simulation sets. Each set is of 2500 observations.

For the natural visibility graph, we can also see from the statistics that the network this time-series generates are very different in comparison to what we saw from the return-series. While we see that the

mean degree goes up significantly, the changes in average shortest path length are relatively small, and does not follow a specific pattern in comparison to the return-series. The fact that the mean degree goes up substantially, without much change in path length indicates that this graph has some structural holes which the flow of information is forced through. We also see significant variations between the sub-sets in both transitivity and especially in assortativity. While these results obviously showcase the different network statistics for two very different financial processes, it is still necessary to investigate the effectiveness of the visibility graph algorithm when applied to time-series without constant mean, as this has not been done in this thesis.

Finally, in the last part of this result section will show some examples of the cumulative frequency and relative frequency of the degree distribution of a network. We have used the one stretching from 2007 to 2016 as our example. We will show it for both the natural and horizontal visibility graph of the return-series, but only for the natural visibility graph of the price series, as we found these results to be more interesting than those of the horizontal visibility graph.

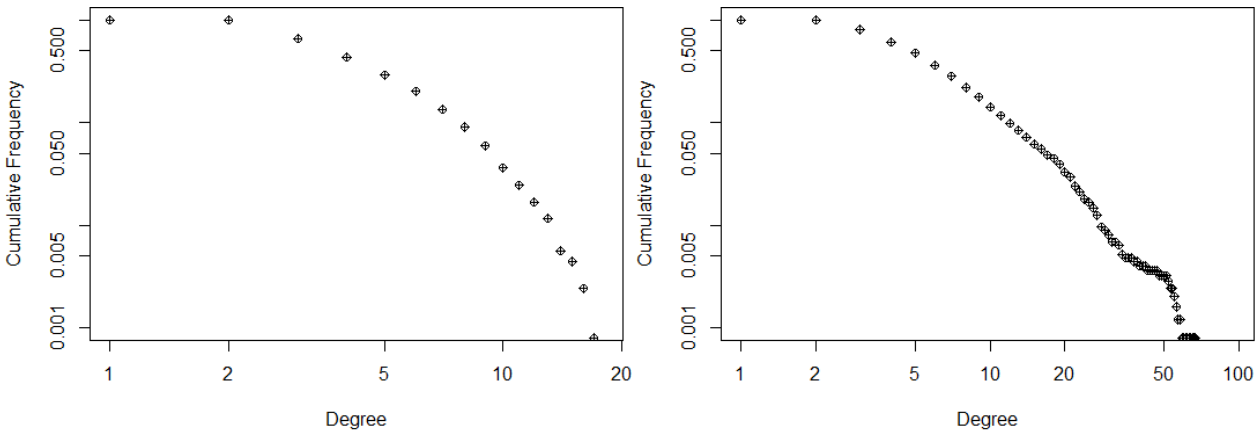

Figure 13: Cumulative frequency of the degree distribution. The data set is the return series of the S&P 500 stretching form 2007-2016. Horizontal visibility graph (left) and natural visibility graph (right).

Figure 13 shows the cumulative frequency of nodes with a certain degree from the horizontal visibility graph and the natural visibility graph. This is undoubtedly an uncommon way of presenting the probability density function in terms of financial data, as we normally would see a graph going from the bottom left corner to the upper right corner, meaning that we here look at $P(X \geq x)$ rather than $P(X \leq x)$ which is the more common way. However, this seems to be the consensus way of doing it within graph-theory, and we are hence choosing to stay consistent with that. As we can see in Figure 13, the cumulative frequency of nodes with a degree of one or two is laying very close to 100 % (in fact, all but one node has at least two degrees in this graph). This is obviously due to the previously discussed nature of the visibility graphs

applied in this paper, where the graph is connected, making it so that all nodes are connected to at least its nearest neighbors.

After the mandatory connections, we see a rather quick decline in the amount of nodes with a given degree. In fact, no more than approximately 50 % of the nodes in the horizontal visibility graph have a degree of three or higher, while the same number is four in the natural visibility graph. Clearly there is a large amount of nodes within the network with a relatively small degree, while only a few nodes, the hubs, showcase a high number of connected nodes. This is common to see in networks, and can be shown even more clearly by looking at the relative frequency of degrees. Figure 14 shows that in the horizontal visibility graph, nearly 35 % of nodes have a degree of two, and that the amount of nodes with a given degree declines quickly, following something that looks like an exponential distribution. Something similar can be seen in the case of the natural visibility graph. Due to the horizontal visibility graph being a subgraph of the natural visibility graph, and the natural visibility graph consequentially having more edges, we observe a lower amount of nodes with degree two. We also see that there are in fact more nodes with degree three than two, but as in the case for the horizontal visibility graph, there is a steep decline from there. Another thing to observe is that the degrees of the hubs in the natural visibility graph is way higher than in the horizontal visibility graph.

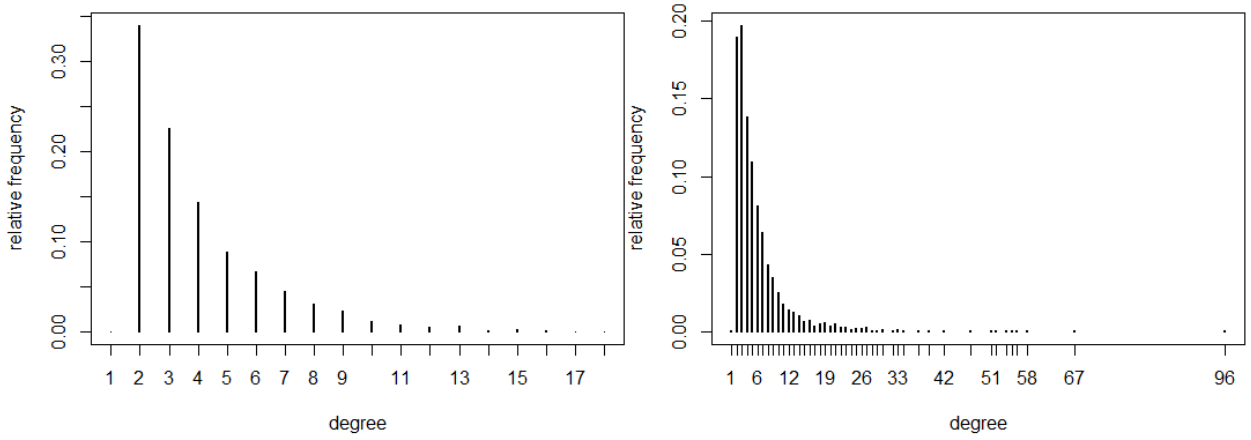

Figure 14: Relative frequency of the degree distribution. The data set is the return series of the S&P 500 stretching form 2007-2016. Horizontal visibility graph (left) and natural visibility graph (right)

Finally, we have in Figure 15 plotted cumulative frequency and relative frequency of the node degree for the subset consisting of data from 2006 to 2017 form the S&P 500 price-series. This is from the natural visibility graph. Visually the distributions themselves look very similar to what we saw from the return-series, but by looking at the axes, we can see the results of a higher mean degree. The price-series has fewer nodes with a low degree, and the hubs are significantly bigger than in the return-series.

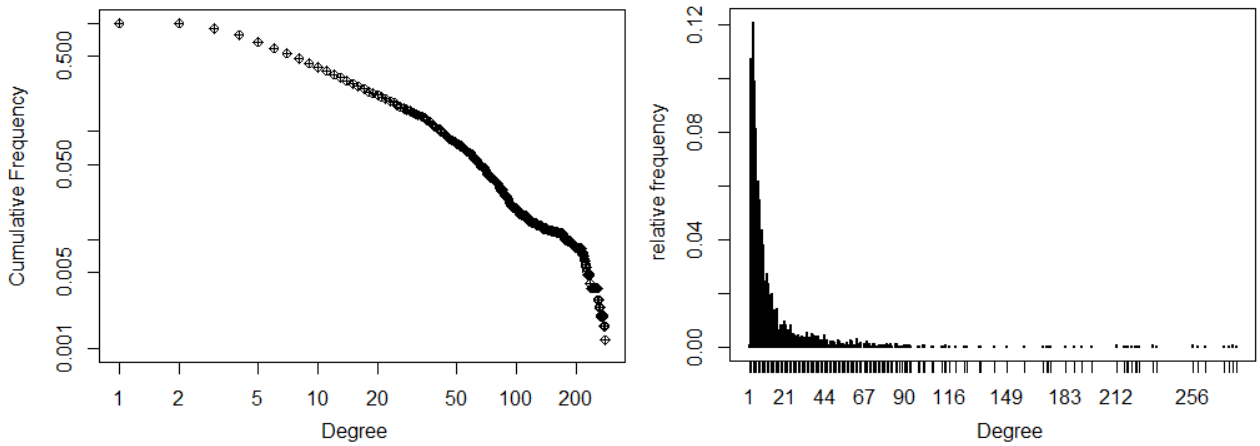

Figure 15: Cumulative frequency and relative frequency from a network generated with the natural visibility graph algorithm. The data set is S&P 500 price-series stretching from 2006 to 2017.

5 Discussion

As we discussed in the literature review, there has not been a lot of studies covering this specific topic, especially in regards to financial time-series. As far as we are concerned, the only comparable study that has looked thoroughly into network statistics of graphs generated from natural and horizontal visibility graph algorithms on simulated time-series are the one of Zhang et al. (2017). Similar to what we have done, they applied the five network-statistics mean degree, average shortest path length, assortativity, local transitivity and global transitivity to their networks, however, with a slightly different approach than what we have done. Consequently, the results obtained in this thesis and that paper are not directly comparable. Zhang et al. (2017) tested whether their network measures showed similar variation patterns as their auto correlation function itself did, and found evidence supporting that for all five of the network statistics. In our thesis we have tested how the network statistics respond to changes in the parameters for a GARCH-process, and unlike Zhang et al. (2017) have found conflicting results between the different network statistics. With that in mind, we do not share the belief in the effectiveness of all the network statistics, especially in the case of horizontal visibility graph, where most of them showed no, or very small changes with variation in the GARCH-parameters, and thus a limited ability to detect differences between time-series.

Another important aspect of this thesis is to investigate how real-life examples compare to those of the simulated time-series. Although the results obtained in the simulations are intriguing and interesting, they are eventually not worth much unless the results show similar patterns as we see in real-life time-series. To account for this aspect, as shown in the results section, we have applied the natural and horizontal visibility graph to ten subsets of the S&P 500 index. The subset with data from 2007-2016 and Case 1 of the GARCH-simulation should be the most comparable cases, since they both share the same GARCH-parameters. Between them, we see that we get similar results in regards to the network statistics, especially in the case of natural visibility graph. The assortativity, global transitivity and the mean degree are almost identical between the return-series and the simulation. We find a bit of discrepancy when it comes to the average shortest path length, but the difference is not huge, and although the result from the S&P 500 are outside of our estimated confidence interval for the mean, it is still within the range of estimated statistics from the 100 simulations (as can be seen in the appendix).

We have seen in the simulation-cases that especially the estimated mean degree and average shortest path lengths seems to be very good at indicating differences between GARCH-processes with different

parameter-combinations when applied to networks generated with the natural visibility graph algorithm. With that in mind, we believe we have a basis to say that in a situation like the simulations, where we can use maximum likelihood estimation to find an estimate of the network statistic from multiple replications, the natural visibility algorithm has ability to differentiate between different processes. However, in real-life situations, like in our study of the S&P 500, we do not have multiple samples with equal GARCH-parameters to find an estimate from, and our estimate will consequently be more prone to outliers. As we saw in the study of S&P 500, it is clearly possible to find relationships similar to what we saw in the simulations, but the close similarity of the ten subsets and the relatively small sample size. This makes it hard to say whether these relationships are in fact real or just a spurious relationship.

Considering this, a method that can better estimate the network statistics might possibly be needed for this algorithm to ever be reliable when we are looking at a single time-series, rather than a large group of replications. Exactly how this should be done is obviously beyond the scope of this thesis, but a possible suggestion is to divide the time-series into subgroups at a variety of different lengths. We can then find network statistics for all the different sub-groups, and then use a suited estimator to find an estimate of the network statistic of for the network as a whole. However, there is obviously a lot to consider in such a process, including how to deal with the fact that the networks statistics will vary between networks of different size.

6 Conclusion

Throughout this thesis, we have tested both the natural and horizontal visibility graph on ten subsets of the S&P 500 index and on thirteen sets of simulations from a GARCH(1,1)-process. The main focus has been the five network statistics mean degree, average shortest path length, assortativity, local transitivity and global transitivity, and how they behave in networks from different time-series with varying GARCH-parameters.

The most intriguing findings in this thesis are those found when using the natural visibility graph algorithm on the GARCH-simulated time-series. We showed that especially mean degree and average shortest path length were able to separate between time-series of different GARCH-parameters in a nice way. Further it was shown that changes in the parameter α_1 lead to larger changes in the estimated network statistics than changes in the β_1-parameter did. Changes in β_1 also seemed to be more substantial the higher the α_1-parameter was set. Even in the case of S&P 500, we saw indications of a relationship between the estimated GARCH-parameters and average shortest path length, but as we previously stated, we do not feel like we have enough information to conclude whether this relationship is real or just spurious.

Our analysis with the horizontal visibility graph algorithm did in general not give as interesting and intriguing results as the natural visibility graph algorithm did. Although far fewer than with the natural visibility graph, we also saw some signs of the horizontal visibility graph being able to separate between different time-series. Consequently, we are not going to dismiss future utilization of this algorithm, but will based on the findings in this thesis rather advocate for the usage of the natural visibility graph in regards to future studies of financial time-series.

Although the sample size of real-life time-series in this thesis is rather small, we believe that our findings from the simulation study, and from the study of S&P 500 suggests that the natural visibility graph can be a useful tool in analyzing financial time-series. However, we also believe that this is a topic that needs further testing before any conclusions of the tools effectiveness can be drawn. In future research we suggest to perform similar simulation studies as in this paper, but with a varying constant, and more importantly, under variations of the length of the time-series. We would also suggest that the algorithm should be tested on a larger portfolio of real-life time series, and compared to the findings in this study. In that way, we would get a better indication of whether the possible relationships we observed in the case of S&P 500 is something that can repeatedly be seen in financial time-series.

7 References

Barabási, A.-L., & Albert, R. (1999). "Emergence of Scaling in Random Networks". *Science, 286*(5439), 509-512.

Bollerslev, T. (1986). "Generalized autoregressive conditional heteroskedasticity". *Journal of Econometrics, 31*(3), 307-327.

Brooks, C. (2008). *Introductory Econometrics for Finance* (2nd ed.). New York: Cambridge University Press.

Cohen, R., & Havlin, S. (2003). "Scale-Free Networks Are Ultrasmall". *Physical Review Letters, 90*(5), 058701.

Elsner, J. B., Jagger, T. H., & Fogarty, E. A. (2009). "Visibility network of United States hurricanes". *Geophysical Research Letters, 36*(16).

Engle, R. F. (1982). "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation". *Econometrica, 50*(4), 987-1007.

Engle, R. F. (2004). "Risk and Volatility: Econometric Models and Financial Practice". *The American Economic Review, 94*(3), 405-420.

Erdos, P., & Rényi, A. (1960). "On the evolution of random graphs". *Publ. Math. Inst. Hung. Acad. Sci, 5*(1), 17-60.

Erdös, P., & Rényi, A. (1959). "On random graphs, I". *Publicationes Mathematicae (Debrecen), 6*, 290-297.

Lacasa, L., Luque, B., Ballesteros, F., Luque, J., & Nuño, J. C. (2008). "From time series to complex networks: The visibility graph". *Proceedings of the National Academy of Sciences, 105*(13), 4972-4975.

Lacasa, L., Luque, B., Luque, J., & Nuño, J. C. (2009). "The visibility graph: A new method for estimating the Hurst exponent of fractional Brownian motion". *EPL (Europhysics Letters), 86*(3), 30001.

Lacasa, L., & Toral, R. (2010). "Description of stochastic and chaotic series using visibility graphs". *Physical Review E, 82*(3), 036120.

Liu, C., Zhou, W.-X., & Yuan, W.-K. (2010). "Statistical properties of visibility graph of energy dissipation rates in three-dimensional fully developed turbulence". *Physica A: Statistical Mechanics and its Applications, 389*(13), 2675-2681.

Long, Y. (2013). "Visibility graph network analysis of gold price time series". *Physica A: Statistical Mechanics and its Applications, 392*(16), 3374-3384.

Luque, B., Lacasa, L., Ballesteros, F., & Luque, J. (2009). "Horizontal visibility graphs: Exact results for random time series". *Physical Review E, 80*(4), 046103.

Mandelbrot, B. B. (1963). "The Variation of Certain Speculative Prices". *The Journal of Business, 36*(4), 394-419.

Mandelbrot, B. B. (1997). *Fractals and scaling in finance : discontinuity, concentration, risk : selecta volume E*. New York: Springer Science+Business Media.

Mao, G., & Zhang, N. (2013). "Analysis of Average Shortest-Path Length of Scale-Free Network". *Journal of Applied Mathematics, 2013*, 5.

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

Newman, M. E. J. (2003). "The Structure and Function of Complex Networks". *SIAM Review, 45*(2), 167-256.

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

Ni, X.-H., Jiang, Z.-Q., & Zhou, W.-X. (2009). "Degree distributions of the visibility graphs mapped from fractional Brownian motions and multifractal random walks". *Physics Letters A, 373*(42), 3822-3826.

Packard, N. H., Crutchfield, J. P., Farmer, J. D., & Shaw, R. S. (1980). "Geometry from a Time Series". *Physical Review Letters, 45*(9), 712-716.

Pierini, J. O., Lovallo, M., & Telesca, L. (2012). "Visibility graph analysis of wind speed records measured in central Argentina". *Physica A: Statistical Mechanics and its Applications, 391*(20), 5041-5048.

Shao, Z.-G. (2010). "Network analysis of human heartbeat dynamics". *Applied Physics Letters, 96*(7), 073703.

Sun, M., Wang, Y., & Gao, C. (2016). "Visibility graph network analysis of natural gas price: The case of North American market". *Physica A: Statistical Mechanics and its Applications, 462*, 1-11.

Telesca, L., & Lovallo, M. (2012). "Analysis of seismic sequences by using the method of visibility graph". *EPL (Europhysics Letters), 97*(5), 50002.

Wang, N., Li, D., & Wang, Q. (2012). "Visibility graph analysis on quarterly macroeconomic series of China based on complex network theory". *Physica A: Statistical Mechanics and its Applications, 391*(24), 6543-6555.

Watts, D. J., & Strogatz, S. H. (1998). "Collective dynamics of `small-world' networks". *Nature, 393*(6684), 440-442.

Xu, X., Zhang, J., & Small, M. (2008). "Superfamily phenomena and motifs of networks induced from time series". *Proceedings of the National Academy of Sciences, 105*(50), 19601-19605.

Yang, Y., Wang, J., Yang, H., & Mang, J. (2009). "Visibility graph approach to exchange rate series". *Physica A: Statistical Mechanics and its Applications, 388*(20), 4431-4437.

Yang, Y., & Yang, H. (2008). "Complex network-based time series analysis". *Physica A: Statistical Mechanics and its Applications, 387*(5–6), 1381-1386.

Yu, Z. G., Anh, V., & Eastes, R. (2012). "Multifractal analysis of solar flare indices and their horizontal visibility graphs". *Nonlinear Processes in Geophysics, 19*(6), 657-665.

Zhang, J., & Small, M. (2006). "Complex Network from Pseudoperiodic Time Series: Topology versus Dynamics". *Physical Review Letters, 96*(23), 238701.

Zhang, R., Zou, Y., Zhou, J., Gao, Z.-K., & Guan, S. (2017). "Visibility graph analysis for re-sampled time series from auto-regressive stochastic processes". *Communications in Nonlinear Science and Numerical Simulation, 42*, 396-403.

Zhuang, E., Small, M., & Feng, G. (2014). "Time series analysis of the developed financial markets' integration using visibility graphs". *Physica A: Statistical Mechanics and its Applications, 410*, 483-495.

8 Appendix

8.1 Acknowledgements

We would like to thank our supervisor Jochen Jungeilges for guidance throughout the whole process of writing this thesis. The input and feedback we have received has been very helpful for us to be able to write a thesis within this field. A special emphasize should be given to the help we received writing the scripts in R that transformed the time-series into adjacency matrices for both the natural and horizontal visibility graph algorithms. This was something way beyond our programming skills in R, and the help we received has been critical for the possibility to write this thesis.

9.2 Tables

Case 7	α_1 = 0, β_1 = 0.8612					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.006	3.013	$-3.79*10^{-6}$	0.005	0.016	-0.016
Standard deviation	0.054	0.099	$7.52*10^{-5}$	$5.92*10^{-5}$	$1.6*10^{-3}$	$1.6*10^{-3}$
Confidence interval	±0.011	±0.019	$\pm1.47*10^{-5}$	$\pm1.16*10^{-5}$	$\pm3.2*10^{-4}$	$\pm3.1*10^{-4}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.055	0.176	0.337	0.122	3.987	
Standard deviation	0.465	0.011	0.002	$1.52*10^{-4}$	0.003	
Confidence interval	±0.091	±0.002	$\pm4.05*10^{-4}$	$\pm2.98*10^{-5}$	$\pm5.4*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.711	0.191	0.376	0.154	5.877	
Standard deviation	0.189	0.006	0.006	$6.2*10^{-4}$	0.052	
Confidence interval	±0.037	±0.003	±0.001	$\pm1.2*10^{-4}$	±0.010	

Table 15: Simulation case 7: α_1 = 0 and β_1 = 0.8612

Case 8	α_1 = 0, β_1 = 0.4755					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.007	3.001	$-2.7*10^{-6}$	$2.4*10^{-3}$	0.008	-0.008
Standard deviation	0.050	0.108	$5.4*10^{-5}$	$3.3*10^{-5}$	$7.4*10^{-4}$	$7.9*10^{-4}$
Confidence interval	$\pm9.7*10^{-3}$	±0.021	$\pm1.1*10^{-5}$	$\pm6.5*10^{-6}$	$\pm1.5*10^{-4}$	$\pm1.5*10^{-4}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.059	0.192	0.336	0.122	3.988	
Standard deviation	0.425	0.014	0.002	$1.6*10^{-4}$	$2.6*10^{-3}$	
Confidence interval	±0.083	±0.003	$\pm4.1*10^{-4}$	$\pm3.2*10^{-5}$	$\pm5*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.739	0.192	0.376	0.154	5.863	
Standard deviation	0.216	0.014	0.005	$6.4*10^{-4}$	0.058	
Confidence interval	±0.039	±0.003	$\pm2.7*10^{-3}$	$\pm1.3*10^{-4}$	±0.011	

Table 16: Simulation case 8: α_1 = 0 and β_1 = 0.4755

Case 9	α_1 = 0, β_1 = 0.1169					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.003	2.987	-2.901	$1.8*10^{-3}$	$6.3*10^{-3}$	-0.006
Standard deviation	0.050	0.096	$3.9*10^{-5}$	$2.6*10^{-5}$	$5.3*10^{-4}$	$6*10^{-4}$
Confidence interval	$\pm9.8*10^{-3}$	±0.019	$\pm7.7*10^{-6}$	$\pm5.1*10^{-6}$	$\pm1*10^{-4}$	$\pm1.2*10^{-4}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.058	0.176	0.336	0.122	3.987	
Standard deviation	0.0445	0.010	0.002	$1.8*10^{-4}$	0.003	
Confidence interval	±0.087	±0.002	$\pm4.5*10^{-4}$	$\pm3.5*10^{-5}$	$\pm5.5*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.750	0.190	0.377	0.154	5.870	
Standard deviation	0.170	0.013	0.005	$6.5*10^{-4}$	0.047	
Confidence interval	±0.033	$\pm2.5*10^{-3}$	±0.001	$\pm1.3*10^{-4}$	±0.009	

Table 17: Simulation case 7: α_1 = 0 and β_1 = 0.1169

Case 10	α_1 = 0, β_1 = 0					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	0.002	2.986	$-3.7*10^{-6}$	0.002	0.006	-0.006
Standard deviation	0.050	0.088	$3.4*10^{-5}$	$2.3*10^{-5}$	$5.3*10^{-4}$	$4.9*10^{-4}$
Confidence interval	±0.010	±0.017	$\pm6.6*10^{-6}$	$\pm4.6*10^{-6}$	$\pm1.0*10^{-4}$	$\pm9.7*10^{-5}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.062	0.176	0.336	0.122	3.986	
Standard deviation	0.421	0.009	0.002	$1.6*10^{-4}$	0.003	
Confidence interval	±0.082	±0.002	$\pm4.3*10^{-4}$	$\pm3.2*10^{-5}$	$\pm5.6*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.742	0.190	0.376	0.154	5.865	
Standard deviation	0.190	0.012	0.005	$7.1*10^{-4}$	0.054	
Confidence interval	±0.037	±0.002	±0.001	$\pm1.4*10^{-4}$	±0.011	

Table 18: Simulation case 10: α_1 = 0 and β_1 = 0

Case 11	α_1 = 0.1169, β_1 = 0					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	$-1.7*10^{-3}$	2.981	$-9.5*10^{-7}$	$1.7*10^{-3}$	0.006	-0.006
Standard deviation	0.043	0.085	$3.4*10^{-5}$	$2.4*10^{-5}$	$5.4*10^{-4}$	$4.9*10^{-4}$
Confidence interval	±0.008	±0.017	$\pm6.7*10^{-6}$	$\pm4.7*10^{-6}$	$\pm1*10^{-4}$	$\pm9.7*10^{-5}$
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.082	0.175	0.337	0.122	3.987	
Standard deviation	0.425	0.011	0.002	$1.4*10^{-4}$	$2.6*10^{-3}$	
Confidence interval	±0.083	±0.002	$\pm3.6*10^{-4}$	$\pm2.8*10^{-5}$	$\pm5*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.723	0.194	0.378	0.154	5.882	
Standard deviation	0.190	0.015	0.006	$7*10^{-4}$	0.045	
Confidence interval	±0.037	±0.003	±0.001	$\pm1.4*10^{-4}$	±0.010	

Table 19: Simulation case 11: α_1 = 0.1169 and β_1 = 0

Case 12	α_1 = 0.4755, β_1 = 0					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	-0.021	6.334	$-1.9*10^{-6}$	$2.4*10^{-3}$	0.014	-0.014
Standard deviation	0.312	4.591	$4.9*10^{-5}$	$8.9*10^{-5}$	$4.5*10^{-3}$	0.005
Confidence interval	±0.061	±0.899	$\pm9.6*10^{-6}$	$\pm1.7*10^{-5}$	$\pm8.9*10^{-4}$	±0.001
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	9.598	0.139	0.338	0.122	3.986	
Standard deviation	0.598	0.013	0.003	$1.7*10^{-4}$	0.003	
Confidence interval	±0.117	±0.002	$\pm5*10^{-4}$	$\pm3.4*10^{-5}$	$\pm6.3*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	5.320	0.157	0.356	0.156	6.237	
Standard deviation	0.258	0.027	0.015	$7.7*10^{-4}$	0.088	
Confidence interval	±0.051	±0.005	±0.003	$\pm1.5*10^{-4}$	±0.017	

Table 20:: Simulation case 7: α_1 = 0.4755 and β_1 = 0

Case 13	α_1 = 0.8612, β_1= 0					
	Descriptive Statistics					
	Skewness	Kurtosis	Mean	SD	Max	Min
Mean	0.815	57.870	$1.4*10^{-5}$	0.004	0.054	-0.052
Standard deviation	3.702	115.983	$8.4*10^{-5}$	0.002	0.054	0.069
Confidence interval	±0.726	±22.733	$\pm1.6*10^{-4}$	$\pm4.4*10^{-4}$	±0.011	±0.014
	Horizontal Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	10.335	0.113	0.339	0.122	3.985	
Standard deviation	0.775	0.012	0.003	$1.7*10^{-4}$	0.003	
Confidence interval	±0.152	±0.002	$\pm5.0*10^{-4}$	$\pm3.3*10^{-5}$	$\pm6.3*10^{-4}$	
	Natural Visibility Graph					
	ASPL	Assortativity	Global Transivitivty	Mean Local Transitivity	Mean Degree	
Mean	4.780	0.066	0.302	0.158	6.920	
Standard deviation	0.329	0.041	0.044	$9.3*10^{-4}$	0.266	
Confidence interval	±0.065	±0.008	±0.009	$\pm1.8*10^{-4}$	±0.052	

Table 21: Simulation case 13: α_1 = 0.8612 and β_1 = 0

Range					
Case 1	α_1 = 0.1169, β_1 = 0.8612				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	10.878	0.154	0.339	0.122	3.984
Max	14.917	0.181	0.345	0.122	3.991
Min	9.150	0.126	0.333	0.122	3.974
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.452	0.229	0.357	0.154	6.107
Max	6.341	0.281	0.376	0.156	6.323
Min	4.971	0.152	0.305	0.152	5.906
Case 2	α_1 = 0.1169, β_1 = 0.4755				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.287	0.164	0.337	0.122	3.986
Max	11.218	0.189	0.343	0.122	3.992
Min	8.016	0.141	0.330	0.121	3.979
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.688	0.164	0.376	0.154	5.974
Max	6.250	0.228	0.390	0.156	6.105
Min	5.281	0.161	0.355	0.153	5.778
Case 3	α_1 = 0.1169, β_1 = 0.1169				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.231	0.167	0.337	0.122	3.986
Max	10.623	0.190	0.343	0.122	3.992
Min	8.313	0.143	0.332	0.121	3.976
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.709	0.192	0.376	0.154	5.925
Max	6.639	0.229	0.396	0.156	6.070
Min	5.221	0.160	0.360	0.153	5.747
Case 4	α_1= 0.4755, β_1= 0. 1169				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.830	0.137	0.338	0.122	3.985
Max	11.298	0.164	0.344	0.122	3.993
Min	8.582	0.115	0.332	0.121	3.978
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.290	0.157	0.353	0.156	6.283
Max	6.125	0.209	0.379	0.157	6.624
Min	4.765	0.064	0.305	0.154	6.026
Case 5	α_1= 0.8612, β_1= 0. 1169				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	11.063	0.106	0.340	0.122	3.984
Max	14.751	0.141	0.350	0.122	3.992
Min	9.416	0.078	0.334	0.121	3.974
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	4.635	0.041	0.283	0.158	7.195
Max	5.607	0.136	0.353	0.160	8.119
Min	3.835	-0.045	0.148	0.157	6.613

Table 22: Range for simulation case 1-5

Range					
Case 6	α_1= 0.4755, β_1= 0.4755				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	11.479	0.122	0.342	0.122	3.983
Max	13.840	0.153	0.347	0.122	3.992
Min	9.955	0.091	0.336	0.121	3.974
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	4.919	0.112	0.315	0.156	6.814
Max	5.801	0.221	0.365	0.159	7.997
Min	3.710	-0.070	0.116	0.155	6.374
Case 7	α_1 = 0, β_1 = 0.8612				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.055	0.176	0.337	0.122	3.987
Max	10.224	0.201	0.341	0.122	3.993
Min	8.283	0.149	0.332	0.122	3.981
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.711	0.191	0.376	0.154	5.877
Max	6.390	0.237	0.390	0.155	6.023
Min	5.390	0.162	0.365	0.152	5.739
Case 8	α_1 = 0, β_1 = 0.4755				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.059	0.192	0.336	0.122	3.988
Max	10.629	0.198	0.342	0.122	3.991
Min	8.248	0.148	0.332	0.122	3.980
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.739	0.192	0.376	0.154	5.863
Max	6.305	0.230	0.388	0.156	6.090
Min	5.334	0.165	0.363	0.152	5.744
Case 9	α_1 = 0, β_1 = 0.1169				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.058	0.176	0.336	0.122	3.987
Max	10.647	0.197	0.342	0.122	3.992
Min	8.164	0.147	0.331	0.122	3.978
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.750	0.190	0.377	0.154	5.870
Max	6.373	0.229	0.396	0.156	5.958
Min	5.400	0.158	0.367	0.153	5.735
Case 10	α_1 = 0, β_1 = 0				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.062	0.176	0.336	0.122	3.986
Max	10.142	0.198	0.341	0.122	3.991
Min	8.228	0.156	0.331	0.122	3.979
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.742	0.190	0.376	0.154	5.865
Max	6.381	0.217	0.388	0.156	5.985
Min	5.318	0.164	0.358	0.153	5.751

Table 23: Range for simulation case 6-10

Range					
Case 11	α_1 = 0.1169, β_1 = 0				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.082	0.175	0.337	0.122	3.987
Max	10.528	0.205	0.342	0.122	3.993
Min	8.247	0.151	0.332	0.122	3.979
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.723	0.194	0.378	0.154	5.882
Max	6.305	0.230	0.391	0.156	5.989
Min	5.271	0.161	0.363	0.153	5.762
Case 12	α_1 = 0.4755, β_1= 0				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	9.598	0.139	0.338	0.122	3.986
Max	11.318	0.114	0.344	0.122	3.994
Min	8.400	0.110	0.330	0.122	3.976
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	5.320	0.157	0.356	0.156	6.237
Max	6.015	0.210	0.386	0.158	6.443
Min	4.795	0.047	0.301	0.154	6.043
Case 13	α_1 = 0.8612, β_1 = 0				
Horizontal	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	10.335	0.113	0.339	0.122	3.985
Max	14.178	0.138	0.345	0.122	3.992
Min	9.024	0.084	0.331	0.121	3.972
Natural	ASPL	Assortativity	Global Tran	Local Tran	Mean Deg
Mean	4.780	0.066	0.302	0.158	6.920
Max	5.902	0.142	0.355	0.163	8.304
Min	3.550	-0.063	0.118	0.156	6.440

Table 24: Range for simulation case 11-13

Test statistics				
1962-1971	Estimate	Std Error	t-value	p > |t|
α_0	$1.124 * 10^{-6}$	$2.443 * 10^{-7}$	4.600	0.00
α_1	0.1671	$1.755 * 10^{-2}$	9.523	0.00
β_1	0.8137	$1.811 * 10^{-2}$	44.919	0.00
1967-1976	Estimate	Std Error	t-value	p > |t|
α_0	$6.991 * 10^{-7}$	$2118 * 10^{-7}$	3.301	0.00
α_1	0.0756	$1.043 * 10^{-2}$	7.434	0.00
β_1	0.9121	$1.149 * 10^{-2}$	79.373	0.00
1972-1981	Estimate	Std Error	t-value	p > |t|
α_0	$6.339 * 10^{-7}$	$2.233 * 10^{-7}$	2.838	0.005
α_1	0.0501	$6.693 * 10^{-3}$	7.481	0.00
β_1	0.9423	$7.638 * 10^{-3}$	123.379	0.00
1977-1986	Estimate	Std Error	t-value	p > |t|
α_0	$8.664 * 10^{-7}$	$2.929 * 10^{-7}$	2.958	0.005
α_1	0.0352	$5.689 * 10^{-3}$	6.189	0.00
β_1	0.9535	$7.601 * 10^{-3}$	125.435	0.00
1982-1991	Estimate	Std Error	t-value	p > |t|
α_0	$5.447 * 10^{-6}$	$1.216 * 10^{-6}$	4.479	0.00
α_1	0.0926	$1.208 * 10^{-2}$	7.660	0.00
β_1	0.8548	$2.105 * 10^{-2}$	40.616	0.00
1987-1996	Estimate	Std Error	t-value	p > |t|
α_0	$1.745 * 10^{-6}$	$5.016 * 10^{-7}$	3.480	0.00
α_1	0.0905	$1.249 * 10^{-2}$	7.242	0.00
β_1	0.8915	$1.639 * 10^{-2}$	54.396	0.00
1992-2001	Estimate	Std Error	t-value	p > |t|
α_0	$5.759 * 10^{-7}$	$1.971 * 10^{-7}$	2.922	0.00
α_1	0.0698	$1.054 * 10^{-2}$	6.624	0.00
β_1	0.9217	$1.066 * 10^{-2}$	86.983	0.00
1997-2006	Estimate	Std Error	t-value	p > |t|
α_0	$7.417 * 10^{-7}$	$3.125 * 10^{-7}$	2.373	0.00
α_1	0.0726	$1.007 * 10^{-2}$	6.739	0.00
β_1	0.9241	$1.113 * 10^{-2}$	83.038	0.00
2002-2011	Estimate	Std Error	t-value	p > |t|
α_0	$1.346 * 10^{-6}$	$3.141 * 10^{-7}$	4.287	0.00
α_1	0.0836	$9.514 * 10^{-3}$	8.791	0.00
β_1	0.9075	$9.909 * 10^{-3}$	91.579	0.00
2007-2016	Estimate	Std Error	t-value	p > |t|
α_0	$2.938 * 10^{-6}$	$5.290 * 10^{-6}$	5.553	0.00
α_1	0.1169	$1.328 * 10^{-2}$	8.802	0.00
β_1	0.8612	$1.429 * 10^{-2}$	60.628	0.00

Table 25: Test statistics for estimation of GARCH-parameters

9.3 R-codes

This is the code used to simulate the GARCH-processes, as well as the code that estimates the network statistics and the confidence interval. We have not included the codes we used to find the network statistics for S&P 500, as these are very similar, with just some minor modifications.

```r
rm(list=ls(all=TRUE))
library(fGarch)
library(igraph)
source("visibility.r")
source("HVG.r")

n = 100
mat=vector()
for (i in 1:n){
  spec = garchSpec(model = list(omega=0.000002938, alpha=0.1169, beta=0.8612))
  xyz = garchSim(spec, n.start = 500, n = 2500)
  xyz = as.vector(xyz)
  mat = cbind(mat, xyz)
}

#Matrices for saving the estimations
col.names = c("av. path length", "global transitivity", "assortativity", "mean degree",
              "loc clust", "H av. path length", "H global transitivity", "H assortativity",
              "H mean degree", "H Loc clust")
col.names2 = c("skewness", "kurtosis", "mean", "sd", "var", "max", "min")
stats = matrix(nrow=n, ncol=length(col.names1))
results= matrix(nrow=n, ncol=length(col.names))

#Network statistics
for (i in 1:n){
stats[i,2]= kurtosis(mat[,i], method="moment")
stats[i,1]= skewness(mat[,i])
stats[i,3]= mean(mat[,i])
stats[i,4]= sd(mat[,i])
stats[i,5]= var(mat[,i])
stats[i,6]= max(mat[,i])
stats[i,7]= min(mat[,i])
}
colnames(results)=col.names
colnames(stats)=col.names1

#Saving degree, local trans and deg. distribution for future use
degree = vector()
loc.trans = vector()
Hdegree = vector()
Hloc.trans = vector()
degdisN=vector("list",n)
degdisH=vector("list",n)

#Visibility graph
for (i in 1:n){
  x = mat[,i]
  adj = visi(x) #performs the visibility alorithm, creates an adj. matrix
  graph = graph.adjacency(adj) #creates a network out of the adj. matrix
  results[i,1]= mean_distance(graph, directed = FALSE, unconnected = FALSE)
  results[i,2]= transitivity(graph, type = "global", vids = NULL, weights = NULL,
                             isolates = "zero")
  results[i,3]= assortativity_degree(graph, directed = FALSE)
  deg = degree(graph,v=V(graph),mode="in")
  degree=cbind(degree, deg)
  degdisN[[i]]<-degree.distribution(graph,cumulative=T, mode="out")
  results[i,4] = mean(deg)
  local =transitivity(graph, type = "local", vids = NULL, weights = NULL, isolates = "zero")
  loc.trans = cbind(loc.trans, local)
  results[i,5] = mean(local)
}

```

```
#horizontal visibility graph
for (i in 1:n){
  x = mat[,i]
  adj = visiH(x) #performs the visibility alorithm, creates an adj. matrix
  graph = graph.adjacency(adj) #creates a network out of the adj. matrix
  results[i,6]= mean_distance(graph, directed = FALSE, unconnected = FALSE)
  results[i,7]= transitivity(graph, type = "global", vids = NULL, weights = NULL,
                             isolates = "zero")
  results[i,8]= assortativity_degree(graph, directed = FALSE)
  Hdeg = degree(graph,v=V(graph),mode="in")
  Hdegree=cbind(Hdegree, Hdeg)
  degdisH[[i]]<-degree.distribution(graph,cumulative=T, mode="out") #Degree distribution
  results[i,9] = mean(Hdeg)
  Hlocal =transitivity(graph, type = "local", vids = NULL, weights = NULL, isolates = "zero")
  Hloc.trans = cbind(Hloc.trans, Hlocal)
  results[i,10]=mean(Hlocal)
}

mean.stats = apply(stats,2,mean)
mean.results = apply(results,2,mean)

mean=as.vector(mean.results)
mean2=as.vector(mean.stats)
std = apply(results,2,sd)
std = as.vector(std)
std2=apply(stats,2,sd)
std2=as.vector(std2)
conf = (std*1.96)/sqrt(n)
conf2=(std2*1.96)/sqrt(n)
results = rbind(results,mean,std,conf)
stats = rbind(stats,mean2,std2,conf2)
```

9.4 Reflection notes

Espen Segberg

The main part of our thesis was time-series analysis using a new method that has been gaining momentum in recent years. The new methods consists of a tool for transforming time-series into graphs and use network analysis to study them further and give another perspective on the network. The reasoning for using this method is, contrary to many of the old methods, one can capture the non-linear properties that is often found in time-series. We focused on the visibility graph method proposed by Lacasa et al. (2008). The network is created from the time-series through something called the visibility criterion. Loosely explained this means that if two realizations in the time-series can “see” each other, no realizations in between have a higher value than either of them, a connection between the realizations are established. By following this criterion, we get a network or a graph from the original time-series. Thereafter we can analyze several different network statistics that tells us something about the underlying time-series. The motivation for our thesis was to find out if the visibility graph method was a useful tool in analyzing financial time-series. The method has been used extensively in other fields, but the use in finance have been relatively sparse. To first test out this method we used it on some real-world data, namely a time-series from the S&P 500, and collected the statistics value from real-world data to use as a baseline. We

then ran 13 different GARCH-simulations, which is a method for generating random time-series that is found to be similar to financial time-series found in the real world. The first simulation was set as to replicate the time-series extracted from the S&P 500 data and the rest we changed up the parameters to see if the visibility graph could separate between different processes in the time-series. Our first simulation case gave us pretty similar results as the real-world data in regards to the network statistics we were analyzing. On the surface this is a good result for us as this shows that the visibility graph method recognizes similar processes in time-series. As we ran the other 12 simulations and shifted parameters, we could see a trend in how the network statistics changed in the different simulations. Again, this tells us that the visibility graph can separate between processes in different time-series and is a good tool for analyzing financial time-series. We were also able to locate specific incidents that drives fluctuations in a time-series using the visibility graph method. As we noted in the thesis, our sample size from real-world data is small and consequently more research is needed to confirm our findings. All in all our results indicates that the visibility graph is a good tool to use in the financial world for people that analyze time-series of this nature.

Our findings can not be directly related to internalization in any way, but the method we have used may play a part in regards to internationalization. To take an example, as we mentioned in our literature review, the visibility graph method have been used to analyze time-series from the north-American natural gas price market to see what drives the price fluctuations. By doing this, one may be able to find some results that may help in dealing with emerging markets as they are maturing and to be better suited to deal with eventual problems that may arise or just be prepared for how the industry will possibly behave later on. The visibility graph is also a useful tool to locate important events, like economic, political or any other major events that play a significant role in fluctuation in prices in any given time-series. Therefore, another way this method can be used in terms of internationalization is to look at different time-series of all types, whether it be geographic, industry based or even compare from time-series from different fields of study. If one can find common events that plays a huge role in multiple time-series one can find solutions and cooperate between international borders or fields to find a solution if there is a problem. To sum up, our findings does not directly play a part in internationalization, but the method we have used may prove to be a useful tool to get a better understanding of different problems and how different time-series behave and in so may play an indirect role in problems regarding internationalization.

In regards to innovation we believe that our findings in and off itself are innovative. We have found evidence that this new method, the visibility graph, is a useful tool in analyzing financial time-series. This

provides a new way to analyze and understand different financial time-series and as mentioned is a good tool to capture non-linear properties that is not found in using the older methods for analyzing time-series. We may discover things we did not already now by looking at time-series in a different way especially in regards to what drives price fluctuations and how prices fluctuate.

It is hard to see that our findings may pose potential ethical challenges. We have found evidence for a new method to study time-series and we do not think there may be any ethical challenges at all considering this is just a research and analysis tool. Our findings may lead to a competitive advantage and/or risk mitigation. In regards to the completive advantages our findings may provide one may get a better understanding of how prices fluctuates for example, which is an extreme advantage in the financial world. If you are able to locate a trend or cycles in for example oil prices, you will have an advantage over competitors. Also by using the visibility graph, you may be able to reduce risk. This method is able to locate events that drives price fluctuations and, for example an oil company, may be able to locate specific events that drives the oil price and hence can hedge against these events occurring.

Sindre Skoglund

This thesis has focused on natural and horizontal visibility graphs, two algorithms for plotting a time-series into a complex network. We have applied these two algorithm to financial data, first by simulating data, and investigating what results were estimated in a controlled context, and then on real-life data, to see how those compared to what we simulated. The motivation behind studying these algorithms, has been to see whether they actually have the ability to differentiate between different types of financial time-series. To achieve this, we have simulated time-series with different GARCH-parameters, as well as estimated GARCH-parameters for the real-life data we used, and compared the estimated network statistics to these. Our findings suggest that especially the natural visibility graph algorithm combined with the network statistic of average shortest path length has the ability to differentiate between time-series, but that further testing on real-life data are necessary before we can conclude on whether this analysis tool works outside of a controlled simulation environment.

The knowledge we have gathered trough courses in the Master's program at UiA has been essential for us to be able to write this thesis. Especially courses in methodology and econometrics, as well as learning how to use the coding language R in Computational Finance has been important. At the same time, complex network theory has been something we had to learn from scratch throughout the process of

writing the thesis. Consequently, it has made the writing of this thesis a demanding task, but the experience of combining this new knowledge with what we already learned from financial and economic courses has been an interesting and enjoyable task.

The rest of this reflection note will cover three themes, namely internationalization, innovation and responsibility. The visibility graph algorithm is not the easiest topic to relate to these themes, to say the least, and whether the findings in this study will be affected by these themes is highly debatable.

However, it can be argued that internationalization clearly affects the financial markets, in the form of increased market integration. When countries become more internationalized, they often become more developed. It then makes sense that their financial markets also become more developed, and consequently become more integrated with the world economy as a whole. It was suggested by Zhuang et al. (2014)that the visibility algorithm can be used as a tool to measure market integration, and the tool can consequently also be relevant to internationalization. In our thesis, we have showed examples of how the network statistics will look like when the visibility graph algorithms are applied to S&P 500, an index consisting of some of the most internationalized companies in the world, listed on the stock exchange in one of the most developed countries in the world. With that in mind, it is possible that the findings in this thesis can be used as a basis if one wants to see whether an index is highly internationalized. Results obtained from a highly internationalized index should probably look approximately like the ones we have estimated in this thesis.

The connection between the visibility graph algorithms and innovation is more apparent. As this is a tool that can be used to analyze financial time-series, including for example stock prices and exchange rate, it can be a competitive advantage if someone were able to implement it effectively. Technical analysis is an important and widely used form of stock-price analysis, and innovation should be welcomed to stay ahead of the trends. Visibility graph analysis can possibly be used for this purpose, but further research is still needed before it can be utilized effectively.

The last theme to relate visibility graph to is responsibility. Right now, I would personally deem it as a stretch to see any responsibility issues in regards to the visibility graph algorithms application in finance, as it is still a tool under investigation. However, if someone were use these tool for any purpose right now, like for example portfolio management, and tried to sell a product based on this tools effectiveness, I believe it would be irresponsible, given how little we still know about it. As I stated, I believe further research surrounding this tool is still necessary, and that responsible use of it, would be to further

investigate how well it works, primarily in controlled environments, like we have done in this thesis, rather than trying to use it on real-life data prematurely.