Master Thesis

Department of Information and Computing Sciences

A model for generating synthetic financial interbank networks

Author
Mark Jan van Lieburg

Supervisors
Dr. I.R. Karnstedt-Hulpus
Dr. E.J. van Leeuwen

January 2023

Abstract

We have evaluated the literature available on several empirical financial networks to find the general structure and measures associated with these networks. Drawing inspiration from generative growth models, spatial models and a static financial model based on payoffs, we propose a novel model aimed at reproducing the characteristics of an interbank network. The results show our model is able to succeed in matching the topological characteristics and provides flexibility through the model parameters.

Contents

1 Introduction

In order to train models for analyzing financial data for applications such as fraud detection, real-world financial data is used. This data is often represented as a graph where the nodes represent participants (banks, companies, accounts, etc.) and the edges represent the existence of a transaction between a pair of nodes.

Two main problems occur when we want to use this real-world data for analysis. First, the information of individual nodes must be protected to respect their privacy. This leads to reduced public availability of financial data, as the owners of this data are generally not able or willing to share it [1]. Second, in applications such as training classifiers for fraud detection we need a balanced data set. However, since detected fraud cases are so small in numbers compared to non-fraudulent transaction we end up having unbalanced data sets [2]. These problems are not limited to financial networks, but also apply to other networking domains such as social networks or the internet.

The general solution to the problems of privacy, availability and flexibility is to use a model that can generate synthetic data which resembles the real-world data well enough such that it can be used to replace real-world data in research. Lim et al. [3] groups the process of generating realistic synthetic graphs into three tasks:

- **Graph generative model selection**. Selecting an appropriate graph model based on the real-world network.
- **Synthetic graph generation**. The computational task of using the model to generate a synthetic network.
- **Graph generation model validation.** Analyze whether the synthetic network resembles the real-world network well enough.

The main goal of this thesis is to apply this process to financial networks, with specific focus on interbank networks. We have evaluated the literature available on several empirical financial networks to find the general structure and measures associated with these networks. We have evaluated several existing graph models and the general characteristics of the topologies produced by these models. Combining some of the components of these models, we propose a novel model aimed at reproducing the topology of interbank networks. For our model, we draw inspiration from generative growth models [4] [5], spatial models [6] and a static financial model that focuses on core-periphery characteristics [7].

We start with some preliminaries and an overview of several earlier studies into the topology of financial networks and insight into the existing graph models in Section 2. Section 3 describes our proposed network generation model in detail. In Section 4 we will evaluate the model using several network measures and analyze the impact of the parameters of the model on the resulting topologies. In Section 5 we will conclude and indicate future work.

2 Background

To reach our goal of producing a model that can generate interbank network topologies we take a look at several areas of research. In Section 2.1 we introduce the network science preliminaries. In Section 2.2, we investigate the earlier studies that have been done regarding real-life interbank networks and their general characteristics. We look at existing models for financial data, a network formation model for interbank networks and random graph models in Section 2.3. These models will form the basis for our proposed model. In Section 2.4, we investigate the earlier efforts in the field of financial data generation.

2.1 Graph theory

Network A graph or network is a data structure describing a set of objects (nodes, vertices) that have a relation to each other (edges, links). This structure can be used to model many different real-world systems, such as social networks, the internet, road networks and biological networks. In formula, graphs are generally defined as $G = (V, E)$, where V represents the set of nodes and E represents the set of edges.

The two most basic parameters of a network are the number of nodes $|V|$ and the number of edges $|E|$. The edges or links can be directed, in which case an edge goes from one node and points towards another node. If an edge is undirected, the relation goes in both ways. The edges can also be augmented with properties, such as a weight.

Network properties A key property of a single node is its degree, describing the number of edges that are connected to this node. In case of a directed graph, this can be separated into its in-degree and out-degree.

A graph or network can have very different layouts while having the same number of nodes and edges. As an example, a star graph has one node connecting to every other node, while a list graph has a chain of nodes in which each node connects to one next node. We refer to a networks layout as its *topology*. To get more information about the network as a whole, we can compute metrics which tell us something about the topology.

Density The *graph density* is the ratio between the number of edges in a graph and the total number of edges a complete graph of size $|V|$ would have. It tells us how dense the network is. The density of an undirected graph is as follows:

$$Density(G) = \frac{2|E|}{|V| * (|V| - 1)}$$

Degree Distribution An important characteristic of a network is its degree distribution. Where the degree describes the number of links connected to a single node, the degree distribution provides the probability distribution of the overall degrees present in the network. This is formalized as $P(k)$ which denotes the probability that if we pick a random node in the network the degree of this node would be k.

The degree distributions most commonly found in real-life networks are highly skewed, where the large majority of the nodes have a low degree and a few nodes have a very high degree. These high-degree nodes act as hubs in the network. Commonly the degree distribution is described with a power-law: $P(k = x) \sim k^{-\alpha}$ where α is the exponent. It must be noted that it is up for debate whether the power-law is truly the best way to describe the degree distributions in real-life networks[8].

Clustering Coefficient The local clustering coefficient, defined for a single node, captures to what extent the neighbors of that node are also connected to each other. The definition for this measure is the density of the subgraph created by the neighbors of a node:

$$CC(i) = \frac{2|E_i|}{k_i * (k_i - 1)}$$

In this formula, E_i are the edges present between the neighbors of i and k_i is the degree of i.

To get to a metric that says something about the whole graph, the average clustering coefficient is defined as the average of all these local clustering coefficients over all nodes.

Assortativity The concept of assortativity describes whether nodes with similar properties tend to connect to each other. Taking the degree as property, we get to degree assortativity. If a high degree node is more likely to connect to another high degree node and a low degree node is more likely to connect to another low degree node, the graph is known to be assortative. If high degree nodes tend to connect more to low degree nodes and vice versa, we find the graph to be disassortative. Whether a graph is assortative or disassortative can be computed using the Pearson Correlation Coefficient, where if > 0 indicates an assortative graph, and if < 0 indicates a disassortative graph.

Distance measures The distance in a graph between two nodes is commonly described using the path length. This denotes the number of hops it takes to get from one node to another. As there can be multiple paths from node a to node b, most important is the shortest path length. To get a sense of the distances in the graph as a whole, a key metric for networks is the average shortest path length, averaging these shortest path lengths over all pairs of nodes. Additionally, the network has a diameter which is defined as the length of the shortest path of the pair of nodes furthest away from each other.

Core periphery structure and measures A core periphery structure is a structure in which the nodes in the graph can be divided into two groups: the core, which is densely interconnected to each other, and the periphery, in which the nodes have almost no links to each other. The periphery and core can have links to each other. This concept was first defined in Borgatti and Everett [9], and was accompanied by a method of finding this structure in graphs by finding the correlation between a graphs adjacency matrix and the adjacency matrix of an idealized core periphery structure. As noted in the analysis of some real-world interbank networks, it is common to find this structure in interbank networks.

A fast algorithm to detect a single core-periphery structure in a graph was proposed by Lip et al. [10] who uses the high degree nodes to get a core that maximizes its density, while the density of the periphery is minimized. While this gives proper results, it does not fully explain why a node belongs to the core beyond its degree.

Another approach for finding core-periphery structures is given by an algorithm based on the Surprise measure [11]. The Surprise measure is a quality measure of a partition of a network, and computes the probability of finding a number of links within a community of the network against these links appearing randomly. In this paper, a bimodular version of Surprise is defined, which is then used to find a network partition that optimizes this measure. A benefit of this approach is that it is less reliant on the degree of the nodes, but more on the link density of the partition that is being considered. A drawback of this method is that it is computationally much more expensive than the Borgatti & Everett and Lip methods.

Kojaku et al. [12] claim that under the configuration model, which fully preserves degree sequences, no significant two block core-periphery structure can be found at all. In other words, the core-periphery structures found by Borgatti and Everett are highly dependent on the degrees of the nodes, where high-degree nodes become hubs. Instead, Kojaku et al. indicate that significant core-periphery structures need more blocks or partitions than just the two core-periphery blocks. These other blocks can be communities or other core-periphery pairs. To find the core-periphery pairs, the authors introduce KMconfig, which can find multiple core periphery pairs in a single network.

To compare core-periphery structures between networks we will use measures such as the size of the core, the density of the core and the density of the periphery.

2.2 Empirical Financial Network Analysis

To be able to produce a model that can generate an interbank network we need to find the general characteristics of such networks. In literature we find several studies that provide insight into the network topology of financial networks, such as the seminal work of Soramaki et al. analyzing transactions from the Fedwire settlement system [13], which creates an interbank payments network. Similar works have subsequently been published analyzing interbank networks in other countries. We survey some of these below. The ones that we chose to include are picked based on how often they are mentioned and whether they model the edges as transactions, as these are the networks we are most interested in.

2.2.1 Financial Network Representation

With some minor exceptions and differences, these works in general consider the daily networks formed by the transactions between banks or financial institutions within a single country or settlement system, which can both be modelled with directed and undirected links. These links can be weighted in terms of the volume or value of the transactions that have taken place.

Soramaki et al. [13] analyze settlement data from the first quarter of 2004 from the US Fedwire system, and splits the data into daily networks. The nodes in these daily networks represent all commercial banks participating in Fedwire and the directed edges represent transactions between these banks. The links are weighted in volume and value of the transactions.

Rordam et al. [14] use a data set of a Danish large value payment system and split this data into a money market network and a payments network, which are analyzed separately. In the money market network the nodes represent banks and the links represent overnight money market loans. In the payments network the nodes also represent banks and the links represent settlements of customer driven transactions. The network has directed edges and weighted edges in terms of value and volume.

Kyriakopoulos et al. [15] analyse Austrian transaction networks where nodes not only represent banks, but also major financial players such as government accounts and insurance companies. The edges represent transactions between the nodes and are directed and weighted in volume, value and number of transactions. This study also considers networks formed on a monthly and yearly scale in addition to the networks formed on a daily scale.

Imakubo et al. [16] study the interbank network from Japanese settlement data in 1997 and in 2005. The nodes in this network are not only banks, but also other financial institutions that had an interbank transaction within the Japanese system. The edges represent transactions that have taken place in the network. The edges are undirected and weighted in volume and value. An interesting finding is that in the period between 1997 and 2005, the network evolved from being a star-shaped network in 1997 to a more decentralized network in 2005.

Embree et al. [17] consider Canadian interbank transaction data. This network only contains the largest financial institutions in Canada, which results in this study only considering activity at the core of the Canadian financial system. In their model of the network of the interbank system, they impose a minimum on the payment value for it to qualify as a link. The links are directed and unweighted.

Martinez et al. [18] use both payment settlement data and interbank exposure data to analyse the Mexican interbank network in the time period between 2005 and 2010. In both their networks, nodes represent banks. In the interbank exposure network, the edges represent exposures between banks (loans/credit lines). The payments networks has directed and weighted edges representing the daily accumulation of the value of the transactions that have taken place between banks.

Forte et al. [19] analyse the Argentine interbank network through overnight loans data from 2003 to 2017. The nodes represent financial institutions, including non-banks. The weighted and directed edges represent exposures between these institutions. In addition to topological analysis, they analyze the impact that macro-economic events, such as the financial crisis in 2008, has on the topological structure of the network. In addition, their work contains an excellent overview of topological features found in earlier studies of interbank networks.

2.2.2 Network structure

Network components Similar to other complex networks, interbank networks can be broken down into a Giant Weakly Connected Component (GWCC), Giant Strongly Connected Component (GSCC) and Disconnected Components (DCs) [20]. Soramaki et al. [13] and Rordam et al. [14] describe the GSCC as the core which consist of banks that are connected to each other via a directed path. Connected to these are the Giant In-Component and Giant Out-Component which contain the banks that have a direct path to or from the core respectively. Furthermore the banks on a directed path to or from the G-In and G-Out components are referred to as tendrils. These components together create the Giant Weakly Connected Component (GWCC). The GSCC contains on average 67% of the banks in the Danish interbank network [14] and 78% of the banks in the Fedwire network [13]. In terms of value and volume almost all is transferred within the core.

Core-periphery structure Imakubo et al. [16] find in their analysis of the Japanese network that it exhibits a core-periphery structure, in which the core is near complete and acts as a hub for the periphery. This notion finds further support in the analysis of the Mexican payments system [18], which concludes the core-periphery model is a better fit than a scale-free model. In an analysis of the Dutch interbank network [21], based on quarterly balance sheets, it is concluded this network also fits well to a core-periphery structure.

2.2.3 Topological features

Network size and density The number of nodes an interbank network contains varies greatly per study. Typical ranges are in the low hundreds, as shown by the Danish network having 60 nodes in the daily networks and 130 nodes in the yearly network [14], the Austrian network [15] having 423 nodes (on yearly scale) and the Japanese network having approximately 350 nodes. The Fedwire system [13] from the USA is the largest found with more than 7500 nodes.

Interbank networks are generally found to be sparse. The density of the Danish interbank network is found to be 0.083 on average, Fedwire is extremely sparse with a density of 0.0003. Reciprocity is another connectivity measure, specifically for directed networks, that measures the extent of links having transactions in both directions. The reciprocity of the Fedwire network is approximately 22% [13], the Danish interbank network analysis shows 22.8% [14], the Mexican Interbank network shows 42% [18]. These values in general are higher than the average connectivity and are also higher than the reciprocity that can be expected in random graphs.

We do not make further use of reciprocity as this is a directed graph measure and we have simplified the model to only generate undirected graphs.

Degree Distribution Degree distributions in financial networks follow heavy tailed distributions. Describing the distribution as following a power law distribution ($P(k = x) \sim k^{-\gamma}$) is most common in literature, with estimates of the coefficient γ being between approximately 2 and 2.5. The Fedwire system [13] is estimated to have a coefficient of $\gamma = 2.11$, the Japanese network is estimated to have $\gamma = 2.0$. The Austrian Network [15] is estimated to have a coefficient of $\gamma = 1.60$ for the daily network, $\gamma = 2.17$ for the monthly network and $\gamma = 2.4$ for the yearly network. Further evidence of interbank networks following a power law distribution is found in the Mexican interbank network [18], which is shown to not reject the power law hypothesis for most of the sampled days.

Some studies find other distributions are a better fit, such as the lognormal distribution in the Argentine network [19] or the exponential or negative binomial distribution in the Danish network [14].

Clustering coefficient The local clustering coefficient indicates the probability that two nodes that are neighbours of another node share a link between themselves. The average clustering coefficient is defined as the average of all local clustering coefficients. In the interbank network of Denmark [14] and the Fedwire network the average clustering coefficient is found to be approximately 0.5. The Mexican [18] and Canadian networks [17] show average clustering coefficients ranging from 0.7 to 0.85. These values are significantly higher than to be expected in fully random graphs. It is to be noted, however, that there can be a high disparity between nodes, with a significant portion of nodes having a clustering coefficient of 0 or 1 [13].

Interestingly, none of the papers mention the global clustering coefficient, which measures the number of connected triples as a ratio of the total number of possible triples, to describe the network. This would be interesting to see as this can significantly differ from the average clustering coefficient.

Assortativity The assortativity of a network indicates the dependence between the degree of a node and the degree of its neighbours. A network is known as assortative if nodes with a given degree are likely to link with other nodes with a similar degree, and known as disassortative if nodes links with neighbours with a dissimilar degree. The analysis of the Fedwire system [13] indicates that the network shows disassortative characteristics, shown by Pearson correlation coefficients and the correlation of the average nearest neighbour degree function (ANND) with a nodes degree. This is further confirmed by the study of Japans interbank network [16], also through the average nearest neighbour degree function, and the analysis of the Argentine network [19] through Pearson correlation coefficients.

For the degree correlation coefficient we find values typically in the range between -0.15 and -0.3. The Fedwire system has a correlation coefficient of -0.31 [13], the Argentine network -0.16 [19].

Distance measures In terms of distance interbank networks are found to exhibit the small-world property. The average path length, defined as the average number of links of all shortest paths between any two nodes, is typically found to be in the range 2 to 3. The Danish interbank network shows an average path length of 2.5 [14], the Fedwire system has an average path length of 2.6 [13], the Argentine interbank network averages 2.8 [19]. The Canadian large value system and the Mexican interbank network find even lower average path lengths, averaging 1.31 and 1.7 respectively.

The maximum distance between two nodes, also known as the networks diameter, is found to be 5.5 in the Danish interbank network [14], 6.6 in the Fedwire network [13], approximately 5 for the daily, monthly and yearly networks in Austria [15] and 7.9 in the Argentine network [19]. These values indicate that a general interbank network can be described as a small world network.

Correlations of topological features The timescale that is used for obtaining the graph has an impact on the network topology that is produced by the data. Kyriakopoulos et al. [15] study a data set containing transactions from the Austrian Real Time Interbank Settlement System. The study compares the average daily network, the average monthly network and the yearly network produced by the data. It is found that taking a longer time period makes for networks with a higher number of nodes and links, higher average degrees, higher maximum degree, higher clustering coefficient, and a higher global network efficiency. Time scale does not make a difference with respect to the diameter of the graph.

The study of the Argentine interbank system [19] provides insight into the impact of macro-economic events on topological features, made possible by the extensive time period that they perform their analysis on. They conclude that in times of financial distress there is both a decline in the network size, connectivity and reciprocity as well as in clustering levels. In times of economic recovery, the topological features quickly stabilized to similar values as before economic crisis. It was however noticed that during the recovery phase, there were higher values of reciprocity and less negative assortativity coefficients.

Distribution of value and volume In terms of volume and value, Soramaki et al. [13] find a power-law relation between out-strength of a node and its degree, where higher degrees lead to higher out-strengths. The existence of this relationship is further supported by the analysis of Japans [16] interbank network and the Argentine interbank network [19].

2.2.4 Summary of observations

Despite having some networks and metrics with slightly conflicting results such as different degree distributions, we find that the general interbank network can be described as follows: a sparse network of fairly small size (between 50-500 nodes), heavy-tailed degree distributions that are mostly described as power-law distributions, high average clustering coefficients (around $0.4-0.5$), disassortative, short average path lengths and diameter and a clear core-periphery structure with

big banks in the core and smaller clusters of local banks in the periphery. We aim to design a model with these observations in mind.

2.3 Models of network generation and formation

We explore the literature to find graph models that might be able to replicate the large-scale features that are commonly found in interbank networks. The literature provides two general directions of research. *Structure-driven models* generally take a graph or multiple graphs as input and try to replicate the structure as closely as possible. *Generative models* try to generate a graph by understanding how the graph evolves, through local rules such as preferential attachment. We survey these directions below. As there is a vast amount of literature for both directions, we limit ourselves to the most commonly mentioned models. For a more extensive overview of the field of random graph models, we refer to the excellent survey by Drobyshevskiy et al. [22].

2.3.1 Structure-driven models

dK Random Graphs dK-random Graphs [23] is a family of random graph models based on the definition of the dK-series. This is a series of probability distributions specifying the degree correlations within subgraphs of size d in graph G. For 0K, we have only the average degree of G, 1K captures the degree distribution of G, 2K captures the joint degree distribution of G, 3K captures the distributions of triangles and wedges in G, and this continues similarly for increasing d having a probability distribution for every non-isomorphic graph of size d. A dK-random Graph is a random graph that satisfies the dK property.

There are several different approaches to constructing graph realizations or characterizing the full space of graph realizations. Tillman et al. [24] provides an extensive overview of graph construction algorithms that satisfy the constraints of the $dK - Series$ and proposes additional models specifically targeting extra topological features, such as 2.25K graphs targeting the average clustering coefficient and 2.5K graphs targeting degree-dependent clustering coefficients [25].

For many real-world networks such as the internet and social networks, dK-random graphs with $d \leq 2.5$ can reproduce many local and global graph properties sufficiently. For more complex networks such as a network of the human brain, it can perform poorly in terms of local clustering effects, path distances and betweenness distributions.[26]

To be able to construct dK-Graphs, we need to provide corresponding degree correlations as input. Since we have practically no financial data that would sufficiently be able to provide target degree correlations beyond a basic 1K-graph, it might prove difficult to use this model effectively for our purpose of generating financial networks.

RMAT The *Recursive Matrix* (R-MAT) Model [27] proposes a procedural generator that recursively partitions the adjacency matrix of a graph into four equal-sized partitions (a,b,c,d) and adds edges into these partitions with non-equal probabilities. This results in a directed graph that is shown to exhibit power-law degree distributions, community structures and small diameter. The model can also be extended to support undirected graphs, bipartite graphs or weighted graphs. Since the model is easily parallelizable, it is able to generate large graphs very quickly.

As input, R-MAT takes values for the probabilities (a,b,c,d). This initial probability matrix can be interpreted as individual attribute similarities. To fit the R-MAT model to a real graph one can use the AutoMAT-fast fitting method [27] to estimate an initiator probability matrix that will lead to a graph resembling the real graph.

Stochastic Kronecker Graphs (SKG) A similar approach is taken in the Kronecker Graph model [28]. This model is based on the *Kronecker product*, which is a matrix operation. It takes as input an initiator adjacency matrix K_1, and then recursively applies the Kronecker product to produce successively larger graphs that are self-similar. To address staircase effects that emerge in the degree distribution, the Stochastic Kronecker Graph model is proposed. With this adaptation the initiator matrix contains probabilities that an edge is created, thus making it a probability matrix. An instance of the graph can then be sampled from this probability matrix. The KronFIT [28] method can be applied to find an initiator matrix that will lead to Kronecker graphs that are similar to the given real graph.

Moreno et al. [29] finds the SKG model produces very little variance in the generated graphs when compared to the real graphs within a domain. To combat this they propose the Mixed Kronecker Product Graph Model (mKPGM), which ties the realizations of intermediate Kronecker product steps to introduce edge dependencies. This shows a higher variance in generated graphs more representative of the real domain, while maintaining the expected graph properties.

2.3.2 Generative models

Barabási-Albert A fundamental model in the category of generative models is the Barabási-Albert model [4]. This model uses the concept of preferential attachment, a mechanism which lets new nodes connect to existing nodes with a probability based on the current degree of the existing nodes. This basic model creates undirected graphs and is shown to produce a power-law degree distribution with $\gamma = 3$ [4]. The model is not able to accurately reproduce distances, clustering or communities as found in real networks.

Preferential attachment underlies many subsequent random graph models such as the Holme and Kim model [30], which targets higher clustering. Soares et al. [31] proposes a model that exploits Euclidian distances between nodes in combination with preferential attachment. Krapivsky et al. [32] combine preferential attachment with a copying principle, leading to networks with a very low distance and densification over time.

Another interesting variation of the Barabási-Albert model is the Bianconi-Barabási model [33]. This model introduces the concept of node fitness into the preferential attachment, which can be used to explain why certain nodes acquire links more easily even if they are born at a later stage.

Nearest Neighbour The nearest neighbour model [5] takes a local, neighbour-based approach based on the intuition that two people who share a friend are more likely to also become friends. This model randomly either adds a new node or connects two nodes that have a common neighbour. It switches these rules based on a probability u, where a node is added with probability u and a potential connection is realized with probability $1 - u$. Model analysis shows that this model produces a power-law degree distribution with $\gamma \geq 2$. The most significant contributions of this model are the power-law distribution it produces for the clustering coefficient as a function of the vertex degree and its positive degree correlations (assortative).

Random Walk Random Walk [5] is, just like the Nearest Neighbour model, based on rules involving the neighbours of a node. Upon adding a new node to the graph, it randomly connects to an existing node, then proceeds to a neighbour of the existing node and connects with a given probability. This process proceeds as long as new connections are made, which leads to a random walk. This model shows high level of clustering and accurately captures a power-law distribution for the clustering coefficient.

Forest Fire The Forest Fire model [34] is also a generative model using local rules, specifically designed with two real-world phenomena in mind: graph densification over time and distance shrinking over time. The model works by adding new nodes one at a time, forming a link to an existing node and then burning outward links from the existing node, connecting to the nodes it discovers with a certain probability. The model takes two parameters, a forward burning probability and a backward burning ratio. In addition to having the desired properties of densification and shrinking diameters the model produces heavy-tailed degree distributions and communities within the graph. Even though there are only two parameters, the model is said to be capable of producing a wide range of both dense and sparse graphs with different degree distributions.

2.3.3 Spatial network models

In real world networks, it is not uncommon for the nodes and edges to have a relation to their spatial distance. As an example, in friendship networks the distance place a huge role in determining which connections are made. One common concept in spatial networks is that spatial distance can be seen as a cost that has to be overcome. In order to choose a node that overcomes a large spatial distance, the node has to offer something beneficial in some other form, like a high degree. This results in the observation that, generally speaking, long-distance links are formed towards hubs. The field of spatial network models studies the topological impact of adding space as a parameter

when forming links and the different ways to incorporate space into a network model. For an extensive study of spatial networks we refer to a survey done by Barthelemy et al. [35].

Since we are interested in creating a generative growth model, we specifically look into the model as proposed in [6]. This study considers the interplay of a crossover between a preferential attachment network and a spatial network. In formula, the edges are chosen according to this probability:

$$p_{i\to j} \propto \frac{Z(k_j)}{\Delta(d_{ij})} \qquad \text{(Equation 2.1)}$$

In this formula, $Z(k_j)$ is a function of the connectivity of node j, $\Delta(d_{ij})$ denotes a function of the spatial distance between node i and j. There are different ways of defining the distance function according to different distributions. For their analysis, this study chooses to define the distance function as $\Delta(d) = e^{d/r_c}$ where d is the Euclidian distance between two nodes and r_c is a finite scale. The distances are given to nodes with a uniform distribution in a space of size L.

The results of simulations of this model show that, when ratio $\eta = \frac{r_c}{L}$ is more than 1, the distance is irrelevant and will the model will reduce to pure preferential attachment. When the ratio is less than 1, the study finds increasingly higher clustering coefficients and positive degree correlations when distance becomes more important.

2.3.4 Heterogeneous financial network model

Since the aforementioned models do not produce a core-periphery structure, we now focus on a static financial network model that specifically targets this feature. In 't Veld et al. [7] propose an interbank network model aiming to explain the empirical findings of the core-periphery structure in interbank networks. Central to their model is the concept of *intermediation*. Banks can be assumed to trade only along established trading relationships through direct links or indirectly through intermediation. In their model, banks can receive a payoff by directly connecting to another bank, of which the size of the payoff π is related to the size α of the bank and the size of the bank it is connecting to. Banks can also receive a payoff by being a *middleman* connecting two other banks. In this case, the size of the payoff depends on the sizes of the banks that are being intermediated between and the number of other middlemen available. The share each middleman receives can be controlled through a parameter δ, known as the *competition level*. Furthermore, the model imposes a constant cost c on each link that is created. The payoff function is a summation of three components. The first component represents payoffs gained by direct linking, the second component represents payoffs gained through indirect linking and the third component represents the payoffs gained through being an intermediator between two other banks. The payoff function is formally defined as follows:

$$\pi_i(G,\delta,c) = \sum_{j\in N_i^1(G)} (\frac{1}{2}\alpha_i\alpha_j - c) + \sum_{j\in N_i^2(G)} \alpha_i\alpha_j f_e(m_{ij}(G),\delta) + \sum_{k,l\in N_i^1(G)|G_{kl}=0} \alpha_k\alpha_l f_m(m_{kl}(G),\delta) \qquad \text{(Equation 2.2)}$$

In this function, N_i^d is the set of nodes at distance d from node i. The notation $m_{ab}(G)$ refers to the number of middlemen that exist between node a and node b, in other words the number of distinct length-2 paths between those two nodes. The function $f_e(m,\delta)$ and $f_m(m,\delta)$ then refer to the share that the endpoints and the middlemen of such indirect trading relations get respectively. These functions are based on a bargaining protocol [36] and will be relevant in our own model described later. The formulas are given as follows:

$$f_e(m,\delta) = \frac{m-\delta}{m(3-\delta)-2\delta} \qquad \text{(Equation 2.3)}$$

$$f_m(m,\delta) = \frac{1-\delta}{m(3-\delta)-2\delta} \qquad \text{(Equation 2.4)}$$

To form the network, this model uses the concept of network stability in which no node can improve its payoff by making another new connection or by deleting a connection. To obtain this stability, each node takes its *best feasible action* in which it either connects to a new set of nodes that improves its payoff or deletes a set of edges that will improve its own payoff. The network is deemed stable if no node can take a feasible action that improves its own payoff.

The main result and takeaway from this model is that when forming a network according to these payoffs, a stable core-periphery network cannot form if all banks are given the same size (homogeneous). The core-periphery structure can and will form, for certain parameters, if the sizes of the banks are dissimilar and a distinction is made between big banks and small banks (heterogeneous). The model can also be extended with a more dynamic version in which the resulting payoffs from one stabilization round feed into the bank sizes of the next stabilization round. The authors show that in this dynamic case the core-periphery structure can emerge as well, even when having homogeneous bank sizes in the initial round.

This model is highly deterministic and theoretical, and will produce near perfect core-periphery structures for certain parameter ranges indicated in their paper [7]. The resulting networks therefore do not resemble real interbank networks in any metric other than the existence of a core-periphery structure. Additionally, there is no presence of network growth in this model as the introduction of new nodes would destabilize the network and require every node to reconsider its positions.

2.3.5 Summary of observations

Structure-driven models, while known to be able to accurately capture many topological features simultaneously, often require extensive real data to be able to fit the model. In addition, the structure-driven models do not tell us much about how a network grows and evolves or what underlying mechanisms drive the network. The generative models are much simpler models in terms of parameters, yet give us more insight into the rules that lead to the networks characteristics.

As described before, none of the discussed generative models is able to sufficiently replicate all the desired network characteristics at the same time. In the network model that we propose and discuss in detail in the following chapter, we incorporate concepts from the generative models and the heterogeneous financial network model with the purpose of obtaining networks that have all the desired properties as described in Section 2.2.4.

2.4 Financial Data Generation

Notable earlier efforts in financial data generation are the PaySim [37] and the AMLSim [38, 39] systems. The data generated by these systems is transactional data.

PaySim PaySim [37] focuses on generating synthetic data based on mobile money systems, mostly popular in developing countries. To generate the data, PaySim uses a multi-agent based approach where each client has a profile containing client characteristics such as transaction and balance limits. After generating these clients, it will generate transactions with several transaction types such as payments and transfers. PaySim is able to match transaction frequency patterns to existing data sets. Their method does however not consider any network structure between the clients that would be present in the real sets, resulting in lots of disconnected components.

AMLSim AMLSim is a multiagent based simulator that is specifically targeted towards anti-money laundering. Their method starts off with generating a network topology, and then using PaySim to generate transactional data on top of the generated topology. The network generation model used in AMLSim is the Configuration Model [40], which takes a fixed degree sequence as its input. This graph model can exactly replicate the given degree sequence, however it constructs graphs that may contain self-loops and multiple edges between the same nodes. This model essentially generates the previously discussed 1K-graphs [23], which generally is not able to match the distances and clustering levels and degree correlations found in real-life networks.

AMLSim also provides fraud patterns which it inserts into the graph and in the generated transaction logs. It would be interesting to investigate how the attribute generation part of this simulator performs when using the topologies we plan to generate using the generative graph models.

3 Interbank growth model

We propose an interbank model aimed at reproducing the network characteristics found in empirical networks, with as main focus the core-periphery structure. At its core, the model is a random generative model with a growth structure similar to the previously discussed Nearest Neighbor model [5]. The first parameter we have is the number of ticks T. Starting from a star graph with $m + 1$ nodes, each execution tick lets the model switch between expanding the network by adding a new node (with probability p) or expanding the connectivity by adding m new links (with probability $1-p$). We choose this structure as opposed to a standard Barabasi-Albert [4] structure to allow for nodes to add edges not only at birth, but also at a later stage in the development of the network. This results in the opportunity of nodes that are born at a later tick to become big banks as well through finding beneficial connections.

In the ticks where nodes are added, the new node also connects to m existing nodes. In the ticks where the connectivity is expanded, we choose one node with a probability based on its overall payoff, which will be discussed below. The edges that are chosen will be selected with a probability based on the additional payoff that will be provided.

3.1 Model components

We now discuss the approach to determining the probabilities that will be used to generate the connections. For every link that is considered, we calculate the additional benefit that this link would give to the node u initiating that link. We based the calculation of this additional benefit on the payoffs as found in the heterogeneous payoff model [7]. The benefits can be split into three different components: direct payoff between the endpoints of the link, the additional nodes that u can reach through the newly connected node, and the middlemen benefits that u will get by expanding the reach for the newly connected node. These components are discussed in depth below.

Direct benefit We define the benefit of a direct link as the product of the degrees of the nodes that are being linked, more formally:

$$\text{Direct}(\text{u}, \text{v}) = \text{Degree}(\text{u})\,\text{Degree}(\text{v}) \qquad \text{(Equation 3.1)}$$

This definition is comparable to the direct payoff as defined in In 't Veld et al. [7], with the main difference being that we take the degree as the size of the bank instead of using a separate α as parameter of the size for each bank. The purpose of using the degree versus α is to simplify the model and eliminate the need for the parameter α. While using the degree might give a bit less control over the initial sizes of banks, it serves its purpose of creating heterogeneity amongst all banks and creating a preference towards connecting to well-connected nodes. Additionally, we do not split the resulting benefit between the banks, also with the purpose of simplifying the model. It might be more realistic to split the benefit, however these benefit values have no basis or connection to any real life values, thus are only relevant in its relation to each other. Therefore we believe the resulting networks will not be changed in any significant way by making this simplification.

Note that we removed the cost component from this part of the equation. This component will be added back in through a different mechanism in which the cost of links is dependent on the size of the node and the distance between nodes.

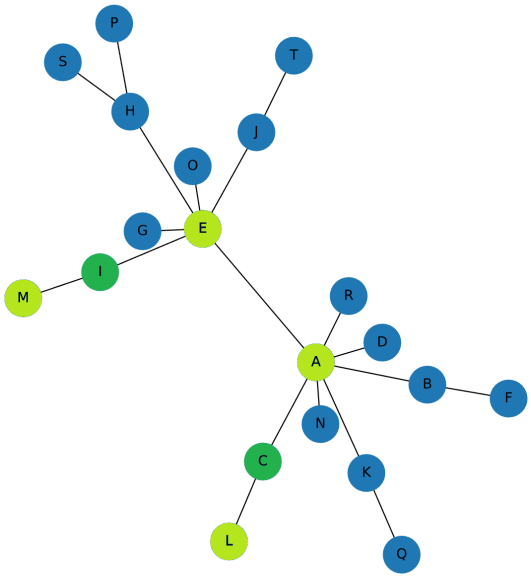

Figure 3.1: Visualisation of the direct benefit obtained for node C and node I

Figure 3.1 provides a visualisation of the direct benefit. When we consider a link from node C to node I, the degrees of C and I are multiplied to determine the direct benefit, which in this case amounts to 4.

Reach benefit We define the reach benefit from adding a link from u to v by the additional indirect benefits that u can get by connecting to v. This includes not only the benefit of the new nodes it can reach through v, but also the extra benefit it will gain from the potential extra path to a node that u already had indirect trading relations with. This extra benefit comes from the competition that occurs between the middlemen if multiple indirect paths to a node exist.

In words, the Reach function considers all neighbors of v that are not neighbors of u (as the neighbors of u already have a direct link to u) and computes the additional benefit of the potential indirect link through v. We again use the degree to obtain heterogeneity in a similar way to the direct benefit. The portion u receives from this benefit is obtained using the bargaining protocol that is also used in the model from In 't Veld [7]. Since we want to give more weight to nodes that have no established trading relationship with v at all, we subtract the current situation. This gives us only the *additional* benefit that this indirect link will provide. Formally, we define the reach function formally as follows:

$$\text{Reach(u, v)} = \sum_{w \in N(v), w \notin N(u)} \text{Degree(u)}\,\text{Degree(w)}\left(\frac{(m_{uw}(g)+1)-\delta}{(m_{uw}(g)+1)(3-\delta)-2\delta} - \frac{m_{uw}(g)-\delta}{m_{uw}(g)(3-\delta)-2\delta}\right) \quad \text{(Equation 3.2)}$$

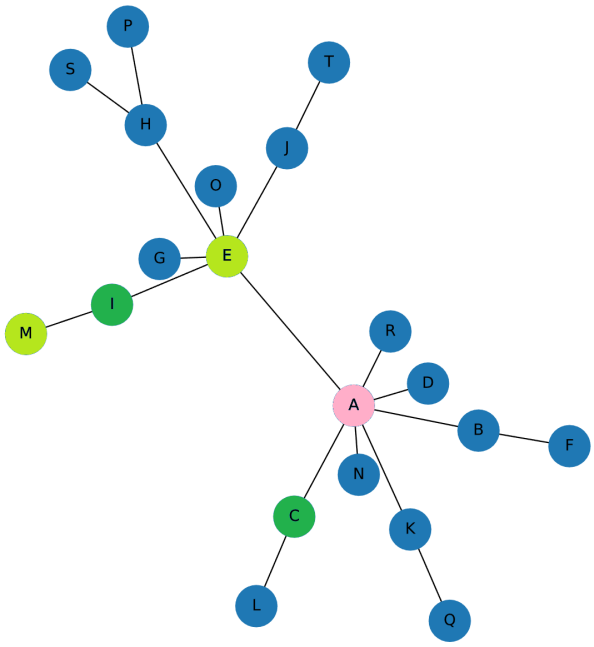

Figure 3.2: Visualisation of the reach benefits for node C considering a link to node I. Node A is highlighted as this is will be of influence as a competing middleman.

Figure 3.2 provides a visualisation of the Reach benefit. We again consider a link from node C to node I. Node I has two neighbors, M and E, highlighted in light-green. When connecting to node I, C obtains new indirect paths to M and E through node I. M was not reached before, so I will be the only middleman on the path from C to M, resulting in an even split. E was reached before through node A, highlighted in purple. In the new situation, A and I will have to compete for benefits, which because of the bargaining protocol will result in a larger share for the endpoints. This additional benefit is then included in the Reach component.

Intermediation benefit We define the intermediation benefit from adding a link from u to v by the additional indirect benefits that u will get as a middleman, intermediating between v and the neighbors of u. Again, this function uses the degrees of the nodes that will be intermediated between to establish benefits. We use the bargaining protocol again to determine the share u gets from intermediating between v and the neighbors of u. This function is formally defined as follows:

$$\text{Intermediation(u, v)} = \sum_{w \in N(u), w \notin N(v)} \text{Degree(w)}\,\text{Degree(v)}\,\frac{1-\delta}{(m_{wv}(g)+1)(3-\delta)-2\delta} \tag{Equation 3.3}$$

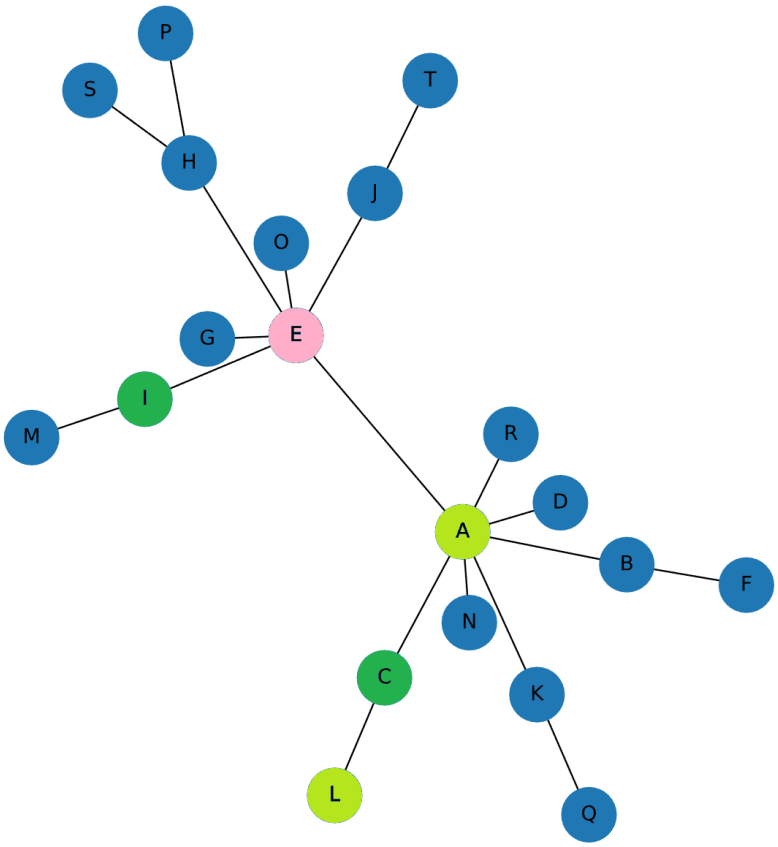

Figure 3.3: Visualisation of the intermediation benefits obtained for node C when considering a link to node I. Node E is highlighted as the competition with this node will diminish the intermediation benefits obtained by indirectly linking A and I.

Figure 3.3 gives a visualisation of the Intermediation benefits. Considering the potential link from node C to node I again, the Intermediation benefit gives us the benefit that C will get as a result of positioning itself as a middleman for I and the neighbors of C. A and L are the neighbors of C. For the indirect link L to I, there are no other middlemen, resulting in an even split for L, I and C. For the neighbor A, there is already an indirect link from I to A through E. Therefore C and E will have to compete for middlemen benefits and the share will be decided according to the bargaining protocol.

Distance and degree cost To replace the cost component we add a different mechanism which involves the distance between two nodes and a degree cost, making this a spatial network model. The intuition behind this is that spatial networks tend to cause the formation of hubs, which might work well in a banking environment, where the big banks end up being hubs for all the small local banks. It would also make sense that in case a bank has two connection options with exactly the same payoff benefits it would prefer the bank in closer proximity.

To get an even more pronounced difference in behaviour of small and big banks, we scale the distance by a component related to the relative degree of a node. The intuition is that big banks will have more resources compared to small banks, making the geographical distances a lot less relevant for them. This will cause big banks to have connection probabilities that are mostly based on maximizing financial benefit, while the small banks will have connection probabilities that are more based on distance and finding a proper balance between distance and financial benefit.

This distance and degree cost is formalized as follows:

$$\Delta(\mathrm{u}, \mathrm{v}) = e^{\frac{d_{uv}}{r_c}(1-\mathrm{RelativeDegree(u)})} \qquad \text{(Equation 3.4)}$$

In this function, d_{uv} is the Euclidean distance between node u and node v. The size of the geographical space can be set through parameter L. r_c is a scaling parameter that can be used to

tune the importance of the distance component in the model. When having $r_c > L$, the distance will have no impact at all and the model will reduce to a model without distance. The relative degree of node u is the degree of node u divided by the maximum degree present in graph G, so this value will be in between 0 and 1.

Complete components Taking all equations together, we obtain the main weight component that will be used to choose edge probabilities. We will implement the spatial component in a similar way as found in Barthelemy et al. [6], where we consider the sum of the Direct, Intermediation and Reach functions (Equations 3.1, 3.2 and 3.3) as the Z function of connectivity and the distance and degree cost (Equation 3.4) as the Δ function. Formally, the weight component is as follows:

$$\text{Weight(u, v)} = \frac{\text{Direct(u, v)} + \text{Reach(u, v)} + \text{Intermediation(u, v)}}{\Delta(u, v)} \qquad \text{(Equation 3.5)}$$

In the densification steps, where the connectivity is expanded, we choose a random node u with a probability based on the overall payoffs that it has accumulated, which we will refer to as $Fitness(u)$. In this case, the Reach benefit is defined slightly differently since we only have to consider the current situation:

$$\text{ReachBirth(u, v)} = \sum_{w \in N(v), w \notin N(u)} \text{Degree(u)}\,\text{Degree(w)}\,\frac{m_{uw}(g) - \delta}{m_{uw}(g)(3 - \delta) - 2\delta} \qquad \text{(Equation 3.6)}$$

This overall fitness for node u then is the sum of benefits obtained from all neighbors of u. Formally:

$$\text{Fitness(u)} = \sum_{v \in N(u)} \text{Direct(u, v)} + \text{ReachBirth(u, v)} + \text{Intermediation(u, v)} \qquad \text{(Equation 3.7)}$$

3.2 Complete model

Putting everything together we obtain our main proposed model. We indicate a single step in the model as a *tick*. The growth of the graph can be limited by the number of ticks T.

3.2.1 Input parameters

p	Probability a tick adds a node, $1 - p$ probability a tick adds an edge
m	Number of edges added at each tick
δ	Competition level between middlemen
L	Distance scale
rc	Typical scale

3.2.2 Graph generation process

1. Initialization.
    - Create a base graph as a star graph n = m + 1
    - Place all nodes in a 2D space of size $L * L$ distributed uniformly at random
2. Growth. With probability p
    - Create new node u
    - Select uniformly at random position in 2D space
    - Connect u to m existing nodes with for each node v a probability proportional to Weight(u, v) as in Equation 3.5

    With probability *1-p*
    - Pick a random node u with for each node a probability proportional to Fitness(u) as in Equation 3.7
    - Connect u to m existing nodes with for each node v a probability proportional to Weight(u, v) as in Equation 3.5

3.2.3 Model variations

To be able to analyze the impact of some of the different components within the model we define four variations of the model. We give the functions that are adapted when choosing an edge in the generation process. The functions for picking a node (through overall payoff) and picked edges at birth are adjusted accordingly. The different variations are defined as follows:

Model without distance and degree cost This variation puts the emphasis on finding what networks are able to form when focusing on the payoff structures only. In this variation, the edges are picked with weight function:

$$\text{WeightNoDist}(u, v) = \text{Direct}(u, v) + \text{Intermediation}(u, v) + \text{Reach}(u, v)$$

Model without degree cost This variation can help us test the impact of adding the degree cost, which we expect causes the big nodes to be able to form long-distance links more easily than small nodes. The definition of the weight function in this variation is as follows:

$$\text{WeightNoDC}(u, v) = \frac{\text{Direct}(u, v) + \text{Intermediation}(u, v) + \text{Reach}(u, v)}{e^{\frac{d_{uv}}{r_c}}}$$

Model without payoffs This variation gets at finding the difference between using payoffs and just using a degree-based preferential attachment. Also this can help analyze the importance of distances in the model. This variation defines the weight of the edges as follows:

$$\text{WeightNoP}(u, v) = \frac{\text{Degree}(v)}{e^{\frac{d_{uv}}{r_c}(1-\text{RelativeDegree}(u))}}$$

Model without payoffs, distances and degree cost This variation provides an even more stripped down version that we can compare with, using only the degree to find its edge weights. To make it formal:

$$\text{WeightNoPNoD}(u, v) = \text{Degree}(v)$$

3.2.4 Model Implementations

There are several well-known graph libraries containing implementations for existing graph generators, such as the SNAP library [41], NetworkX [42] and iGraph [43]. Besides these common generators, the libraries provide extensive graph manipulation and graph analysis functionality. For our purposes we used NetworkX to implement the models. We use the Python package *powerlaw* [44] to analyze the degree distributions of the resulting networks. We use the Python package *cpnet* [45] to analyze core-periphery structure in the resulting networks.

3.2.5 Discussion

At birth, the node has no connections yet, so it will not be able to provide any intermediation targets to other nodes. Therefore we leave out this component when determining which edges to choose at birth.

Impact of the initialization graph Another minor issue is that the spatial position of the nodes from the base graph is determined after they are connected as a result of generating a star graph. This means that this might create edges that span a distance between two low degree nodes that would almost never connect if they were to come into existence at a later stage in the graph generation process. The size of this star graph is very small compared to the end graph, so the impact of this problem should be limited.

4 Model analysis

4.1 Theoretical analysis

Number of nodes and edges Since the model is based on a number of ticks that can either add an edge or a node, we can not obtain an exact number of nodes. Starting from a base graph with size $m + 1$ and expecting to go through $T * p$ ticks that add a node, the expected number of nodes the resulting network will have is $T * p + m + 1$.

Since the model adds both m edges when it goes trough a tick of adding nodes or a tick of adding edges and has exactly m edges from the base graph, the expected number of edges will be $m(T + 1)$. This gets us to an expected density of $\frac{2m(T+1)}{(T*p+m+1)(T*p+m)}$

4.2 Fitting to empirical data

Having a model that has six parameters means we have to deal with an extremely large parameter space. Additionally, some of the parameters are heavily correlated. As an example, all the parameters T, p and m have a direct impact on the density of the resulting graph, yet there are many different combinations of these three parameters that result in the same expected density. These combinations however might result in very different graphs, as there can be strong differences in the timing of when edges are added or how many edges are added by a single node. Therefore, it can be hard to determine what kind of graphs the model is exactly capable of and how the different parameters have an impact on the resulting graphs.

Since time and resources are limited in this project, we are not able analyze the full extent of the parameter space. To get to an analysis of the models general behaviour and impact of its parameters, we try to fit the model to some of the empirical networks discussed previously. We fit it to a somewhat small network, specifically the (daily) Danish network [14], and a medium-sized network, specifically the Japanese network in 2005 [16]. The target values are summarized in table 4.1:

	Denmark	Japan
N	60	354
Density	0.083	0.027
Clustering Coefficient	0.5	-
Average Path Length	2.5	2.5-3.0

Table 4.1: Target values for synthetic graphs

After fitting some graphs to these values, we pick a few combinations of T, p and m to further analyze the parameters δ and r_c. Additionally, we will used these settings to analyze the core-periphery structure of the resulting networks and compare them to the alternative, simplified versions defined previously.

4.2.1 Danish network

Density parameters Starting with fitting to the Danish interbank, we run several combinations of T, p and m for our main model that approximately target the 60 nodes with a density of 0.083. For every setting, we run the model 100 times to obtain the network characteristics. The networks in Table 4.2 are run with $\delta = 0.5, L = 10, r_c = 2$. These values are chosen as early results showed these parameters to be somewhat in the middle of what the model is able to produce. These individual parameters will be analyzed deeper next.

T	p	m	N	|E|	Dens.	CC	Assort.	APL	AvgΔ	MedΔ	Plaw α
200	0.30	1	62.1	192.5	0.105	0.425	-0.268	2.43	7.18	6.32	3.25
170	0.35	1	61.0	164.2	0.093	0.378	-0.256	2.53	7.03	6.16	2.99
150	0.40	1	61.4	146.0	0.081	0.333	-0.248	2.62	6.83	5.95	3.01
133	0.45	1	61.9	129.8	0.070	0.274	-0.233	2.74	6.55	5.59	3.00
120	0.50	1	62.9	118.4	0.062	0.230	-0.214	2.86	6.40	5.42	2.79
110	0.55	1	63.8	109.1	0.055	0.186	-0.195	2.97	6.13	5.16	2.90
100	0.60	1	62.5	99.4	0.053	0.154	-0.175	3.06	6.07	5.10	3.09
100	0.60	2	63.4	197.1	0.102	0.465	-0.173	2.39	6.32	5.38	2.86
92	0.65	2	63.4	181.9	0.093	0.429	-0.163	2.45	6.15	5.23	2.88
85	0.70	2	63.2	169.1	0.087	0.403	-0.143	2.52	5.98	5.05	2.94
80	0.75	2	63.4	160.2	0.082	0.369	-0.124	2.59	5.83	4.90	3.05
75	0.80	2	63.4	150.6	0.077	0.340	-0.115	2.66	5.62	4.75	3.17
66	0.90	2	62.4	133.4	0.070	0.292	-0.113	2.79	5.37	4.57	3.37

Table 4.2: Results of varying T, p and m. $L = 10, \delta = 0.5, r_c = 2.0$

There are several interesting things to be found in this table. First off, we do not find a clustering coefficient that is as high as the empirical Danish network in any of these results, except for some outliers. Also, the clustering coefficient drops off quite heavily once the density decreases. This drop off is less present in the models with $m = 2$ compared to the models with $m = 1$. As an example from the table, the model $T = 170, p = 0.35, m = 1$ ends up with the same density as the model $T = 92, p = 0.65, m = 2$, yet the clustering coefficient of the latter is higher.

For the assortativity coefficient the model also produces different results for different values of m. Graphs produced with $m = 2$ are less disassortative, meaning the small banks connect more to other small banks than with $m = 1$. Also the average path lengths are slightly shorter with $m = 2$ compared to $m = 1$, however these all compare well to the path lengths found in the Danish network.

To further analyze the distance component of the model we also look at the average spatial distance that is spent by a link. It is important to note that these values are all relative to the scale L, so these values should only be looked at in its relation to other models with the same scale value. The results in table 4.2 show that models with $m = 2$ are connected much more local spatially compared to $m = 1$. Additionally, when the density of the graphs increases, so does the average and median distance spent by the edges. A possible explanation for this is that when the density increases the average degree will also increase, resulting in it being easier for the average node to overcome the distance cost through the degree cost mechanism.

Next we look at the average α found when fitting a power-law distribution to the degree distribution. With the lowest average at $\alpha = 2.78$ and most networks of this size averaging an α of around 3, we find these values to be a bit higher than expected when looking at the empirical networks.

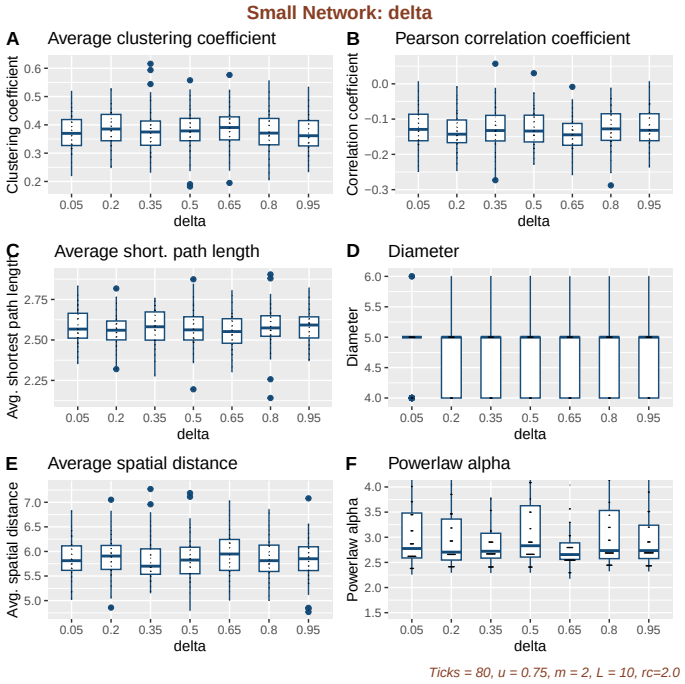

Figure 4.1: Graph metrics when varying delta in the networks with $T = 80, m = 2, p = 0.75$

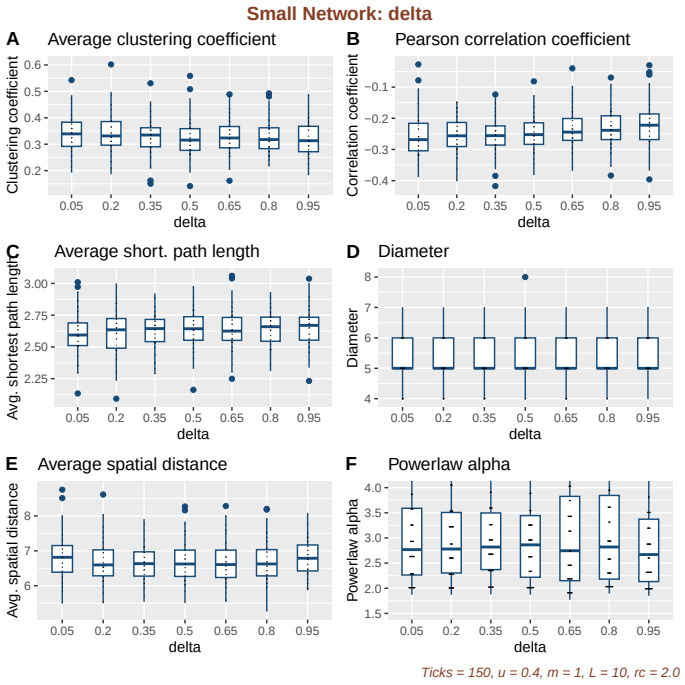

Figure 4.2: Graph metrics when varying delta in the networks with $T = 150, m = 1, p = 0.4$

Impact of δ To analyze the impact of parameter δ, we select two combinations of settings: $T = 150, p = 0.4, m = 1$ and $T = 80, p = 0.75, m = 2$. These are chosen because they approximate the density the closest. Again, we used $r_c = 2.0$ and $L = 10$ to obtain these networks. In terms of trends that result from varying the δ parameter, we expect them to be similar for the other combinations of settings in Table 4.2. We vary δ in the range $0.05 - 0.95$ with a tick size of 0.15.

Interestingly, varying δ for networks of this small size seems to show no impact or trends in the model setting $T = 80$, as shown in figure 4.1. A possible explanation could be that, since the impact of δ through middlemen benefits is subtle, the number of ticks is too small for any patterns

as a result of varying δ to emerge. In the network with $T = 150$, for which results are shown in figure 4.2, we find a slightly decreasing clustering coefficient and a slight increase in the correlation coefficient as delta increases.

In terms of distances, the average shortest path length and diameter results are well in line with the findings of the empirical Danish network [14]. Varying delta does not seem to change the values of the distances. In terms of the power law fit exponent α we find values in between 2.75-3.0, but again no real trend when varying delta in both combinations of settings.

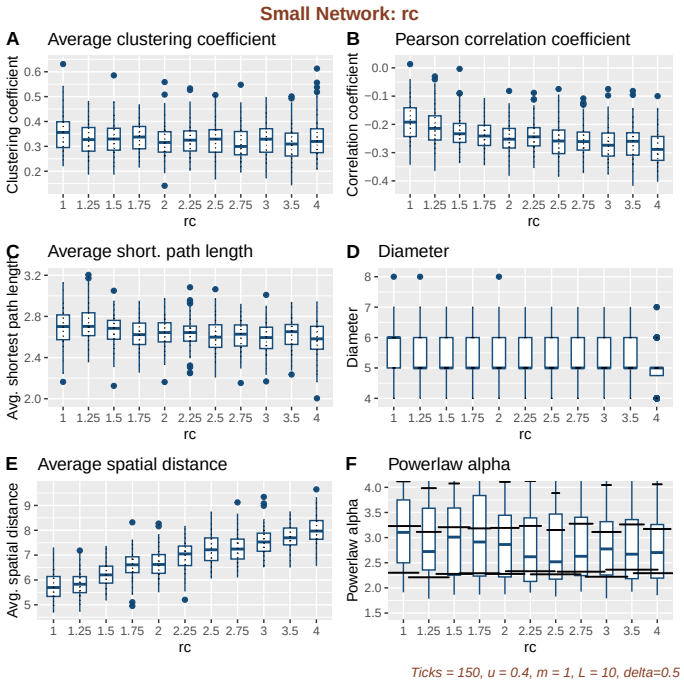

Figure 4.3: Graph metrics when varying r_c in the networks with T=150, p=0.4, m=1

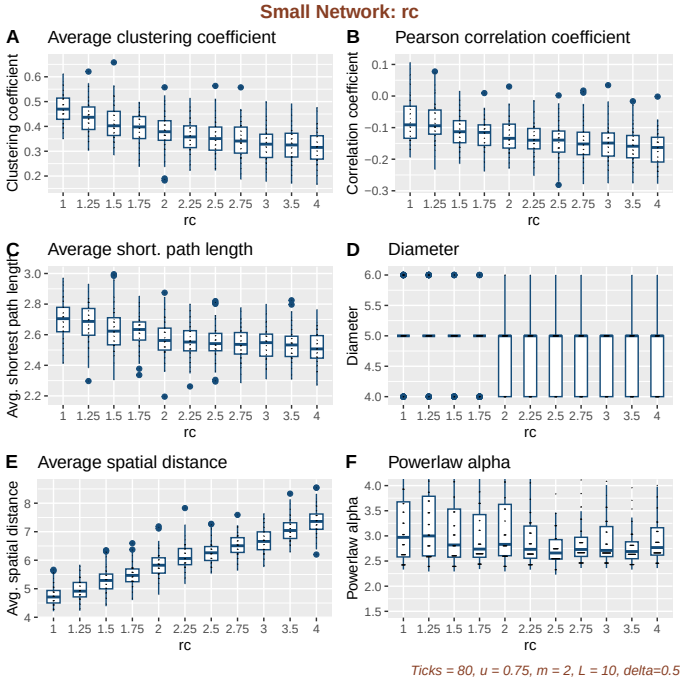

Figure 4.4: Graph metrics when varying r_c in the networks with T=80, p=0.75, m=2

Impact of r_c We similarly analyze the impact of r_c by varying this parameter using the same fixed two combinations of settings as above. In theory, the value of r_c is important in its relation of the selected value for L. The more we increase r_c, the less impact spatial distance will have on the overall probabilities of the edges. If $r_c > L$, we end up in a situation where the distance does not matter at all. We vary r_c in the range 1.0-4.0 with varying tick sizes, as the closer we get to L, the smaller the change in ratio $\frac{r_c}{L}$ if we keep a static tick size.

In terms of clustering coefficient, the trend found is a decreasing clustering coefficient when r_c increases. The mechanism happening here might be that, since the distance plays a more important role with a decreasing r_c, connections made will be more local, increasing the probability of getting local cliques leading to a higher clustering coefficient. The notion that the connections are more local with low r_c is confirmed by the average and median edge distances, that show a clear upwards trend with increasing r_c. This result is in line with the expectations from making it a spatial network, as described in Section 3.5.

Looking at assortativity, we find a trend where increasing r_c makes the graph more disassortative (so a decrease in the correlation coefficient). This makes sense for similar reasons as the trend found with clustering. As r_c increases, the small nodes will have an easier time connecting to the high-degree nodes, thus making the graph more disassortative. The power-law exponent α shows a slight decrease when increasing r_c. This makes sense as a low r_c could make some nodes unreachable for small nodes, leading to a lower probability of high-degree nodes.

Some of the trends, especially clustering coefficient, seems to be more pronounced in the graphs with $m = 2$ compared to the graphs with $m = 1$. Additionally, we find a slight trend regarding path lengths in the graph with $m = 2$, whereas this trend does not seem present from the results from $m = 1$.

Impact of payoff structure Next we analyze the impact that the payoff structure has on the resulting topologies compared to a version of the model where the payoffs have been replaced by degrees. Note there is still some form of preference for larger banks as these have a higher degree, however there is no longer any form of intermediation present.

We pick the same combinations of settings as above on which we vary r_c with a tick size of 1. The results for $T = 80$ can be found in Table 4.3, the results for $T = 150$ in Table 4.4.

model	r_c	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	1.0	62.8	159	0.083	0.473	-0.075	2.70	4.76	3.76	3.76
main	2.0	62.8	160	0.084	0.380	-0.128	2.58	5.85	4.94	3.22
main	3.0	63.1	160	0.082	0.327	-0.151	2.54	6.69	5.93	2.96
main	4.0	63.0	160	0.083	0.316	-0.166	2.51	7.38	6.66	3.04
nopayoffs	1.0	63.4	160	0.082	0.470	-0.062	2.76	4.62	3.69	3.34
nopayoffs	2.0	63.0	160	0.083	0.403	-0.159	2.57	5.77	4.86	2.95
nopayoffs	3.0	63.1	160	0.083	0.358	-0.189	2.52	6.70	5.93	2.92
nopayoffs	4.0	62.9	160	0.083	0.341	-0.199	2.50	7.39	6.62	3.09
nopay,nodist	-	64.0	160	0.080	0.325	-0.218	2.49	-	-	2.91
nodistance	-	63.3	160	0.082	0.287	-0.161	2.52	-	-	2.84

Table 4.3: Results for alternative models related to payoffs as defined in section 4.5 with T=80,p=0.75 and m=2

model	r_c	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	1.0	61.8	147	0.080	0.357	-0.187	2.70	5.74	4.57	3.25
main	2.0	62.5	147	0.079	0.321	-0.247	2.64	6.70	5.76	3.18
main	3.0	60.9	146	0.082	0.330	-0.270	2.60	7.53	6.76	2.98
main	4.0	61.5	146	0.081	0.330	-0.281	2.57	8.04	7.41	2.85
nopayoffs	1.0	62.2	146	0.079	0.318	-0.107	2.82	5.54	4.44	3.44
nopayoffs	2.0	62.2	146	0.079	0.298	-0.230	2.67	6.62	5.74	2.81
nopayoffs	3.0	61.7	147	0.081	0.310	-0.282	2.60	7.50	6.80	2.98
nopayoffs	4.0	62.5	146	0.078	0.289	-0.283	2.63	8.00	7.35	2.75
nopay,nodist	-	61.4	146	0.081	0.306	-0.305	2.57	-	-	2.65
nodistance	-	61.7	146	0.080	0.323	-0.288	2.58	-	-	2.86

Table 4.4: Results for alternative models related to payoffs as defined in section 4.5 with T=150 p=0.4 and m=1

The results for $T = 150$ show a slight increase in the average clustering coefficient when using payoffs. Furthermore the payoffs limit the assortative trend that a decrease in r_c causes. An opposite trend is happening with $T = 80$, where the clustering coefficient in the main model is lower than the model without payoffs. Similar to the analysis of δ this indicates that the impact of the payoff structure is quite different for node birth ticks compared to the densification ticks.

To get an idea of a possible cause for these trends we consider the maximum probability that any node has in the ticks for choosing a node to expand from. For $T = 150$, once the graph has grown a bit the maximum probability settles around 0.15-0.17 for $\delta = 0.05$, whereas this probability settles around 0.07-0.08 for $\delta = 0.95$. This maximum probability even lowers to around 0.03 when considering the model without any payoffs present. Overall this means get a more even degree distribution and there is less of a difference between high degree nodes and low degree nodes when δ increases and intermediation becomes less relevant.

Impact of distance and degree cost To get a better understanding of the impact of the spatial component of the model and the added degree cost, we run the alternative models that remove these components as defined in Chapter 4. We again select the same combinations of settings, $T = 150, p = 0.4, m = 1$ and $T = 80, p = 0.75, m = 2$. Since we compare to models that have no distance cost, we do not vary r_C. We do vary δ, but for computational reasons we only take a few ticks in the overall range. The results for these settings can be found in Tables 4.5 and 4.6.

model	δ	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	0.05	62.9	160	0.083	0.373	-0.125	2.58	5.84	4.98	3.14
main	0.5	62.8	160	0.084	0.380	-0.127	2.58	5.85	4.94	3.22
main	0.95	62.9	160	0.083	0.370	-0.126	2.58	5.85	4.95	3.13
nocost	0.05	63.3	160	0.082	0.420	-0.124	2.64	5.16	4.49	3.23
nocost	0.5	63.6	160	0.081	0.415	-0.129	2.66	5.10	4.42	3.02
nocost	0.95	62.8	160	0.083	0.416	-0.121	2.64	5.11	4.39	3.22
nodistance	0.05	63.0	160	0.083	0.283	-0.161	2.52	-	-	2.76
nodistance	0.5	63.3	160	0.082	0.287	-0.161	2.52	-	-	2.84
nodistance	0.95	63.2	160	0.082	0.288	-0.161	2.52	-	-	2.87

Table 4.5: Results for alternative models related to distance as defined in section 4.5 with T=80 p=0.75 and m=2

model	δ	N	|E|	Dens.	CC	Assort.	APL	AvgΔ	MedΔ	Plaw α
main	0.05	61.3	146	0.081	0.340	-0.256	2.60	6.81	5.90	3.15
main	0.5	62.5	147	0.079	0.321	-0.247	2.64	6.70	5.76	3.18
main	0.95	61.4	146	0.081	0.321	-0.222	2.66	6.78	5.84	3.01
nocost	0.05	61.9	146	0.079	0.375	-0.245	2.66	5.74	5.05	3.02
nocost	0.5	62.3	146	0.079	0.375	-0.244	2.66	5.66	4.92	2.96
nocost	0.95	62.0	146	0.080	0.364	-0.225	2.70	5.70	5.03	3.15
nodistance	0.05	62.1	146	0.080	0.32	-0.286	2.59	-	-	2.88
nodistance	0.5	61.7	146	0.080	0.323	-0.288	2.58	-	-	2.86
nodistance	0.95	62.3	146	0.079	0.303	-0.258	2.63	-	-	3.19

Table 4.6: Results for alternative models related to distance as defined in section 4.5 with N=150 p=0.4 and m=1

Several interesting trends are happening in this table. Looking first at the impact of using the relative degree to scale the distance cost, a clear decrease in clustering coefficient is found. An explanation for this might come from the average and median spatial distances spent, as the version without degree scaling of the cost spends significantly lower distances. This results in more local connections, thus increasing the amount of triangles in the graph. This makes sense as in the main model the distance is less restrictive for high degree nodes than in the model without degree scaling. It will therefore overcome distances more easily, leading to higher average spatial distances in the main model. In terms of assortativity and power-law exponent there seems to be no real difference.

Considering the model without any distance component at all next, what stands out immediately is the strong decrease in clustering coefficient. Additionally, the resulting graphs are more disassortative and have lower power-law exponents. The distance component thus causes local banks to stay local and limits their ability to connect to high degree nodes. Since these local banks are limited, the maximum degree present in the graph will also be limited. This is reflected in the power-law exponent being much lower when this limit is not present.

These results combined with the previous analysis of r_c indicates that the spatial component of the model has a high impact on the resulting topology. The scaling of the cost according to the relative degree works as intended and succeeds in generating heterogeneous behaviour between small and big banks.

Core-periphery analysis To analyze whether there is a core-periphery structure as defined by Borgatti & Everett [9] we run the Lip algorithm [10]. Similar to the analyses above, we run it for different values of δ and r_c. Additionally, we run the Lip algorithm for the alternative models as defined previously. We evaluate the differences that occur between the different versions of the model using the core density and size. The resulting values can be found in Tables 4.7 and 4.8.

model	r_c	δ	Core size	Core density	Periphery density
main	1.0	0.5	9.7	0.588	0.039
main	2.0	0.5	9.7	0.614	0.036
main	3.0	0.5	9.7	0.641	0.033
main	4.0	0.5	9.6	0.633	0.032
main	2.0	0.05	9.7	0.622	0.035
main	2.0	0.95	9.7	0.624	0.034
nopayoffs	2.0	0.5	9.8	0.616	0.033
nopay,nodist	2.0	0.5	9.5	0.640	0.029
nodistance	2.0	0.5	9.6	0.645	0.032
nocost	2.0	0.5	9.7	0.579	0.036

Table 4.7: Results for core-periphery structure using Lip algorithm [10] for networks with T=80, p=0.75, m=2

model	r_c	δ	Core size	Core density	Periphery density
main	1.0	0.5	9.8	0.698	0.024
main	2.0	0.5	9.9	0.734	0.020
main	3.0	0.5	9.8	0.770	0.020
main	4.0	0.5	9.7	0.766	0.019
main	2.0	0.05	9.8	0.747	0.020
main	2.0	0.95	10.0	0.756	0.021
nopayoffs	2.0	0.5	10.3	0.778	0.019
nopay,nodist	2.0	0.5	10.2	0.833	0.015
nodistance	2.0	0.5	9.8	0.780	0.018
nocost	2.0	0.5	9.6	0.698	0.022

Table 4.8: Results for core-periphery structure using Lip algorithm [10] for networks with T=150, p=0.4, m=1.

For every network that was created and evaluated, the Lip algorithm finds a statistically significant core-periphery partitioning when using the Erdos-Renyi model as null model. As this measure is mostly focused towards maximizing the link density between core members, we compare the differences in core density as a result of varying parameters.

We find that the distance has a significant impact on the density of the core, where an increase of impact of the distance (thus decrease in r_c) leads to less dense cores in both the networks with $T = 150$ and $T = 80$. This could be a result of it becoming less attractive to span a large distance to a high degree node, which results in the nodes initiating edges opting for periphery nodes nearby instead. The core size does not seem to change with distance.

Interestingly, the payoff structure reduces the core density and slightly reduces the core size for the networks with $T = 150$.

Another trend that stands out when comparing the networks $T = 150$ and $T = 80$ is a strong difference in core density when p is lower. This most likely comes from the large nodes getting to initiate more links in contrast to all nodes getting to initiate links at birth. As a result of their size, the large nodes link more to the other large nodes resulting in a higher core density.

4.2.2 Japanese Network

T	p	m	N	|E|	Dens.	CC	Assort.	APL	AvgΔ	MedΔ	Plaw α
2000	0.175	1	352.2	1975	0.032	0.433	-0.255	2.60	6.26	5.33	2.55
1750	0.20	1	354.3	1730	0.028	0.397	-0.244	2.69	6.14	5.23	2.61
1500	0.23	1	346.7	1485	0.025	0.377	-0.238	2.74	6.05	5.09	2.67
1250	0.28	1	352.8	1240	0.020	0.326	-0.223	2.86	5.85	4.89	2.65
1000	0.35	1	352.7	995	0.016	0.263	-0.207	3.02	5.68	4.72	2.68
1000	0.35	2	352.9	1981	0.032	0.441	-0.204	2.54	5.74	4.83	2.62
900	0.39	2	350.9	1784	0.029	0.413	-0.192	2.61	5.68	4.72	2.58
800	0.44	2	357.2	1590	0.025	0.378	-0.175	2.69	5.48	4.57	2.63
700	0.50	2	354.3	1393	0.022	0.343	-0.161	2.79	5.35	4.42	2.62
700	0.50	3	354.3	2083	0.033	0.375	-0.157	2.53	5.50	4.55	2.59
600	0.58	2	350.0	1196	0.020	0.300	-0.135	2.90	5.12	4.21	2.66
600	0.58	3	352.1	1791	0.029	0.327	-0.130	2.64	5.25	4.32	2.52
550	0.64	3	357.9	1646	0.026	0.291	-0.109	2.72	5.03	4.15	2.59
500	0.7	3	352.2	1498	0.024	0.267	-0.095	2.78	4.91	4.04	2.65
500	0.7	4	354.3	1995	0.032	0.288	-0.088	2.61	5.01	4.15	2.55
450	0.77	4	350.9	1799	0.029	0.253	-0.069	2.68	4.82	3.99	2.66
400	0.875	4	353.9	1602	0.026	0.201	-0.038	2.81	4.54	3.80	2.85
400	0.875	5	356.9	2002	0.032	0.215	-0.033	2.67	4.61	3.87	2.81

Table 4.9: Results of varying T, p and m. $L = 10, \delta = 0.5, r_c = 2.0$

Impact of density parameters Looking at the table for the medium-sized networks (Table 4.9), we again find several interesting trends. First off, despite having a much lower density compared to the smaller Danish networks, the clustering coefficients still are averaging around 0.3-0.4. Also the drop-off in clustering coefficient with decreasing density is less pronounced. What stands out is that generally at its highest with $m = 2$. A possible explanation for this could be that there will be many nodes with degree 2, where only one additional edge in the network has to happen to close the triangle and get the node to a clustering coefficient of 1. This could lead to having a large portion of the nodes with a clustering coefficient of 1, thus further pushing up the average. With $m \geq 3$, there will be no nodes with degree 2, so it will be harder to randomly get to nodes with a clustering coefficient of 1. We see this trend continue when further increasing m to 4 and 5, where the clustering coefficients keep dropping lower.

In terms of assortativity we find a similar trend to the smaller networks. As p and m go up, so does the assortativity coefficient, in this case to the point of almost being an assortative network when $p = 0.875$. These assortativity values are all in line with what is to be expected of an interbank network when looking at the empirical networks. A possible explanation for this trend is that the destinations of most edges are decided on at the birth of one node, where at that point the nodes always have a low degree and will never span large distances to connect to bigger, further away nodes. Instead, these nodes are more likely to connect to the local, smaller nodes, thus bringing up the assortativity.

The possible explanation regarding the assortativity coefficient is further supported by the results of the average spatial distance spent by the nodes. The degrees of most nodes are low when they are deciding where to connect to, so the spatial distance will not be overcome easily.

Looking at the power-law fits again, the average values of α are between 2.52 and 2.85, which is a lot closer to the values found in empirical networks (2.0-2.5) than the smaller Danish networks got, but is still a bit on the high end of the range.

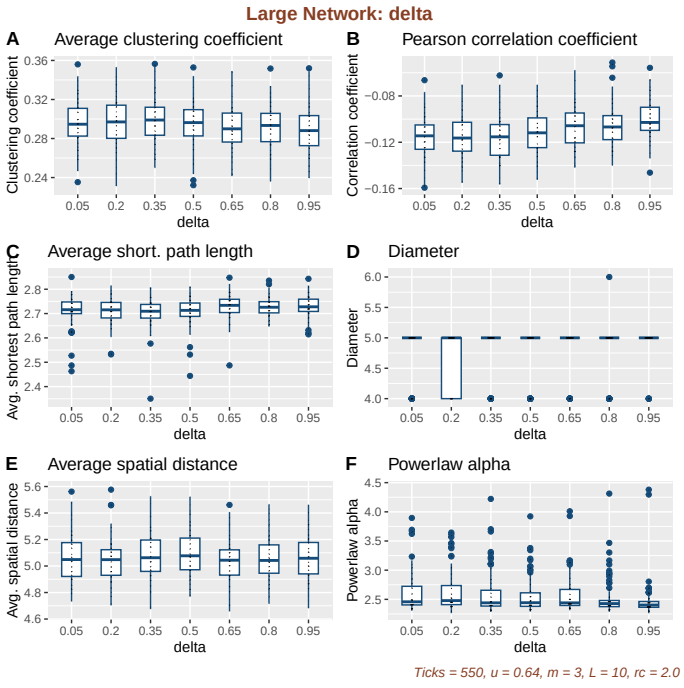

Figure 4.5: Graph metrics when varying δ in the networks with T=550, p=0.64, m=3

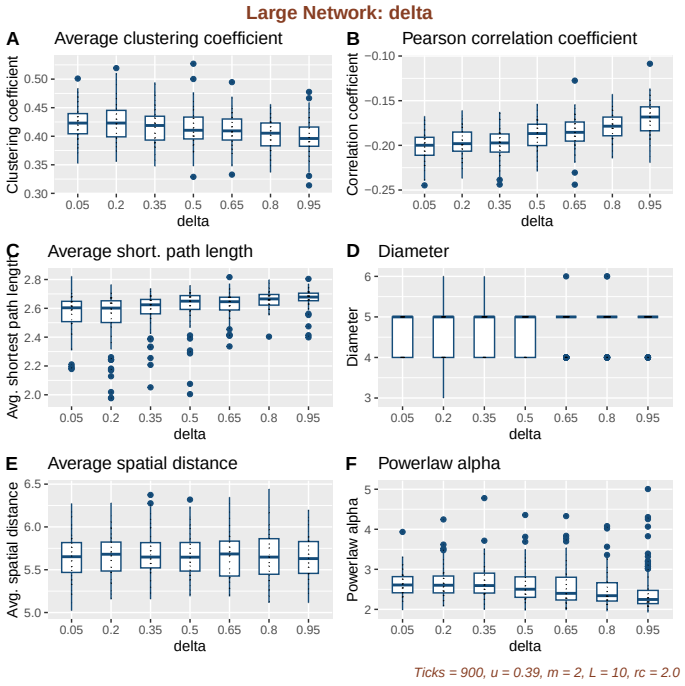

Figure 4.6: Graph metrics when varying δ in the networks with T=900, p=0.39, m=2

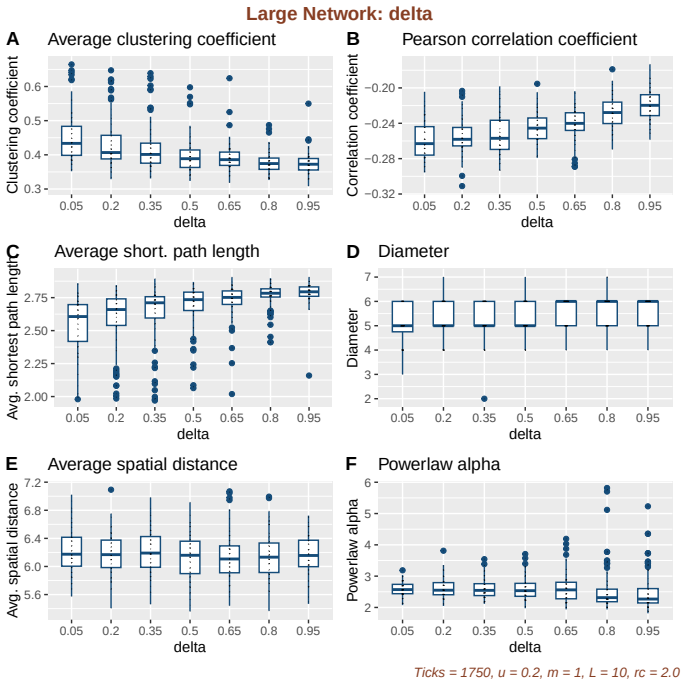

Figure 4.7: Graph metrics when varying δ in the networks with T=1750, p=0.2, m=1

Impact of δ In contrast to the results from the small network, we do find interesting trends in this medium sized network when varying δ. Again we picked several network settings from the table to use to specifically analyze the parameters. The selected combinations of settings are $T = 1750, m = 1, p = 0.2$, $T = 900, m = 2, p = 0.39$, $T = 550, m = 3, p = 0.64$. We again vary δ between 0.05 and 0.95 with a tick size of 0.15.

We find a trend where the average clustering coefficient decreases when we increase δ. The trend is quite subtle, as in the graphs with $T = 1750$ we find an average of 0.43 with $\delta = 0.05$ compared to an average of 0.37 with $\delta = 0.95$. Interestingly, the trend is even more subtle when we

look at the graphs with $T = 900$ where the average range of clustering coefficients is between 0.39 and 0.42 and the trend even seems to disappear completely in the graphs with $T = 550$. The reason of the trend only happening on graphs with a higher number of ticks combined with a lower value of p might be related again to the difference in type of ticks happening. When p is low, we have a lot of ticks where a node is picked by its and initiates edges as opposed to having a lot of ticks where most edges are created as nodes are born. When nodes are born, they have no connections yet so they have no intermediation power, resulting in δ having no impact on the intermediation component of the payoff. In the reach component a difference in δ will equally impact all nodes that provide any reach to begin with, so when proportionally computing the probabilities they will be fairly similar for any δ.

In terms of assortativity the medium sized graphs also show a trend when changing δ, where an increasing delta produces a graph with a higher correlation coefficient, thus a less disassortative graph. Again, the trend is more pronounced in the graphs with lower values for m. Additionally, we find a slight trend of increasing average shortest path lengths with increasing delta. With $m = 1$ we also find a higher average diameter of the graph in the higher range of δ. For the degree distribution we find an interesting trend where an increase in δ causes a lower value for the power-law exponent α.

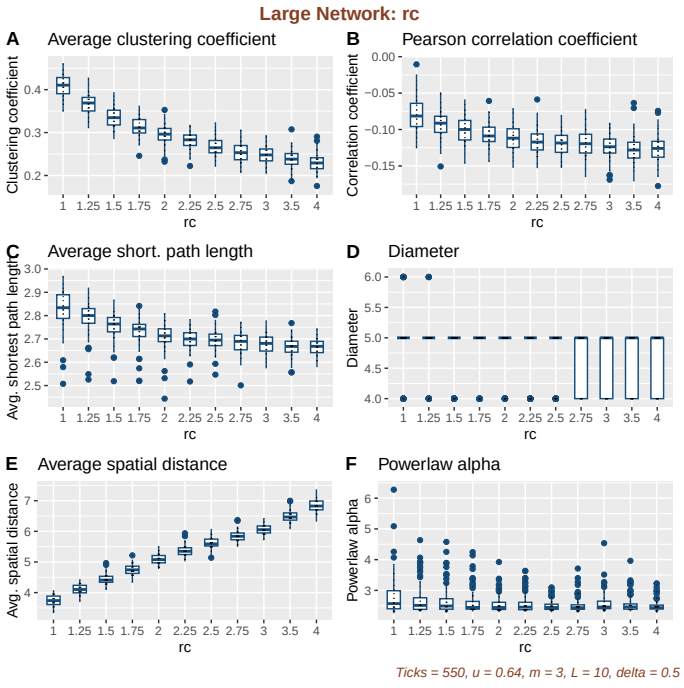

Figure 4.8: Graph metrics when varying r_c in the networks with T=550, p=0.64, m=3

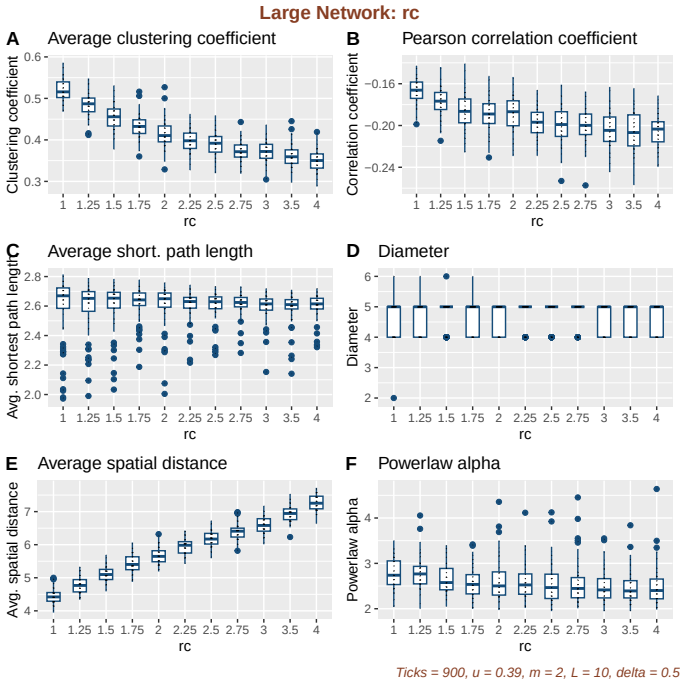

Figure 4.9: Graph metrics when varying r_c in the networks with T=900, p=0.39, m=2

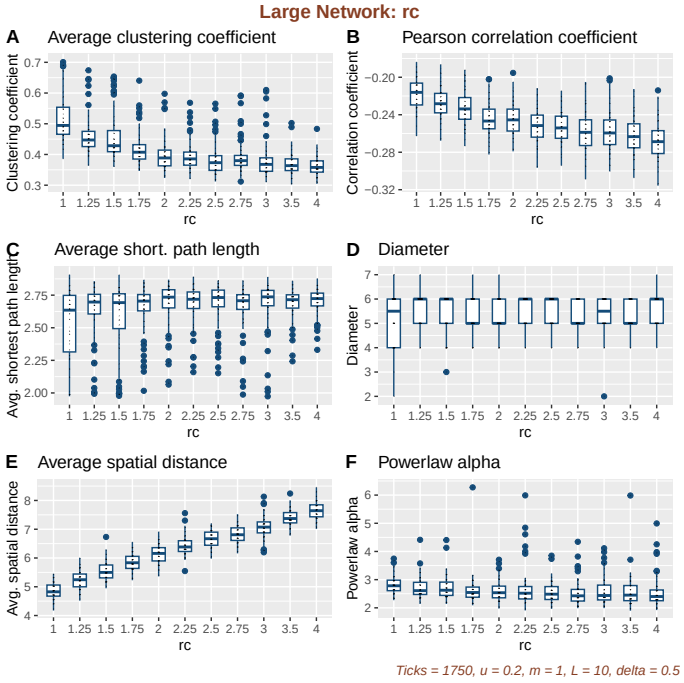

Figure 4.10: Graph metrics when varying r_c in the networks with T=1750, p=0.2, m=1

Impact of r_c Once again, with the same three picked combinations of settings we vary r_c from 1 to 4 with the same tick sizes as the analysis of the small network. The results can be found in Figures 4.8, 4.9 and 4.10

The impact of changing r_c in the medium-sized networks is similar to what we found in the small networks previously. When increasing r_c, we again find a decrease in the average clustering coefficient, a decrease in the correlation coefficient and a decrease in the power-law exponent. The trends are in line with the expectations we got from the base spatial model [6].

In terms of path lengths we find the impact of r_c to be more pronounced in the networks with $m = 3$ then in the networks with $m = 2$ and $m = 1$, where almost no trend seems to

be present. Higher values of r_c lead to shorter path lengths, which makes sense as an increased distance component might prevent more efficient routes to be formed.

Impact of payoff structure We consider the same three picked combinations to specifically analyze the impact of the payoff component. We again vary r_c with a tick size of 1. Results are shown in Tables 4.10, 4.11 and 4.12

model	r_c	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	1.0	358	1646	0.026	0.408	-0.079	2.83	3.74	2.79	2.80
main	2.0	355	1646	0.026	0.294	-0.112	2.71	5.09	4.21	2.55
main	3.0	356	1645	0.026	0.248	-0.123	2.68	6.06	5.25	2.58
main	4.0	356	1644	0.026	0.230	-0.127	2.67	6.83	6.09	2.49
nopayoffs	1.0	357	1645	0.026	0.402	-0.017	2.95	3.52	2.72	2.57
nopayoffs	2.0	356	1646	0.026	0.315	-0.095	2.75	4.91	4.09	2.43
nopayoffs	3.0	357	1646	0.026	0.278	-0.130	2.68	5.99	5.19	2.44
nopayoffs	4.0	355	1645	0.026	0.265	-0.142	2.65	6.77	6.04	2.40
nopay,nodist	-	357	1645	0.026	0.234	-0.143	2.64	-	-	2.44
nodistance	-	355	1645	0.026	0.202	-0.126	2.66	-	-	2.50

Table 4.10: Results for alternative models related to payoffs as defined in section 4.5 with T=550 p=0.64 and m=3

model	r_c	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	1.0	351	1784	0.029	0.520	-0.167	2.60	4.43	3.3	2.77
main	2.0	356	1785	0.028	0.414	-0.189	2.62	5.67	4.72	2.61
main	3.0	352	1785	0.029	0.372	-0.204	2.60	6.59	5.78	2.48
main	4.0	358	1786	0.028	0.349	-0.206	2.61	7.27	6.55	2.48
nopayoffs	1.0	353	1786	0.029	0.522	-0.074	2.84	4.26	3.29	2.45
nopayoffs	2.0	356	1786	0.028	0.459	-0.171	2.69	5.71	4.79	2.13
nopayoffs	3.0	352	1785	0.029	0.438	-0.207	2.63	6.68	5.88	2.07
nopayoffs	4.0	355	1785	0.029	0.430	-0.229	2.62	7.34	6.65	2.08
nopay,nodist	-	353	1785	0.029	0.411	-0.236	2.61	-	-	2.07
nodistance	-	352	1783	0.029	0.327	-0.208	2.59	-	-	2.42

Table 4.11: Results for alternative models related to payoffs as defined in section 4.5 with T=900 p=0.39 and m=2

model	r_c	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	1.0	352	1728	0.028	0.516	-0.218	2.52	4.85	3.67	2.80
main	2.0	354	1729	0.028	0.397	-0.244	2.69	6.14	5.23	2.61
main	3.0	351	1729	0.028	0.377	-0.258	2.69	7.08	6.30	2.58
main	4.0	353	1730	0.028	0.361	-0.268	2.71	7.66	7.01	2.53
nopayoffs	1.0	352	1732	0.028	0.396	-0.120	2.93	4.96	3.96	2.79
nopayoffs	2.0	351	1731	0.028	0.370	-0.237	2.8	6.55	5.70	2.13
nopayoffs	3.0	350	1733	0.029	0.364	-0.274	2.77	7.45	6.74	2.04
nopayoffs	4.0	350	1733	0.029	0.362	-0.291	2.76	7.96	7.35	1.96
nopay,nodist	-	354	1731	0.028	0.357	-0.305	2.76	-	-	1.86
nodistance	-	348	1730	0.029	0.357	-0.270	2.69	-	-	2.40

Table 4.12: Results for alternative models related to payoffs as defined in section 4.5 with T=1750 p=0.2 and m=1

The tables reveal some interesting trends and information. First off, the payoffs make the graphs more disassortative for the graphs with $m = 1$, but more assortative for the graphs with $m = 3$ (except when $r_c = 1$). For the average clustering coefficient, we find the payoff structure

gives a slight increase at $m = 1$. For $m = 2$ and $m = 3$ however, the payoffs drastically reduce the clustering coefficient. The payoff structure shortens the average path length in the graphs with $m = 1$ and $m = 2$.

In terms of power-law exponent we find the versions with payoffs to exhibit higher values. Since this trend is again more visible in the graphs with lower values for p, this is possibly a result of the node picking process in the densification ticks.

Impact of spatial and degree cost We consider the same three picked combinations to analyze the impact of the spatial components, again with varying δ. Results are shown in Tables 4.13, 4.14 and 4.15.

The patterns we found in the results for the Danish network also emerge in the results for this Japanese network. Again the clustering coefficient is quite a bit higher when we remove the scaling of the distance by relative degree. Additionally, we find slightly more disassortative results, something that was not found in the smaller network above. The decrease in restrictiveness of the distance in the main model is also reflected in the power-law exponent, which is slightly lower. This indicates that high degree nodes are more likely.

Once again, when all distance restrictions are removed, the clustering coefficient goes down greatly. The power-law exponent is clearly impacted as well. Interestingly, the clustering coefficient stays above 0.3 in the results for $T = 900$ and $T = 1750$. This indicates that, when the number of ticks is high and p is lower, the height of the clustering coefficient is less explained by the distance and other factors contribute to a high clustering coefficient as well.

model	δ	**N|**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	0.05	357	1646	0.026	0.296	-0.116	2.71	5.06	4.19	2.61
main	0.5	355	1646	0.026	0.294	-0.112	2.71	5.09	4.21	2.55
main	0.95	355	1645	0.026	0.289	-0.099	2.73	5.06	4.18	2.47
nocost	0.05	355	1645	0.026	0.329	-0.124	2.74	4.47	3.86	2.72
nocost	0.5	357	1644	0.026	0.325	-0.116	2.75	4.44	3.83	2.63
nocost	0.95	359	1646	0.026	0.317	-0.101	2.80	4.41	3.79	2.55
nodistance	0.05	357	1644	0.026	0.204	-0.129	2.65	-	-	2.54
nodistance	0.5	355	1645	0.026	0.202	-0.126	2.66	-	-	2.50
nodistance	0.95	358	1645	0.026	0.193	-0.112	2.67	-	-	2.46

Table 4.13: Results for alternative models related to distance as defined in section 4.5 with T=550 p=0.64 and m=3

model	δ	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	0.05	354	1785	0.029	0.424	-0.201	2.57	5.65	4.72	2.64
main	0.5	356	1785	0.028	0.414	-0.189	2.62	5.67	4.72	2.61
main	0.95	355	1786	0.029	0.399	-0.169	2.67	5.64	4.74	2.45
nocost	0.05	355	1784	0.029	0.460	-0.208	2.57	4.92	4.26	2.63
nocost	0.5	353	1784	0.029	0.446	-0.192	2.60	4.87	4.22	2.72
nocost	0.95	351	1785	0.029	0.438	-0.175	2.70	4.77	4.15	2.52
nodistance	0.05	355	1782	0.029	0.34	-0.218	2.57	-	-	2.51
nodistance	0.5	352	1783	0.029	0.327	-0.208	2.59	-	-	2.42
nodistance	0.95	354	1786	0.029	0.317	-0.191	2.63	-	-	2.23

Table 4.14: Results for alternative models related to distance as defined in section 4.5 with T=900 p=0.39 and m=2

model	δ	**N**	**|E|**	**Dens.**	**CC**	**Assort.**	**APL**	**AvgΔ**	**MedΔ**	**Plaw α**
main	0.05	353	1729	0.028	0.452	-0.259	2.55	6.22	5.24	2.59
main	0.5	354	1729	0.028	0.397	-0.244	2.69	6.14	5.23	2.61
main	0.95	352	1730	0.028	0.374	-0.219	2.79	6.16	5.27	2.49
nocost	0.05	353	1726	0.028	0.505	-0.258	2.50	5.33	4.62	2.62
nocost	0.5	352	1731	0.028	0.472	-0.248	2.60	5.21	4.54	2.59
nocost	0.95	350	1732	0.028	0.427	-0.22	2.79	4.98	4.38	2.64
nodistance	0.05	349	1729	0.029	0.380	-0.283	2.65	-	-	2.46
nodistance	0.5	348	1730	0.029	0.357	-0.270	2.69	-	-	2.40
nodistance	0.95	353	1733	0.028	0.339	-0.250	2.78	-	-	2.26

Table 4.15: Results for alternative models related to distance as defined in section 4.5 with T=1750 p=0.2 and m=1

Core-periphery analysis Similar to the core-periphery analysis above, we use the Lip algorithm to partition every network into a core and a periphery and compare the networks using the found core density and size. Results can be found in Tables 4.16, 4.17 and 4.18

model	r_c	δ	**Core size**	**Core density**	**Periphery density**
main	1.0	0.5	25.2	0.486	0.013
main	2.0	0.5	25.6	0.542	0.012
main	3.0	0.5	25.6	0.555	0.012
main	4.0	0.5	25.6	0.571	0.011
main	2.0	0.05	25.3	0.535	0.012
main	2.0	0.95	25.9	0.551	0.012
nopayoffs	2.0	0.5	26.7	0.555	0.012
nopay,nodist	2.0	0.5	26.1	0.616	0.011
nodistance	2.0	0.5	25.6	0.576	0.012
nocost	2.0	0.5	25.1	0.505	0.013

Table 4.16: Results for core-periphery structure using Lip algorithm [10] for networks with T=550, p=0.64, m=3.

model	r_c	δ	**Core size**	**Core density**	**Periphery density**
main	1.0	0.5	26.7	0.560	0.012
main	2.0	0.5	27.8	0.619	0.010
main	3.0	0.5	28.0	0.645	0.010
main	4.0	0.5	27.9	0.652	0.010
main	2.0	0.05	27.2	0.607	0.010
main	2.0	0.95	28.9	0.636	0.010
nopayoffs	2.0	0.5	30.9	0.687	0.009
nopay,nodist	2.0	0.5	30.7	0.769	0.007
nodistance	2.0	0.5	28.1	0.666	0.010
nocost	2.0	0.5	27.2	0.578	0.011

Table 4.17: Results for core-periphery structure using Lip algorithm [10] for networks with T=900, p=0.39, m=2.

model	r_c	δ	Core size	Core density	Periphery density
main	1.0	0.5	26.6	0.602	0.010
main	2.0	0.5	28.5	0.674	0.008
main	3.0	0.5	28.7	0.708	0.008
main	4.0	0.5	28.7	0.717	0.007
main	2.0	0.05	26.7	0.669	0.008
main	2.0	0.95	30.0	0.692	0.008
nopayoffs	2.0	0.5	33.5	0.786	0.006
nopay,nodist	2.0	0.5	33.5	0.858	0.004
nodistance	2.0	0.5	29.1	0.743	0.007
nocost	2.0	0.5	27.6	0.633	0.009

Table 4.18: Results for core-periphery structure using Lip algorithm [10] for networks with T=1750, p=0.2, m=1.

Once again we find increased core densities as p decreases, indicating that the densification ticks mostly take place within the core. In the results for these larger graphs, we also find that the core size gets increasingly bigger when we lower p.

Interestingly, the version without payoffs and without distance actually exhibits the densest core, and also the largest cores in the networks with $m = 1$.

4.3 Discussion

Density parameters As has become clear from analyzing both the small and large networks, the choice for a low p with a low m produces very different resulting graphs compared to higher values of p and m. The strongest differences are found in the clustering coefficient and assortativity values. Additionally, the connections stay more local and are more based on spatial distances with high p and m. We find the networks with $m = 2$ to provide the best balance as an overall network, as it generally has high clustering coefficients and is disassortative with correlation values around -0.2 and -0.15 in the large network and around -0.17 and -0.12 in the small network. These fall well into the range of -0.3 and -0.15 that we found for empirical interbank networks. The clustering coefficients found for these networks are generally between 0.35 and 0.45, which is slightly below what is found in empirical interbank networks.Through use of r_c and δ the model is able to produce higher clustering coefficients if needed.

Distance component The distance component can have a strong impact on the resulting networks, which can accurately be controlled by r_c. The differences in resulting topology are strongly present in all networks with different m. The range of r_c in which the differences are strongest is between 1 and 3, for graphs with size $L = 10$. When going lower for r_c the model will end up being mostly spatial and exhibit a high clustering coefficient and be more assortative.When going higher for r_c we will end up with a mostly preferential model, with lower power law coefficients and lower clustering coefficients.

The degree cost component slightly softens the impact of the spatial component, especially for large nodes. This is reflected in the lower clustering coefficient and power-law alpha. In future work, this component could be adapted to scale the degree cost by a different function than a linear function. It would be interesting to see if changing this function could give another tool to subtly impact the topology.

Payoff component The payoff component has a strong impact on the degree distribution, as without it the power law exponent will drop below 2. This effect is especially clear in the networks with $r_c > 2.0$. In the networks with $r_c = 1$ we see the payoffs put a limit the assortative effect that the distance produces. Interestingly, the clustering coefficient is lower for the networks with payoffs.

The impact of δ on the network is a bit harder to characterize intuitively, as it is of opposite, but not equal, influence in both the Intermediation function and the Reach function. We mostly see the impact of δ with low values of p, so when there are a lot of densification ticks. To recap, when delta is low, intermediation benefits are high and we find higher clustering and less assortative graphs.

Also the path lengths are a bit lower. When delta is high, there are almost no intermediation benefits and we see lower power law exponents, indicating a more even degree distribution. We suspect a big part of this result comes from big nodes gaining more intermediation benefits than it loses on reach benefits when δ goes down, while the smaller nodes end up with a smaller piece of the trading benefits when δ is low. Since the smaller nodes do not have any intermediation benefits to begin with, the difference between small and big nodes is a lot higher, resulting in the big nodes being picked more often in the densification ticks, and thus expanding their connectivity.

Core periphery Using the Lip algorithm [10], we find significant core periphery structure as defined in Borgatti et al. [9] in all resulting networks. When we look at the core densities we find the densest for graphs with $m = 1$. We find that increasing the impact of middlemen by decreasing δ decreases the density and size of the core. Additionally, we find that increasing the impact of the distance component also has a negative impact on the core density. Since we added the distance component to increase presence of hubs and limit the ability of periphery nodes to form links, we would have expected to find a clearer core with increased density within the core. The result, however, is opposite as it turns out the distance cost also negatively impacts the probability of links that hubs could form to each other. It is also a possibility that the core periphery measures we chose do not accurately reflect the intuition of hubs that we tried to bring in through distances.

We used the payoff component to get the network to exhibit a core-periphery structure, yet we find the cores and core densities to be higher in the models without payoff. This indicates that the core-periphery structure that we find is more likely to be a result of other model components. Since we find the highest core density in the model with $m = 1$, no payoffs and no distance components, we believe the presence of many densification ticks (low p) might contribute the most to the emergence of the core-periphery structure that we find using the Lip algorithm [10].

Besides the Lip algorithm we have tried some other core periphery algorithms. Using the Surprise algorithm [11] we also find significant core periphery structures. However, due to the running times of this algorithm and the time limited nature of this project we have not been able to fully investigate this result. The KM-Config algorithm [12] we find some significant core-periphery pairs for some networks. The pairs selected, however, do not always make sense as it sometimes includes nodes as core that have a very low degree and are clearly not well connected to any of the bigger nodes. This is a consequence of this algorithm claiming to find a core that is significant beyond the degree of nodes as explanation. As these notions are part of ongoing discussion on what core-periphery structure entails exactly and how the significance should be measured, this is decided to be beyond the scope of the project.

5 Future work

Network model So far this model produces undirected and unweighted edges. Empirical data showed that the edges can be weighted in terms of both volume and value. A next version of this model could be made to be directed and include weights. These can also be used to further adapt payoff functions in order to more accurately model the way edges are chosen.

The model is composed of many different components, some of which have alternatives that could result in somewhat different topologies. First off, we currently use an exponential function for the distance cost in a uniformly distributed space. Both these distributions can be changed to obtain new ways of computing edge cost. In addition, the degree cost is a linear function based on the relative degree of the node. One could set up this function with another distribution, or based on something other than the degree, to obtain other ways of adding cost.

Next, the current model sums the Direct, Reach and Intermediation components to obtain overall payoff. This could also be balanced in different ways by adding weight parameters to each of these separate components.

The impact of what nodes are selected to expand in the densification step is also not to be underestimated. We currently choose based on probabilities proportional to the overall payoff of the node. This could also be adjusted to be selected proportional to different distributions or based on other measures. Additionally, the model could be made more flexible in the number of edges that is produced per step. For a large network, a static m might not be much of problem. For smaller networks, however, it could be beneficial to have a dynamic m, in which for example some steps add one new edge and some steps add multiple.

Attribute data Our model serves as a first step towards realistic financial synthetic data. The next step is to generate transaction data and attribute data on top of the resulting topologies. First directions for this could come from a method such as AMLSim [39], which takes a full degree sequence as network and generates transactions on top of this.

Implementation Next, the current implementation of the model was made as a proof of concept, and can easily be made more efficient by reducing the amount of double computations that are currently being done. The current implementation recomputes all edge weights at every consideration point. A new implementation can be made which makes more efficient use of memory and only recomputes the affected edge weights when a new edge is added. This improvement could get us to generation of larger networks, which would be needed to reproduce networks such as the Fedwire system that contains over 7000 nodes [13]. It is also interesting to see whether the network properties produced in the smaller networks will hold up when scaling up the networks to larger sizes.

6 Conclusion

In this work, we have proposed a model that attempts to reproduce interbank networks. To achieve this we analyzed the existing literature on empirical interbank networks and network generation models. Our resulting model uses an adaptation of fitness-based preferential attachment based on a payoff structure for interbank networks [7]. We combined this with a spatial component to add a dynamic cost component and a densification mechanism from the Nearest Neighbor model [5] to ensure the growth of big banks.

We tested our model by fitting to some of the empirical measures found in earlier reports on interbank networks. For key metrics such as clustering, assortativity, path lengths and degree distribution we found the model to be able to reproduce the topology of the general interbank network as found from reports on empirical networks well. Additionally, we find the desired core periphery structure in all resulting networks with high density cores. It is doubtful whether the found core periphery structure is a result of using the payoff structure that is supposed to lead to a core-periphery structure rather than a result of the densification steps present in the model.

The model provides high flexibility in terms of link costs through the distance and degree cost mechanisms. The payoff structure gives a more subtle control over the network through use of δ. Therefore we believe this model can be the basis for further use in synthetic financial data generation.

Bibliography

[1] X. Wu, X. Ying, K. Liu, and L. Chen, "A survey of privacy-preservation of graphs and social networks," in *Managing and mining graph data*, pp. 421–453, Springer, 2010.

[2] R. Longadge and S. Dongre, "Class imbalance problem in data mining review," *arXiv preprint arXiv:1305.1707*, 2013.

[3] S.-H. Lim, S. Lee, S. S. Powers, M. Shankar, and N. Imam, "Survey of approaches to generate realistic synthetic graphs,"

[4] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of modern physics*, vol. 74, no. 1, p. 47, 2002.

[5] A. Vázquez, "Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations," *Physical Review E*, vol. 67, no. 5, p. 056104, 2003.

[6] M. Barthélemy, "Crossover from scale-free to spatial networks," *EPL (Europhysics Letters)*, vol. 63, no. 6, p. 915, 2003.

[7] D. In't Veld, M. Van der Leij, and C. Hommes, "The formation of a core-periphery structure in heterogeneous financial networks," *Journal of Economic Dynamics and Control*, vol. 119, p. 103972, 2020.

[8] A. D. Broido and A. Clauset, "Scale-free networks are rare," *Nature communications*, vol. 10, no. 1, pp. 1–10, 2019.

[9] S. P. Borgatti and M. G. Everett, "Models of core/periphery structures," *Social networks*, vol. 21, no. 4, pp. 375–395, 2000.

[10] S. Z. Lip, "A fast algorithm for the discrete core/periphery bipartitioning problem," *arXiv preprint arXiv:1102.5511*, 2011.

[11] J. v. L. de Jeude, G. Caldarelli, and T. Squartini, "Detecting core-periphery structures by surprise," *EPL (Europhysics Letters)*, vol. 125, no. 6, p. 68001, 2019.

[12] S. Kojaku and N. Masuda, "Core-periphery structure requires something else in the network," *New Journal of physics*, vol. 20, no. 4, p. 043012, 2018.

[13] K. Soramäki, M. L. Bech, J. Arnold, R. J. Glass, and W. E. Beyeler, "The topology of interbank payment flows," *Physica A: Statistical Mechanics and its Applications*, vol. 379, no. 1, pp. 317–333, 2007.

[14] K. B. Rørdam, M. L. Bech, *et al.*, "The topology of danish interbank money flows," *Banks and Bank Systems*, vol. 4, no. 4, pp. 48–65, 2009.

[15] F. Kyriakopoulos, S. Thurner, C. Puhr, and S. W. Schmitz, "Network and eigenvalue analysis of financial transaction networks," *The European Physical Journal B*, vol. 71, no. 4, pp. 523–531, 2009.

[16] K. Imakubo, Y. Soejima, *et al.*, "The transaction network in japan's interbank money markets," *Monetary and Economic Studies*, vol. 28, pp. 107–150, 2010.

[17] L. Embree and T. Roberts, "Network analysis and canada's large value transfer system," tech. rep., Bank of Canada Discussion Paper, 2009.

[18] S. Martinez-Jaramillo, B. Alexandrova-Kabadjova, B. Bravo-Benitez, and J. P. Solórzano-Margain, "An empirical study of the mexican banking system's network and its implications for systemic risk," *Journal of Economic Dynamics and Control*, vol. 40, pp. 242–265, 2014.

[19] F. D. Forte, "Network topology of the argentine interbank money market," *Journal of Complex Networks*, vol. 8, no. 4, p. cnaa039, 2020.

[20] S. N. Dorogovtsev and J. F. Mendes, "Evolution of networks," *Advances in physics*, vol. 51, no. 4, pp. 1079–1187, 2002.

[21] I. Van Lelyveld *et al.*, "Finding the core: Network structure in interbank markets," *Journal of Banking & Finance*, vol. 49, pp. 27–40, 2014.

[22] M. Drobyshevskiy and D. Turdakov, "Random graph modeling: A survey of the concepts," *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, pp. 1–36, 2019.

[23] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat, "Systematic topology analysis and generation using degree correlations," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4, pp. 135–146, 2006.

[24] B. Tillman, *Graph Construction Using the dK-Series Framework*. University of California, Irvine, 2019.

[25] M. Gjoka, M. Kurant, and A. Markopoulou, *2.5 k-graphs: from sampling to generation*. IEEE, 2013.

[26] C. Orsini, M. M. Dankulov, P. Colomer-de Simón, A. Jamakovic, P. Mahadevan, A. Vahdat, K. E. Bassler, Z. Toroczkai, M. Boguná, G. Caldarelli, *et al.*, "Quantifying randomness in real networks," *Nature communications*, vol. 6, no. 1, pp. 1–10, 2015.

[27] D. Chakrabarti, Y. Zhan, and C. Faloutsos, "R-mat: A recursive model for graph mining," in *Proceedings of the 2004 SIAM International Conference on Data Mining*, pp. 442–446, SIAM, 2004.

[28] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and Z. Ghahramani, "Kronecker graphs: an approach to modeling networks.," *Journal of Machine Learning Research*, vol. 11, no. 2, 2010.

[29] S. Moreno, S. Kirshner, J. Neville, and S. Vishwanathan, "Tied kronecker product graph models to capture variance in network populations," in *2010 48th annual Allerton conference on communication, control, and computing (Allerton)*, pp. 1137–1144, IEEE, 2010.

[30] P. Holme and B. J. Kim, "Growing scale-free networks with tunable clustering," *Physical review E*, vol. 65, no. 2, p. 026107, 2002.

[31] D. J. Soares, C. Tsallis, A. M. Mariz, and L. R. da Silva, "Preferential attachment growth model and nonextensive statistical mechanics," *EPL (Europhysics Letters)*, vol. 70, no. 1, p. 70, 2005.

[32] P. L. Krapivsky and S. Redner, "Network growth by copying," *Phys. Rev. E*, vol. 71, p. 036118, Mar 2005.

[33] G. Bianconi and A.-L. Barabási, "Competition and multiscaling in evolving networks," *EPL (Europhysics Letters)*, vol. 54, no. 4, p. 436, 2001.

[34] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, pp. 2–es, 2007.

[35] M. Barthélemy, "Spatial networks," *Physics reports*, vol. 499, no. 1-3, pp. 1–101, 2011.

[36] J.-P. Siedlarek, "Intermediation in networks," *Federal Reserve Bank of Cleveland, Working Paper No. 15-18*, 2015.

[37] E. Lopez-Rojas, A. Elmir, and S. Axelsson, "Paysim: A financial mobile money simulator for fraud detection," in *28th European Modeling and Simulation Symposium, EMSS, Larnaca*, pp. 249–255, Dime University of Genoa, 2016.

[38] M. Weber, J. Chen, T. Suzumura, A. Pareja, T. Ma, H. Kanezashi, T. Kaler, C. E. Leiserson, and T. B. Schardl, "Scalable graph learning for anti-money laundering: A first look," *arXiv preprint arXiv:1812.00076*, 2018.

[39] T. Suzumura and H. Kanezashi, "Anti-Money Laundering Datasets: InPlusLab anti-money laundering datadatasets." http://github.com/IBM/AMLSim/, 2021.

[40] M. E. Newman, "The structure and function of complex networks," *SIAM review*, vol. 45, no. 2, pp. 167–256, 2003.

[41] J. Leskovec and R. Sosič, "Snap: A general-purpose network analysis and graph-mining library," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 8, no. 1, pp. 1–20, 2016.

[42] A. Hagberg, P. Swart, and D. S Chult, "Exploring network structure, dynamics, and function using networkx," tech. rep., Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[43] G. Csardi and T. Nepusz, "The igraph software package for complex network research," *InterJournal*, vol. Complex Systems, p. 1695, 2006.

[44] J. Alstott, E. Bullmore, and D. Plenz, "powerlaw: A python package for analysis of heavy-tailed distributions," *PLoS ONE*, vol. 9, p. e85777, jan 2014.

[45] S. Koyaku, "A python package for detecting core-periphery structure in networks." https://github.com/skojaku/core-periphery-detection, 2022.