Universiteit Leiden
Opleiding Informatica

Algorithms for detecting and analyzing
multiplex motifs in large-scale corporate networks

Name: Boyd Witte
Date: April 4, 2017
1st supervisor: Frank Takes
2nd supervisor: Walter Kosters

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Network analysis is a frequently applied method of studying relations in a dataset. Many network measures assume that each edge in the network conveys the same information. This allows us to express measures such as distance, but can also lead to an incomplete or incorrect view of the data as only one relation in the data can be taken into account. In this thesis we study motif detection on multiplex networks, in which multiple types of interaction occur at the same time. Motifs are subgraphs that occur more frequently in an empirical network than they would do in a synthetically created network. As such they are the basic building block of the network, and can express important structures for a specific type of network.

We create a multiplex network from a corporate database by joining two uniplex networks: an ownership network and a board interlock network. By comparing the motifs from uniplex networks to multiplex networks we show the difference in information between the two types of networks. To do so we first define what type of motif is needed to best capture important corporate structures. Then we extend that definition to multiplex motifs. We augment an existing algorithm for motif detection to multiplex motif detection and compare the frequency of motifs in the empirical network to their frequency in a collection of synthetic graphs. We find that multiplex motifs provide information on certain industry sectors, as several motifs contain significantly higher concentrations of a certain sector compared to the full dataset.

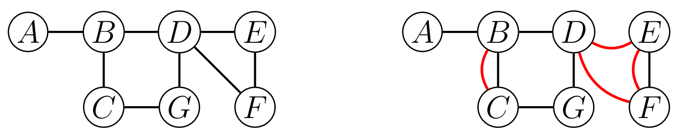
1 Introduction

Social networks have been around since as early as the 1930's [12]. They provide us with a way to visualize data and the connections that link different instances of that data. By depicting data instances as nodes, and drawing relations between them as lines, a graph is created. This helps us understand the underlying structure and enables us to analyze the data in a new way. For example, social networks have been used to model disease outbreaks [19], explain social behavior [39], and analyze economical business structures [38, 35]. Likewise social networks have been used to better understand how the real-world data came to be. By defining a set of rules, called a model, synthetic networks can be created that show similar properties to a real-world network, thus explaining how this data could occur.

Often analysis is only applied to a single kind of relation. Such a network is called a uniplex network. Think for example of an online social network, where everyone you know is classified as a *friend*. This network is easy to analyze as every relation has the same meaning. This gives the properties of a network, such as density and distance, a clear meaning, which can easily be interpreted by researchers. When two people A and C have a mutual friend B, but A and C are not friends, we can state that the distance between A and C is exactly 1 friend and thus that they are closely related in a network of friendship. We can imagine an implied relation. A and C might meet in real life, or could become friends themselves. This situation is measurable and understandable.

However, in the real world there are often multiple relations to consider. People are not only friends with each other. They are also family, neighbors or colleagues. Each of these relations explains how data instances relate, but all do so in a different way. This complicates network analysis. Different relations may be incomparable with each other and thus it is unclear how to express most network properties. Are two people who are friends just as strongly related as two people who are colleagues? To simply regard all relations as equally important or to leave out some relations could lead to wrong conclusions [16, 17, 23]. Thus we have a need to extend network analysis to networks that reflect all relations of the data.

A network with multiple types of relations is called a *multiplex* network. It is also referred to as a *multi-relational*, *multi-dimensional* or *multi-layered* network. Each type of relation is shown using a different edge color in the network visualization. Instances of data are not restricted to only one type of relation. Any node in a multiplex network can have multiple different relations with any other node in the network. A person can be both a friend and a colleague. See Figure 1 for an example.



(a) Uniplex network (b) Multiplex network

Figure 1: Uniplex and multiplex example

There are many examples of datasets with multiple types of relations which have been studied. A scientific collaboration network contains data on researchers and their published papers. Each node represents a researcher, and each relation between two nodes expresses those researcher's collaboration on a paper. The different scientific fields are translated into different types of relations. Similarly an actor network contains information on actors and the movies they starred in. Each node is an actor, and each relation between two nodes means those actors co-starred in a movie. The different movie genres are used as different types of relations. Such networks can reveal the preference of researchers or actors to collaborate with people already close to them, even if that person has done work in a different field or genre [3].

In this study we will focus on a corporate database. This corporate database contains information on companies such as revenue, directors, location, industry, and stock ownership. From this database we can extract multiple relations. It is possible to relate companies that share a board member in the board of directors, or to relate a company to another company in which it owns stock. We call these relations *board interlock* and *ownership* relations respectively.

Note that many of the given network examples are inherently bipartite, i.e., based on two different node types. Scientific collaboration is done via a paper, co-starring is done via a films, board interlocks are realized via directors. In all these situations a relation between two nodes is realized by a common node in a bipartite layer. This drastically changes the interpretation of node properties. An actor with many co-star relations might have starred only in one movie, which happened to include a lot of actors. The bipartite aspect of a network is important to keep in mind when explaining phenomena in a network.

The main analysis technique in this thesis is the detection of network motifs [26]. A network motif is a pattern in a network that describes a relation between multiple nodes that is unlikely to happen at random. These motifs are thus probably the results of real-world events, and can tell us more about the way data is linked. This information is neither a feature of the whole network nor a feature of a single node. Macro-level network features, such as density and diameter, provide information on the network as a whole.

Micro-level network features, such as degree and centrality, provide information on single nodes from a network. Instead motifs are a meso-level network feature and as such they provide information on a small group of nodes. Because motifs can be expressed using only structural properties, they can be extended to multiplex networks.

Motifs express those parts of a graph that are characteristic for that graph. In effect they also indicate the difference between networks as each network has a different set of motifs [24]. In a gene regulation network motifs tell which pathways are important for this specific gene. In a food web, a network visualizing hunter-prey relations, motifs can describe which species are dependent on each other [26]. The motifs describe a structure that is important for the dataset. In case of a corporate network, a motif can describe the structure that a company uses to regulate their subsidiaries, protect itself from bankruptcy or hostile and strategic takeovers, or create a diverse portfolio. With the recent economic turmoil in mind, motifs might prove very effective in indicating which corporate structures are hazardous in regards to financial stability. Since financial security between companies is not solely dependent on one type of relation, but on many, such as ownership, loans, transactions, and contracts. It is important to take into account the multiplex aspect of corporate data.

To discover the difference in information provided by a multiplex network as opposed to a uniplex network, we compare the motifs found in both types of networks and ask the question if multiplex motifs provide a better understanding of the basic building blocks of a corporate network compared to uniplex motifs and if so, does this indicate that in general multiplex motifs provide a better view of a network compared to uniplex motifs?

In Section 3 we discuss other studies on the topics of motif discovery and multiplex networks. In Section 4 we explain how to gather motifs and how to extend the existing motif recognition algorithms to find multiplex motifs. In Section 5 we introduce the empirical data. In Section 6 we expand on how to analyze and compare both the uniplex motifs and the multiplex motifs. In Section 7 we examine the found motifs, and apply previously discussed analysis and comparison methods. Finally in Section 8 we present a conclusion and provide leads for further research.

2 Notation

For the sake of clarity throughout this thesis, we will first define our notations. This keeps our definitions more concise and structured.

A network, or graph, is noted as G. A graph G consists of a set of nodes V and a set of edges E. Likewise $V(G)$ and $E(G)$ are the nodes and edges of graph G. There may only be one edge in the same direction between any two nodes.

A graph g is a subgraph of G if and only if $E(g) \subseteq E(G)$ and $V(g) \subseteq V(G)$, where all nodes incident with an edge in $E(g)$ occur in $V(g)$. The size k of a subgraph g is the number of nodes in $V(g)$.

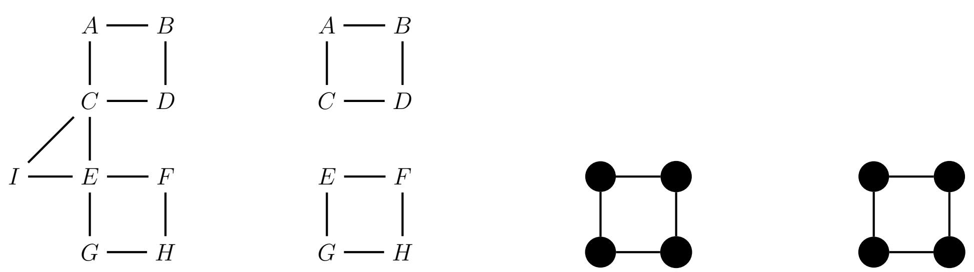
The pattern of a graph is its abstract representation without node labels or ID's. All isomorphic graphs thus have the same pattern. We say I is the collection of all patterns. $\mathcal{S}_k^i(G)$ is the set of subgraphs of pattern $i \in I$ and size k in graph G. Thus $|\mathcal{S}_k^i(G)|$ is the number of occurrences of pattern $i \in I$ of size k in graph G, which we call this the *frequency* of pattern i.

Motifs are based on subgraphs. A motif M is a pattern that is considered important according to a statistical metric. $M(G)$ denotes the set of subgraphs that form motif M in graph G.

An overview can be seen in Figure 2. Figure 2a shows an example graph. From this graph we take two size 4 subgraphs (Figure 2b). These subgraphs have the same pattern (Figure 2c). If this pattern is considered significant, then it is a motif (Figure 2d). Note that a pattern and motif are both abstract representations of a subgraph, and are in all aspects identical except that a motif has been deemed important.

A multiplex graph $\mathcal{G}$ contains multiple types of edges. The collection of edge types is called J. We use $E_j(\mathcal{G})$ with $j \in J$ to refer to a specific type of edge. There may only be one edge in the same direction of any type between any two nodes. Throughout this paper we will use different colors to show different types of edges. See Figure 1b for an example. Subgraphs of multiplex graphs are also noted as g.

Every graph has a degree sequence D, which is a list of all degrees per node. The function $D(v)$ with $v \in V(G)$ is the degree of node v in graph G. When the edges of a graph are directed, the degree sequence D is split into D_{in} and D_{out}, to specify in-degree and out-degree. When the graph is a multiplex graph, each type of edge has its own degree sequence D_j with $j \in J$. A degree sequence of edge type $j \in J$ that is directed is noted as $D_{j\text{-in}}$ and $D_{j\text{-out}}$. An example of a degree sequence for Figure 2a would be $[2, 2, 4, 2, 4, 2, 2, 2, 2]$, given the nodes are alphabetically ordered.



(a) Graph (b) Subgraphs (c) Pattern (d) Motif

Figure 2: Breakdown of motifs

We also make use of bipartite networks. A bipartite network is a graph where the nodes can be divided into two groups. Edges may only exist between nodes of different groups. See Figure 3a for an example. We express the degree sequence of such a network as a directed graph. For the example in Figure 3a this results in an in-degree sequence of $[0, 0, 0, 0, 1, 3, 1, 2]$, and and out-degree sequence of $[2, 1, 3, 1, 0, 0, 0, 0]$, given that the nodes are ordered as $A, B, C, D, 1, 2, 3, 4$.

(a) Bipartite Network (b) One-mode projection

Figure 3: Transforming a bipartite network to a one-mode projection.

A bipartite network can be transformed into a one-mode projection. This projection shows only one of the node groups, and draws edges between those nodes which are related via a node in the other group. In the example Figure 3a, node 1 and 2 would be related through node A. A complete one-mode projection of Figure 3a can be seen in Figure 3b.

The degree sequence can be used as basis to generated a random graph. A model that generates a graph with a similar, but not equal, degree sequence is called a canonical model. A model that strictly generates graphs with equal degree sequence is called a micro-canonical model.

3 Related work

In this section we will look at several studies in the field of motif recognition, network modeling and multiplex networks. We end this section with a discussion on how the presented studies relate to our study.

3.1 Pattern recognition

Motif recognition is a #P-hard task [33]. The most computationally expensive part of motif recognition is counting the frequency of all patterns. To do so each subset of nodes in a graph has to be compared against all known (possibly isomorphic) subgraphs. Therefore any motif recognition algorithm must be provided with a small enough input to finish in exponential time, or give an approximation of the frequency of motifs [33]. Many motif recognition algorithms work around these limitations by only finding a specific subset of patterns, or accepting only a specific type of graph.

CODENSE [14] is an algorithm that discovers only coherent dense subgraphs. CODENSE also uses graph specific information, such as node labels, to circumvent the "subgraph isomorphism proble", whose complexity is still unresolved. This reduces both the total number of subgraphs that should be analyzed and the time needed to analyze a subgraph, thus greatly reducing computational time.

SEuS [13] uses a summary method to reduce the size of a graph. This summary method combines nodes of the same type, and thus requires node labels to work. During subgraph frequency counting SEuS asks for input if a discovered subgraph should be counted or not. By doing so SEuS reduces the input and total number of subgraphs that should be analyzed.

G-Tries [31], FANMOD [37], and Subenum [34] find only induced subgraphs. Induced subgraphs contain all edges between its nodes, if those edges are present in the graph:

Definition 1 ***Induced Subgraph***
Subgraph g is an induced subgraph of G if for any pair of nodes $u, v \in V(g)$, it holds that if $(u, v) \in E(G)$ then $(u, v) \in E(g)$.

This prevents counting the same set of nodes but with fewer edges, as a different subgraph, thus reducing the number of possible subgraphs.

Another method to reduce computational complexity is to only determine if the frequency of a subgraph passes a certain threshold. Because SEuS uses graph summaries, SEuS can use the downwards closure property to quickly determine if the frequency of a subgraph passes the threshold. The downwards closure property states that the frequency of a subgraph does not increase as the size increases.

Similarly to SEuS, hSiGraM and vSiGraM [20] also use the downwards closure property to determine if the frequency of a subgraph passes a threshold. Instead of using a graph summary like SEuS, these algorithms search only for edge-disjoint subgraphs. The edge disjoint property states that no subgraphs shall contain the same edge. Thus each counted subgraph contains a unique set of edges. This allows for the downwards closure property to be used.

SUBDUE [15] finds many similar structures in a graph, and replaces them with a smaller description of the original structure. This compresses the graph and makes it easier to visually interpret, but it does not count the frequency of a subgraph. Replacing structures also prevents structures from overlapping. This is a similar restriction as the edge-disjoint property.

3.2 Multiplex Networks

With the multiple dimensions in a multiplex network a big problem arises: many of the properties used to describe a network must be redefined to include every dimension of a multiplex network. Some edge types might indicate a very significant relation, while other edge types provide no useful information on relations at all. In order to interpret any network property of a multiplex network, it is necessary to know how the edge types relate to each other. We are not aware of any study that solves this problem for the general case. However, motifs do not require network measures based on the information of edges. A motif can be extracted regardless of edge type. It matters only if there is an edge.

One study solves this problem for community detection [32]. The proposed solution is to combine all dimensions into one. With a linear equation the weight of each dimension is summed up. The linear equation is chosen in such a way that the quality of the resulting communities is maximal. For community detection such an approach can work. Communities have clear measures of quality, such as modularity. Other network properties such as diameter do not have a comparable quality measure.

Multiplex networks also offer new possibilities. One study aimed to create a feature vector for each node in a multiplex network [9]. These vectors could then be used to compare and analyze the nodes to find similar nodes, or make predictions about missing or future links. Likewise the authors of [17] created a multiplex network relating movies based on viewer ratings by selecting significant edges from a bipartite multiplex network.

3.3 Corporate network analysis

This study is not the first research to apply automatic motif recognition on a large corporate networks. In 2009 a study was done on size 3 motifs in a corporate transaction network [30]. By linking companies that provided services or materials for each other, a network of process chains is created. The size 3 motifs show which parts of the chains are interesting and can be used to classify companies based on their position in the motif.

Corporate networks have also been studied using others techniques. It is suggested that social network analysis can help improve economic models, and it has been shown that network models could have predicted the 2008 economic crisis [5]. More specifically a large scale ownership-network structure analysis provides a global overview of where influential companies are located in a corporate network [36]. Likewise a study on board interlock networks has shown that board member decisions can be explained by the links in a network [4].

3.4 Relations to this study

We have discussed many studies regarding our field of interest, and have seen that both ownership and interlock networks hold interesting information with regards to economic events. In this section we will conclude which of these studies would best suit our needs for motif recognition and analysis.

For motif discovery we have seen many algorithms that apply restrictions and boundaries to speed-up pattern recognition [13, 14, 20]. We consider these algorithms to be too restrictive. In this study we are interested in relations between all nodes. The algorithms that find induced subgraphs would therefore fit best. Especially the very recently published Subenum [34] is promising as it claims to outperform the other algorithms. SUBDUE [15] is also very promising. SUBDUE might detect similar patterns instead of just identical patterns. This could uncover other structures that help to better understand the network as a whole. However, because SUBDUE does not count frequencies and is also too restrictive on the patterns found, we will not use it. Thus the first point of action is to expand Subenum to work with multiplex networks.

4 Approach: Motif Recognition

In this section we will discuss the expansion of motif recognition to multiplex motif recognition. First we define what a multiplex motif is. Second we closely study the original Subenum algorithm. Finally we implement a multiplex version of the Subenum algorithm.

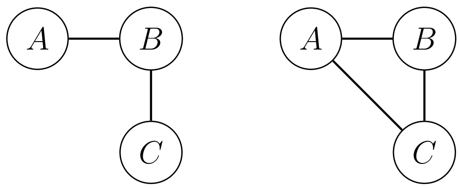
4.1 Multiplex Motifs

In Section 1 we have seen a brief introduction of a motif. However, in Section 3 we have seen that there are several different types of subgraphs to be found. Many motif recognition algorithms find induced subgraphs (see Definition 1). An induced subgraph captures all the available information between the set of nodes of a subgraph. We reason that leaving out information (edges) does not reflect on a real situation. An example is given in Figure 4, showing graphs with the same nodes. Figure 4a leaves out the edge between node A and C. Both graphs contain the subgraph $A - B - C$. In Figure 4a this subgraph makes sense as it conveys the way node A and C are related. However in Figure 4b the subgraph suggests that node A and C are not directly related, which is wrong. Not taking into account all the edges could give a wrong impression of the graph. For this reason we argue that all edges of every type must be taken into account for a multiplex subgraph. The graphs in Figure 5a and 5b therefore do not contain the same triangle subgraph A, B, C. With knowledge of the domain information represented by the graph, one could still determine for every edge type if it is important for a motif or not, and thus that the triangles shown in Figures 5a and 5b are actually the same. However we do not always have domain knowledge, and in the general case we can only be sure that two subgraphs are equal if and only if all edges of all types match. Therefore we need multiplex induced subgraphs.

We adjust Definition 1 of an induced subgraph to the definition of a multiplex induced subgraph.

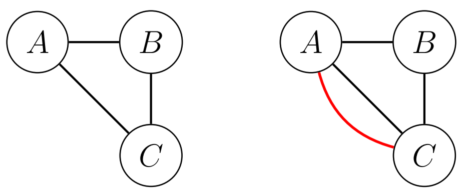
Definition 2 ***Multiplex Induced Subgraph***
Let g be a subgraph of the multiplex graph $\mathcal{G}$. Let J be all types of edges in $\mathcal{G}$.
Let $E_j(\mathcal{G})$ be all edges of type $j \in J$.
Then g is a multiplex induced subgraph if for any pair of nodes $u, v \in V(g)$, it holds for each type of edge $j \in J$ that if $(u, v) \in E_j(\mathcal{G})$ then $(u, v) \in E_j(g)$.

Furthermore we only look at connected subgraphs.



(a) Line shaped   (b) Triangle shaped

Figure 4: Induced graph example



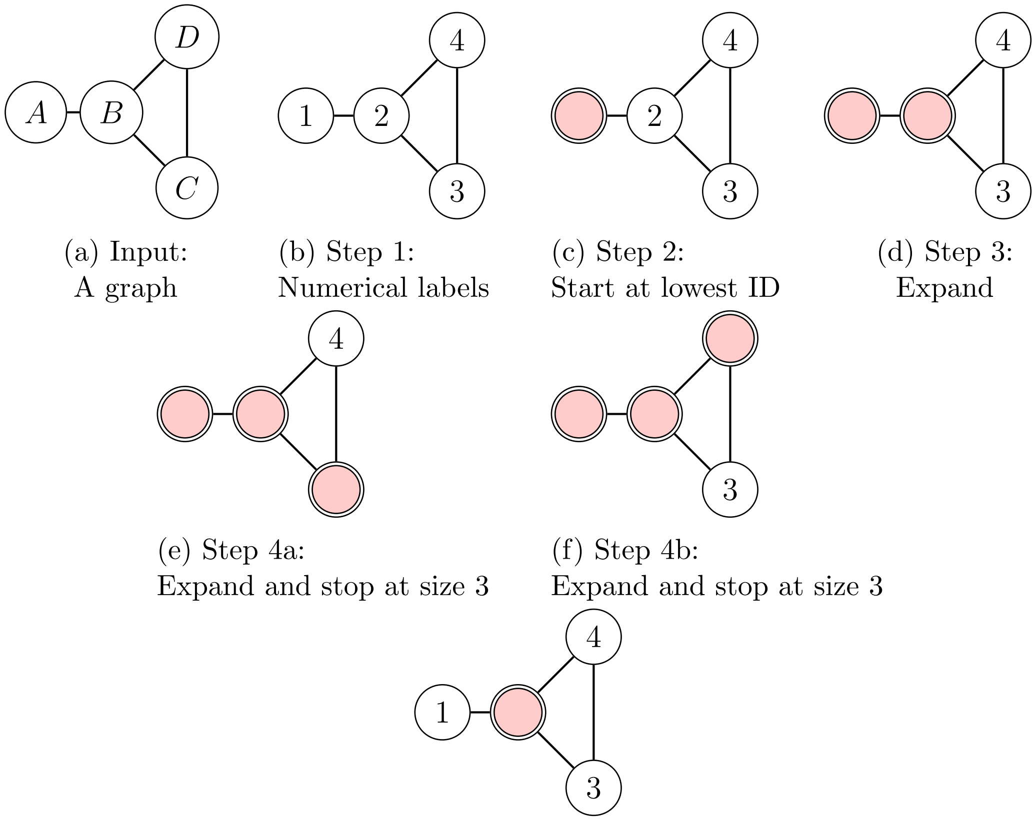
(a) Triangle shaped   (b) Multiplex Triangle shaped

Figure 5: Induced multiplex graph example

4.2 Subenum

In order to change the Subenum algorithm to work on multiplex graphs, we must first understand the exact mechanics of the algorithm. We briefly discuss the algorithm in this section. For a more detailed description see [34].

The Subenum algorithm is based on the Enumerate Subgraph algorithm (ESU) proposed in [37]. This algorithm counts subgraphs in directed unweighted graphs. To find all subgraphs of size k, ESU labels all nodes with a unique numerical ID (Figure 6b). It then loops over every node starting at the lowest ID node (Figure 6c), recursively expanding on every neighboring node with a higher ID (Figure 6d). The expansion lasts until the set of nodes is of size k (Figures 6e and 6f). The resulting set of nodes with all edges between them is a subgraph. The subgraph is then given a canonical label with the Nauty [22] algorithm. This label is guaranteed to be equal for all isomorphic subgraphs.



(a) Input: A graph
(b) Step 1: Numerical labels
(c) Step 2: Start at lowest ID
(d) Step 3: Expand
(e) Step 4a: Expand and stop at size 3
(f) Step 4b: Expand and stop at size 3
(g) Step 5: Restart at next lowest node

Figure 6: ESU algorithm example for subgraphs of size 3

Subenum aims to parallelize the ESU algorithm. Each recursive expansion per node could be done in parallel, as they are unrelated to each other. However the load balance per thread can be very uneven as the number of expansions is directly related to the degree of a node being expanded. Therefore it expands on edges instead of nodes. Nodes are given a unique numerical ID. For each edge, Subenum expands on the neighboring nodes of both nodes of the edge, if and only if the numerical ID of the neighboring node is higher than the lowest ID of the node of the edge. That node is then added to the subgraph node set, and a new set of neighboring nodes is defined. Each edge can be analyzed in a separate thread, achieving well balanced parallelism.

Subenum also changes the way subgraphs are checked for isomorphism. Keeping all found isomorphic subgraphs in the main memory rapidly becomes inviable when the number of subgraphs becomes large. This quickly happens when the input graph is large or the subgraph size is large. Subenum uses a two phase isomorphism check to work round the limited main memory. In the first phase when a subgraph is discovered, the nodes are relabeled into an *ordered form*. The ordered form labels all nodes in the subgraph based on their degree. The lowest degree node gets the lowest ID, the node with the highest degree the highest ID. It is undefined what happens with nodes that have the same degree. This does not matter as this can only create different node labels which Nauty will correct later. Graphs that are isomorphic will likely result in the same ordered labeling and thus result in identical adjacency matrices. When the memory limit is reached, all graphs (and their frequencies) are written to disk. Any subgraph added to memory will then be seen as a new graph again. The resulting output can thus contain multiple entries of graphs that are identical. Either because the ordered labeling was different or because the memory has been written to disk. In the second phase Nauty is used to give each subgraph (labeled in ordered form) a labeling in canonical form. This guarantees that any isomorphic subgraphs receive the same label. See Figure 7 for an overview of all processes. The input of the whole process is an edge list. Each line in an edge list is a tuple describing an edge in the graph. Each tuple consisting of a source node, a target node, and an ID for an edge type. The output is a motif list. A motif list is a file where each line specifies a unique motif identifier, followed by the frequency of that motif.

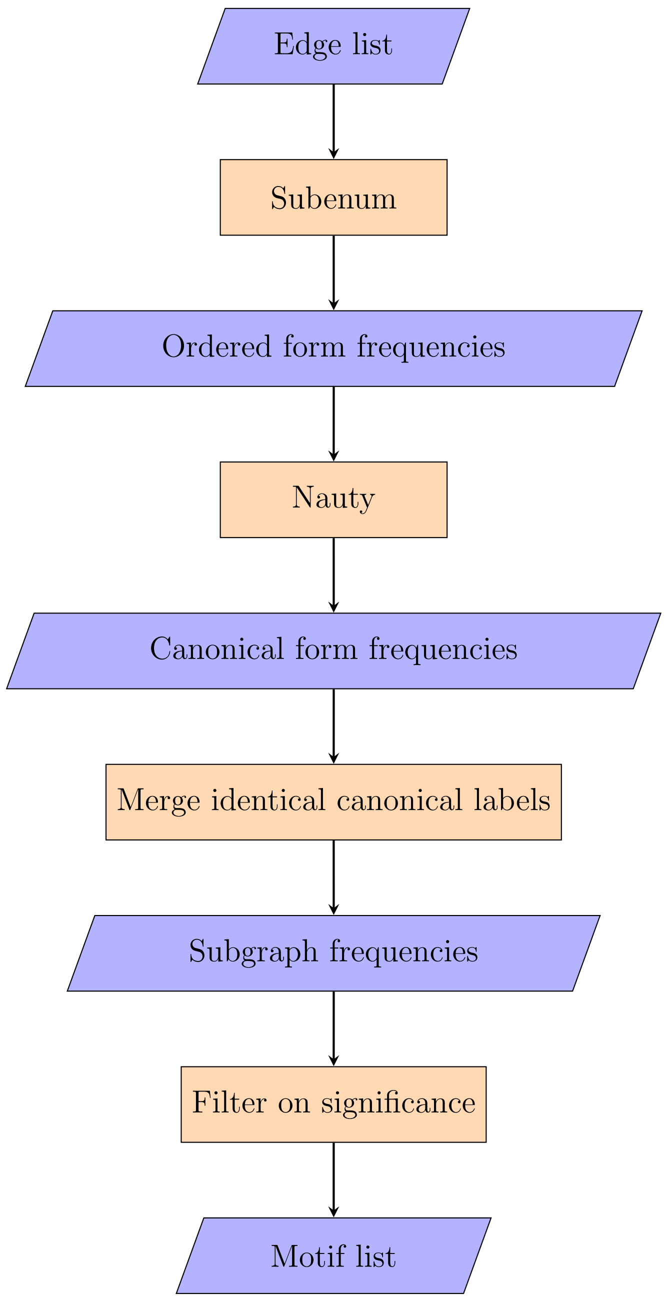


Figure 7: Subgraph enumeration flowchart

4.3 Multiplex Adaptation

We now have a clear definition of multiplex motifs, and a good understanding of the Subenum algorithm. With this we can adapt the algorithm to work with multiplex networks. The adaptation is a two step process. The first step is to adapt the motif recognition algorithm Subenum, the second step is to adapt the isomorphism detector Nauty.

Subenum and Nauty can already process directed graphs. Thus they only have to be adapted to handle multiple edge types. Any multiplex graph (without edge weights) can be seen as a weighted graph. Instead of different edge types, a single edge is shown which is given a binary label. This binary label is based on the edge types present between two nodes. First we fix an order of all the edge types present in the network. We then create a bit string for each edge in the weighted graph. The length of the string is equal to the number of edge types. A 1 in the bit string indicates the type of edge that occurs between the two nodes, according to the fixed order of the edge types and the location of the 1 in the string. A binary label can be seen as an edge weight. This is illustrated in Figures 11a, 11b and 11c. Note that by rewriting a multiplex network to a weighted graph, the ability to express weighted multiplex networks is lost. We do not consider weighted graphs in this thesis, so this does not pose a problem. However, for future studies this could be problematic. To preserve weights, one could encode weight and edge type into one number. This number should be unique for every weight and edge type combination. Furthermore, this conversion to a weighted graph does not make the graph equal to an actual weighted graph. The used weights only encode edge types, and do not express actual weights.

The basic principle of the ESU algorithm is that it detects *if* there is an edge, regardless of the type or weight. When a subgraph of size k is found, the label given to it is based on the adjacency matrix of the subgraph. See Figure 8a for an example. This idea can be extended to weighted graphs. The algorithm detects a motif, and then a label is created based on the adjacency matrix with weights. See Figure 8b for an example. Both graphs from Figure 8 are triangles (see Figure 4b), but their labels differ significantly due to the edge weights. This means that to adapt Subenum for weighted graphs, we only have to adapt the label constructor so that it accounts for the edge weights. The same holds for directed graphs, for which an example can be seen in Figure 9. In this example we see a directed multiplex graph, which would, if rewritten into a weighted graph, have an adjacency matrix as shown in Figure 9b. This matrix can be converted into a graph label just as the adjacency matrix in Figure 11c can be converted.

$$\begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{c} \begin{array}{ccc} A & B & C \end{array} \\ \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \end{array} \rightarrow 011101110$$

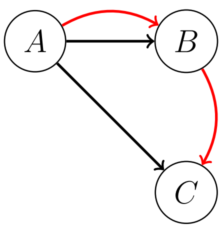
(a) Graph label of Figure 4b

$$\begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{c} \begin{array}{ccc} A & B & C \end{array} \\ \begin{pmatrix} 0 & 3 & 1 \\ 3 & 0 & 2 \\ 1 & 2 & 0 \end{pmatrix} \end{array} \rightarrow 031302120$$

(b) Weighted Graph label of Figure 11c

Figure 8: Graph Labels

Unfortunately Nauty is not as versatile as Subenum. Nauty does not support weighted graphs, but does suggest a solution in its manual. We have seen that any unweighted multiplex graph $\mathcal{G}$ can be transformed into a weighted uniplex graph G. Any weighted uniplex graph G can be converted into a colored-node graph G', which Nauty does support.



(a) Directed multiplex graph

$$\begin{array}{c} \\ A \\ B \\ C \end{array}\begin{array}{c} \begin{array}{ccc} A & B & C \end{array} \\ \begin{pmatrix} 0 & 3 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix} \end{array} \rightarrow 031002000$$

(b) Directed weighted Graph label of Figure 9a

Figure 9: Directed Graph Labels

Graphs with colored nodes are similar to multiplex graphs, but instead of multiple edge types they have multiple node types (colors). Each node color represents a different kind of node. In general this can be used to create a network where not every node is the same data instance. For example, one can combine nodes that represent humans with nodes that represent machines. See Figure 10 for an example.

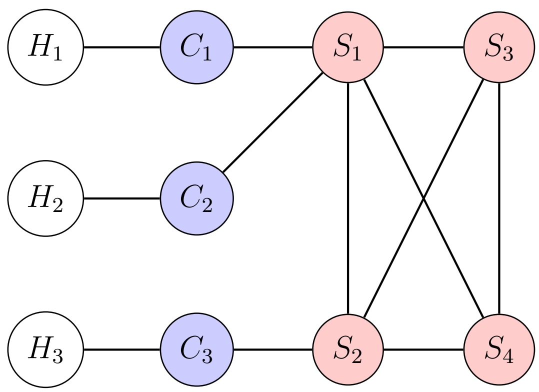
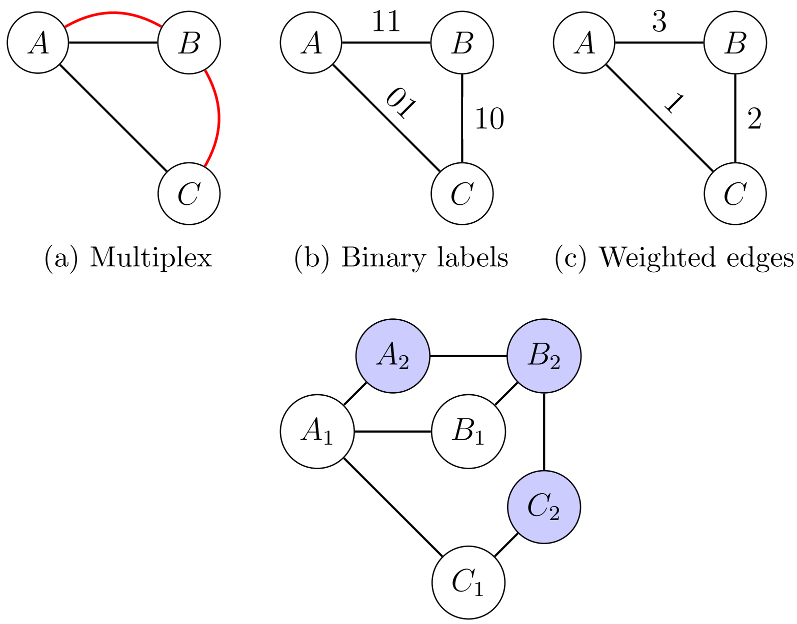


Figure 10: Colored node network: Humans H on computers C connected to a server cluster S.

Graphs with colored nodes can be used to express multiplex graphs. This method is similar to Nauty's suggestion for expressing weighted graphs. To do so we create a new graph G' from a graph G, where G is a graph with binary labels representing multiplex graph $\mathcal{G}$. For each node in $V(G)$, a set of colored nodes is created in G'. The number of colored nodes is equal to $|J|$, where J is the set of all edge types in $\mathcal{G}$. So for every node $X \in V(G)$, a set $\{X_1, X_2, \ldots, X_{|J|}\}$ with different colors is created in G'. Every X_j is connected with X_{j+1} with an undirected edge, for $0 < j < |J|$. This creates a string of colored nodes in G' for every node in G. Each color is then used to express a single edge type, according to the binary label. So an edge between two nodes X_j and Y_j is used to express the j^{th} edge type encoded in the binary label. An example can be seen in Figure 11d, where the multiplex graph from Figure 11a with two types of edges is shown rewritten with two types of colored nodes.

5 Data and network properties

Before we apply the motif recognition algorithms to real-world data, we will first analyze the general properties of the data. For this study we will use a database from the University



(a) Multiplex (b) Binary labels (c) Weighted edges

(d) Colored nodes

Figure 11: Equal graphs
There are 2 types of relations: Type 1, shown in black straight lines; Type 2, shown in curved red lines. Each graph conveys the same network information.

of Amsterdam [1], made available through guest-membership of the CORPNET group. This database is a corporate database containing information on corporations from all over the world. As the entire world dataset would contain too much data to process, this study focuses on Germany only.

5.1 Data properties

Germany, or rather the dataset, contains a total of 309 521 companies. For each company the database provides information on the number of employees, revenue, total assets, and sector, board composition and ownership. Most data is complete, but for 35 709 (11.5%) companies the data on employees is missing, and for 34 753 (11.2%) companies the asset value is missing. The data on sectors and revenue is complete.

Table 19 shows the division of companies amongst the sectors. It shows that most German companies are in the industrial sector (91%), with the financial sector as the second largest sector with 8% of all companies.

Figure 12 shows a scatter plot of companies based on their number of employees and revenue. It shows that generally speaking the revenue increases when the number of employees increases. As the revenue increases, so do the total assets. Most sectors follow this trend (Figure 13) except for the venture capital sector (Figure 13h), which is scattered on the lower end of both revenue, employees and assets.

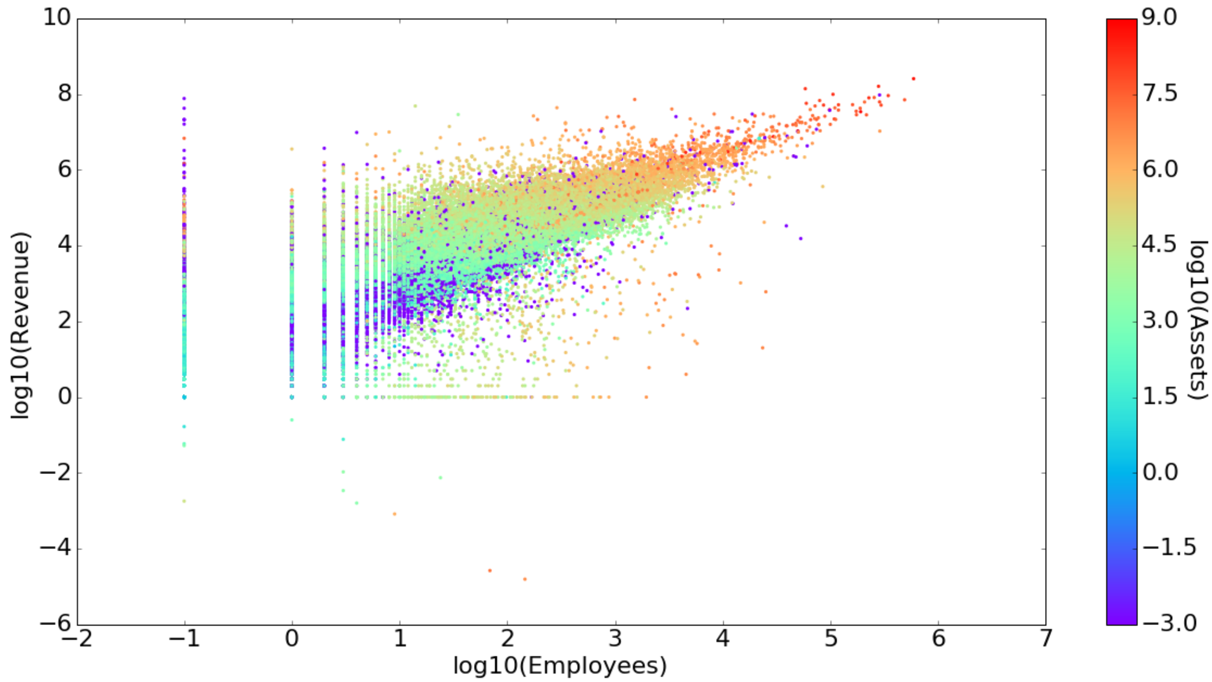
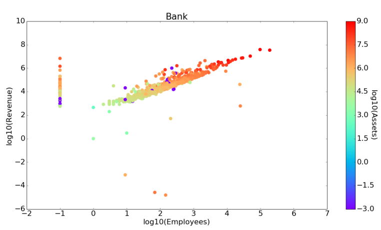
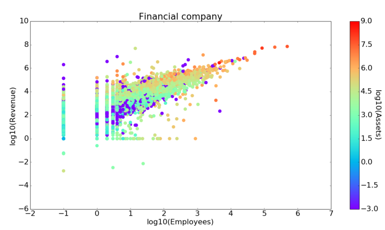


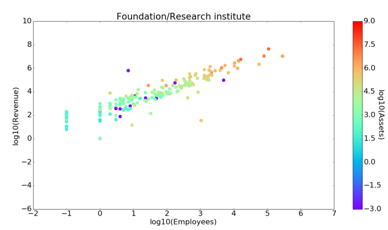
Figure 12: Scatterplot of firms, based on employees and revenue. Symmetrical log scale. Firms without data on employees are displayed with with -1 employees.



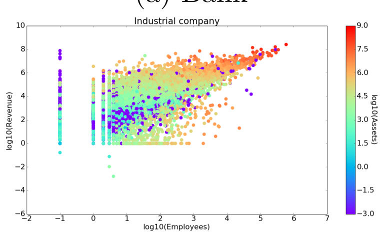
(a) Bank



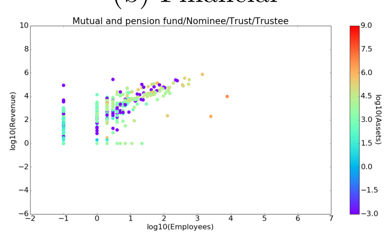
(b) Financial



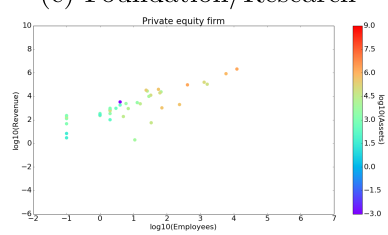
(c) Foundation/Research



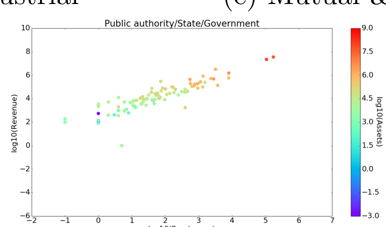
(d) Industrial



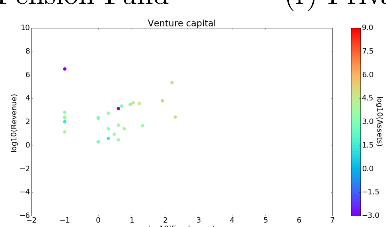
(e) Mutual & Pension Fund



(f) Private Equity



(g) Public Authority



(h) Venture Capital

Figure 13: Scatterplot of firms based on employees and revenue, per sector.

Table 14 shows three different top 10 companies, one ranked by employee count, one by total revenue, and one by owned assets. The top 10 employers are responsible for 12.42%

of the total employees in the database not counting those companies without data, and 80% of these companies are industrial. The top 10 earners account for 11.40% of all revenue in the dataset. All these companies are industrial. When the data is ranked on total asset value, the top 10 is almost entirely from the banking sector, with the exception of *DE2070000543* from the industrial sector. The top 10 asset owners hold one third of all asset value not counting those companies without data.

Note that 60% of the top 10 employers and top 10 earners overlap.

	Employees (12.42%)	**Revenue** (11.40%)	**Assets** (33.25%)
1	DE2070000543	DE2070000543	DE13216
2	DE5030147191	DE7330530056	DE40257
3	DE2010000581	DE5050056484	DE13223
4	DE2010198197	DE8170003036	DE13190
5	DE4230120196	DE7290397825	DE13328
6	DE7290397825	DE2010000581	DE17881
7	DE7330000658	DE5050314384	DE2070000543
8	DE7330530056	DE5030147137	DE46802
9	DE5050314384	DE5110120872	DE40185
10	DE5030147137	DE2150004419	DE47734

Figure 14: Top 10 companies. Colors mark companies that appear in multiple top 10's

5.2 Network properties

We use the data to create three networks:

1. An ownership-network. A directed network that shows which corporations own other corporations. Ownership edges are drawn in red, as seen in Figure 15.

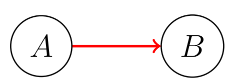


Figure 15: Ownership network example: company A owns a share of company B

2. A board interlock network. An undirected network that shows which corporations share board-members. Board interlock edges are drawn in black and as two directed edges, as seen in Figure 16.

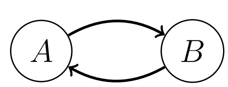


Figure 16: Interlock network example: company A and B share a board-member

3. A multiplex network that contains both ownership and board interlock relations. When a black interlock and red ownership overlap, we draw a single blue *multiplex* edge, as seen in Figure 17.

Figure 17: Multiplex example: company A owns a share of B, and both share a board-member. Drawn as separate edges and with a single blue edge.

The dataset contains many relations of ownership and board interlock, but not every relation is of importance. First we filter out unrealistic cliques. These cliques are caused by outliers of the data. In the board interlock network we see such a clique being formed by a small group of directors. We notice that any of these board members works in at least 85 different companies, whereas other directors work in no more than 50 companies. Furthermore all these directors work for the same companies. As a result an unrealistically large clique is created. This is likely an administrative structure of the same company, and thus not of importance for this analysis. We thus remove any board members with 85 or more positions. For ownership we consider a relation when at least 3% of a company is owned. This reduces the number of edges by half, and also limits the maximum in-degree of any node to 33. After removing all ownership relations with less than 3% weight and all directors with 85 or more positions, we remove all companies from the dataset that have no relations left. This is the processed dataset used for this study.

In the multiplex network, an ownership edge and interlock edge between the same two companies is combined. Of all edges in the multiplex network, 5.9% is a multiplex edge, which connect a total of 17 679 (23.5%) firms. The total number of companies and relations per network can be seen in Figure 18.

Network	**Nodes**	**Edges**
Ownership	37 724	31 506
Board interlock	61 209	175 108
Multiplex	75 224	195 073[1]

Figure 18: Network statistics

[1] This includes multiplex edges.

Not every company is present in every network, as is evident from the number of nodes shown in Table 18. Most companies are free to buy stock in other firms with which they share no board members, and likewise most board members are free to work at other companies regardless of the ownership relation with firms they are already employed at. There might be legal issues that certain companies are not allowed to be be involved with other companies, or that a person working in a certain sector is forbidden to work in another sector, but this is not the general case. It is thus possible that some companies have no interlock ties, and others have no ownership relations with any other firm. The interlock and multiplex networks however contain all companies from Table 14. The ownership network omits four companies: DE4230120196 from the top 10 employers, DE7290397825 from both the top 10 employers and earners, DE5110120872 from the top 10 earners and DE40185 from the top 10 asset owners. Table 19 shows the percentage of each sector per network. Note that the set of nodes in the multiplex network is equal to the processed dataset.

Sector	Ownership		Interlock		Multiplex		Dataset	
	count	%	count	%	count	%	count	%
Bank	474	1.25%	865	1.41%	972	1.29%	1 493	0.48%
Financial	4 648	12.32%	6 250	10.21%	8 338	11.08%	25 239	8.15%
Foundation/Research	55	0.14%	51	0.08%	88	0.12%	199	0.06%
Industrial	32 350	85.75%	53 767	87.84%	65 484	87.05%	281 830	91.05%
Insurance	19	0.05%	26	0.04%	34	0.05%	103	0.03%
Mutual & Pension Fund	112	0.30%	175	0.29%	213	0.28%	489	0.16%
Private Equity	29	0.08%	30	0.05%	37	0.05%	44	0.01%
Public Authority	22	0.06%	31	0.05%	41	0.05%	98	0.03%
Venture Capital	15	0.04%	14	0.02%	17	0.02%	26	0.01%

Figure 19: Companies per industry sector per network

6 Approach: Motif Analysis

In this section we discuss how to analyze motifs found in the empirical network. In order to do any further analysis on the data, we must first answer the question which motifs are interesting. Second, we are interested in the extra information a multiplex network contains compared to its single-edge-type network equivalents. Thus we compare the motifs from the multiplex network to the motifs from the ownership network and the board interlock network.

6.1 Motif Selection

Thus far we have regarded motifs and patterns mostly as the same thing, but did refer to them as motif. However [26] states that motifs are "those patterns for which the probability of appearing in a randomized network an equal or greater number of times than in the real network is lower than the cutoff value". Any pattern should thus be tested for significance. This ensures the pattern is not just the result of random chance, but represents a real-world structure.

We apply two methods for declaring a pattern significant. The first method compares the frequency of a pattern with the frequency in randomly generated graphs. This method is suggested by [26]. The second method declares a pattern significant solely on its frequency in a graph. By using an estimate of a random graph, the frequency can be compared to a randomly generated network without actually generating one, as suggested by [37] and [25]. Finally we consider a warning given in [26]. Some motifs could be functionally interesting, but not significant. Therefore we will also compare significant motifs to functionally interesting motifs, based on the opinion of experts.

6.1.1 Significant Subgraphs

In order to detect which patterns are significant in the empirical network, we will need to compare their frequency in the empirical network to their frequency in a set of randomly generated graphs. Modeling the ownership network is straightforward. This network consists of one type of node, with one type of edge, where each edge represents a direct relation between two companies. Unfortunately this is not the case for the interlock network and the multiplex network.

Bipartite modeling The interlock network is actually a one-mode projection of a bipartite network. It is a network of relations between companies, but the underlying data is a bipartite set of board members and companies. A board member can only have relations with companies and visa versa. A random model that does not take this into account can easily create patterns that do not occur in the real world. Think for example of a triangle, as shown in Figure 5a. A model that directly creates relations between companies only needs three companies with a degree of two or higher to create a triangle. While a model that takes into account the bipartite division would either need a board member that manages three companies, or a set of board members that link the companies in a triangular way. See Figure 20 for examples.

(a) A single board member $D1$ forming an interlock triangle between the firms A, B and C.

(b) Multiple board members $D1$, $D2$ and $D3$ forming an interlock triangle between the firms A, B and C.

Figure 20: Two examples how a bipartite network can form the triangle pattern in Figure 5a.

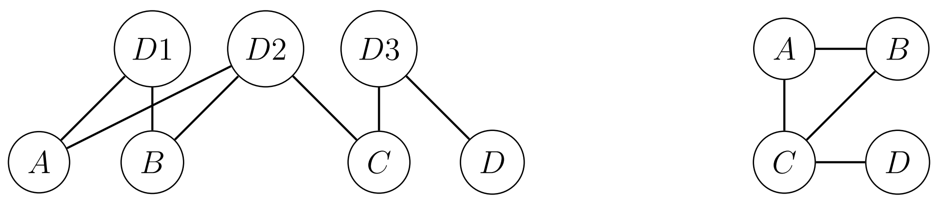
When the model is based on a degree sequence and uses the underlying bipartite division, it follows the real world more closely. We can then use the one-mode projections of the random bipartite network as a random interlock network for this study. Unfortunately it proves difficult to model a bipartite network so that the resulting one-mode projection resembles the original one-mode projection. The one-mode projection can easily end up with too many edges or with only a specific selection of edges. We shall thus discuss several methods for bipartite modeling that keep the one-mode projection in mind.

The first method is the simplest to apply. A random model creates edges between directors and companies. Then a one-mode projection is made. This preserves all information and requires no special selection of edges or analysis of the random network. However, in practice this method creates too many edges in a random configuration based on our dataset. The average number of edges can be seen in Table 21 under *empirical directors*.

Bipartite network	**Avg. Edges in one mode projection**
Empirical directors	235 996
Removal redundant directors	190 908
Minimal set of directors	125 190
Edge selection	-
Original one-mode projection	175 108

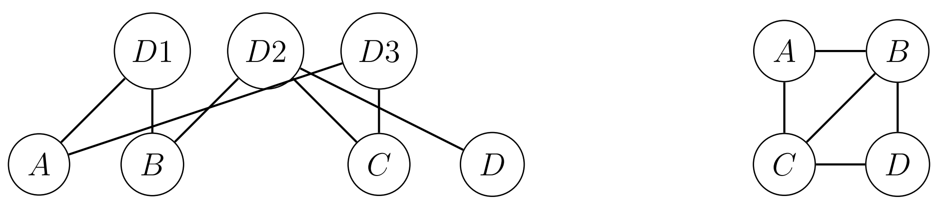
Figure 21: Average edges over 20 randomly created networks created by the stub-matching model, with different bipartite networks as base.

In a real-world bipartite network, a layer can act in a non-random fashion, and pick edges based on existing edges. In our dataset this can be seen as board members who enjoy working together, and thus will likely work at the same companies. These board members are redundant when creating a one-mode projection from the empirical data, but can become non-redundant when a model randomly connects them to companies. An example can be seen in Figure 22 and Figure 23. Both figures show a bipartite network with equal degree sequence, but the one-mode projection of Figure 23 has one additional edge compared to the one-mode projection of Figure 22. Overall this method creates a very different number of edges in a one-mode projection from a random network, compared to the one-mode projection from the real-world data.



(a) Bipartite network (b) One-mode projection

Figure 22: One-mode projection of three board members and four companies.



(a) Bipartite network (b) One-mode projection

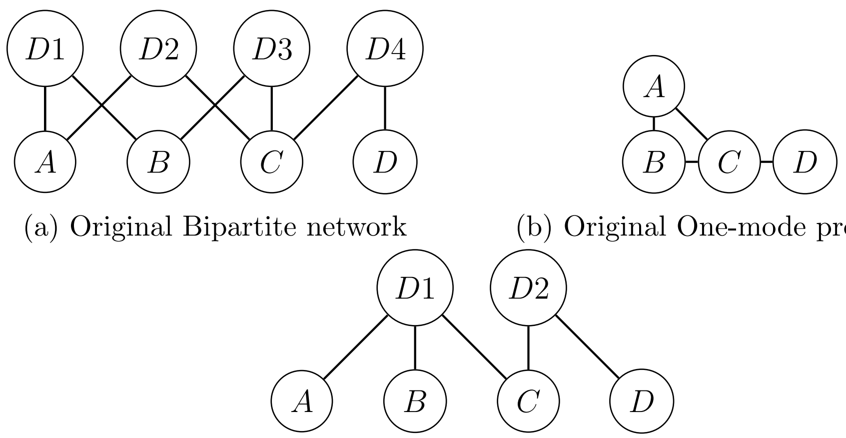
Figure 23: One-mode projection of three board members and four companies.

To solve the problem of redundant board members, one can remove all board members from the dataset that have no edges that are not also defined by other board members. In Figure 22 this would be director $D1$. The number of edges created based on this network resembles the original one-mode projection the most, as can be seen in Table 21, under *removal of redundant directors*. Despite the good result, this method is not preferred. A board member is only redundant if another board member provides identical information. This can only happen with a director that has at least as many edges as the redundant director. In this way a bias to larger directors is introduced.

Another way to solve the problem of redundant board members, is to create a fictional set of board members such that the no fictional board member is completely redundant. This method does not use the original bipartite network, but instead creates a new bipartite network called a two-mode projection. This two-mode projection contains the smallest number of nodes (board members in our case) necessary to create a bipartite network. Figure 24 shows an example. Given a one-mode projection (Figure 24b) from a bipartite network (Figure 24a), a minimal two-mode set is created (Figure 24c) such that its one-mode projection is equal the one-mode projection of the original bipartite network.

To create such a minimal two-mode set, the largest possible clique for each edge needs to be found. In the example network from Figure 24b we find two cliques. Edges $A - B$,

$B-C$ and $A-C$ are all part of the clique ABC. Edge $C-D$ is part of clique CD. For each found clique, a two-mode projection node D_x (board member in our case) is created. Then for every edge $U-V$ from the one-mode projection, $U-V$ is removed and a new edge is created between U and D_x, and V and D_x. This two-mode projection is unlikely to be similar to the original bipartite network.



(a) Original Bipartite network (b) Original One-mode projection

(c) Minimal two-mode set

Figure 24: Creating a minimal bipartite set

In practice however, cliques overlap, and an edge between two nodes is often attributed to multiple two-mode projection nodes. So while no board member is completely redundant, many edges still are. Furthermore the randomness is limited with this technique. The created artificial bipartite network contains no redundancy in director positions, and the one-mode projection has exactly the correct number of edges. A random model could thus only introduce redundancy, ensuring that any resulting random network has at most the correct number of edges, but likely less. Hence modeling based on a minimum two-mode set will result in a one-mode projection with a different number of edges. The average number of edges can be seen in Table 21 under *minimal set of directors*.

A final method for bipartite modeling with interest in the one-mode projection is to make a selection of edges to put in the one-mode projection. If information about the edges is known, it is possible to select only those edges above a certain weight, or of a certain type [17]. This would create a one-mode projection with the same number of edges as the real data. However, the edge-selection process is not ideal in this situation. In our case each edge is of equal importance, thus we cannot make a selection without losing valuable information.

We do think that using a bipartite model with empirical directors is the correct choice, as it follows the generative rules of the real data the most and introduces no bias. Thus the goal is to closely resemble the original bipartite network. Since we do not want to make a selection of edges, and simulating fictional board members provides no additional benefits over using the actual board members, we use the first discussed method where a bipartite network is modeled without any selection or synthetic data.

Multiplex modeling The multiplex network does not only deal with different types of relations, but also with the dependencies between them. In many cases it is likely that different types of edges are not created independently from each other. In a scientific collaboration network one can see that researchers prefer to work with people from their field of study that are close to them. This preference carries over when researchers collaborate with researchers from another field of study. This property is called triadic closure. With this knowledge one can create a multiplex preferential attachment model that resembles the original network [3]. If every node in a network has at least one edge of every type, it is always possible to apply a preferential attachment method such that the probability of forming a triangle is high.

The dataset used in this study also shows dependencies between the different edge types. A quick test shows that merging a random ownership network and a random interlock network creates a multiplex network with very few edges that are both ownership and interlock. In the real data we have 5.9% of all edges that are both ownership and interlock. Unfortunately, not every node has an edge of every type, making the proposed multiplex model from [3] not suitable for this case. In order to preserve the relation between all edge types, we use a degree-sequence based model with three different degree-sequences:

1. Ownership edge
2. Interlock edge
3. Combined edge

By specifically stating how many ownership and interlock edges are dependent on each other, we force te model to create around 5.9% combined edges.

Model options The models used to generate random graphs have great impact on which patterns are present in those graphs [24, 21]. Thus we must use a model that describes our corporate networks best. Models such as the Erdős-Rényi model [10] (which is used in [26]) and the Barabási-Albert model [2] are well established and studied, but would generate a too generic network for comparison. The model used should also generate graphs with the same degree sequence as the original network [37], as these kinds of graphs fit a board interlock and ownership network very well [29]. Preferably the model should also be stochastic, so that it does not enforce too harsh restrictions [27]. Therefore we test three different graph models that are based on a degree sequence. These models are the Chung-Lu model [8], the Park-Newman model [28], and the Stub-matching model [7]. With each model we generate 500 graphs. For each of these sets of graphs, we measure the average number of nodes, edges, and degree distribution. For more information on the implementation of the models, see Appendix A.1.

Table 25 and Table 26 show the average number of nodes and edges (and standard deviation σ) for each model. We see that all models are very consistent as they have relatively small standard deviations. The Stub-matching model is by far the most precise, generating almost the exact number of edges. It also generates the exact number of nodes, but this is enforced by the model itself as it is a micro-canonical model. Both Chung-Lu and Park-Newman are canonical models, and thus have more freedom in generating nodes and edges. However, in effect this means that with this specific degree distribution, the number of nodes is much lower than desired.

Figures 27, 28, 29 (ownership) and 30, 31, 32 (interlock) show the difference in average degree distribution of the models compared to the real-world network. The blue bars (best

Interlock Model	**Avg. Nodes**	(σ)	(%)	**Avg. Edges**	(σ)	(%)
Stub-matching	101 095	(0)	(100)	100 375	(2)	(100)
Chung-Lu	51 307	(16)	(51)	43 772	(14)	(44)
Park-Newman	49 595	(16)	(49)	40 419	(13)	(40)
Real world	101 095		(100)	100 377		(100)

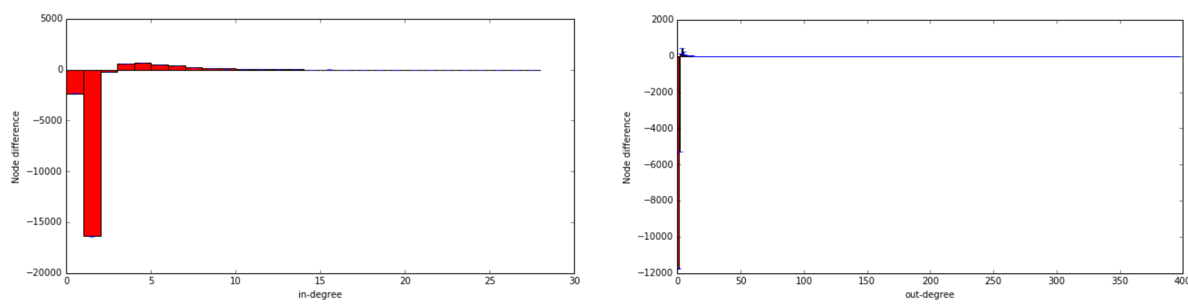
Figure 25: Network statistics of models generating random interlock networks.

Ownership Model	**Avg. Nodes**	(σ)	(%)	**Avg. Edges**	(σ)	(%)
Stub-matching	37 724	(0)	(100)	31 498	(3)	(100)
Chung-Lu	25 502	(40)	(68)	42 070	(52)	(134)
Park-Newman	21 702	(36)	(58)	30 335	(43)	(96)
Real world	37 724		(100)	31 506		(100)

Figure 26: Network statistics of models generating random ownership networks.

visible in Figure 29a) indicate the standard deviation of the average. Again we see that all models are very consistent, as the standard deviations are relatively small. Both Chung-Lu and Park-Newman generate far less nodes with an in-degree of 1 and out-degree of 0. The Stub-matching model is again very similar to the real-world data, with an average difference of at most 4 nodes.

These tests show that the stochastic models have relatively large errors compared to the stub-matching model. The stub-matching model is by far the most similar to the real-world data, and therefore the preferred model to test significance on. For motif recognition we generate 1 000 random graphs, as suggested in [37].

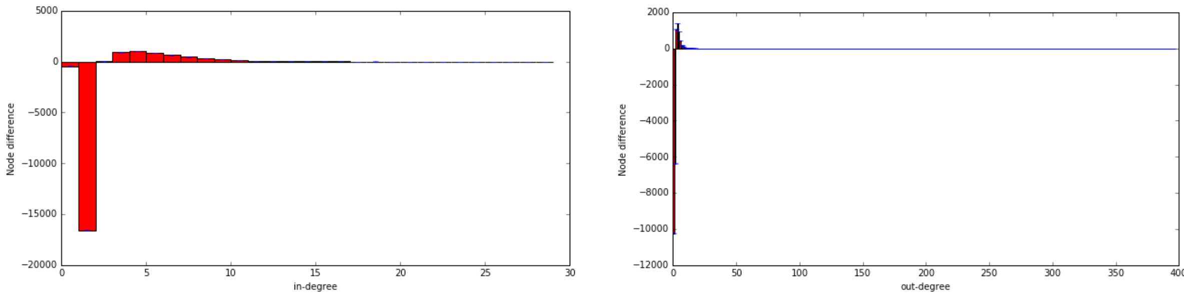


(a) In-degree

(b) Out-degree

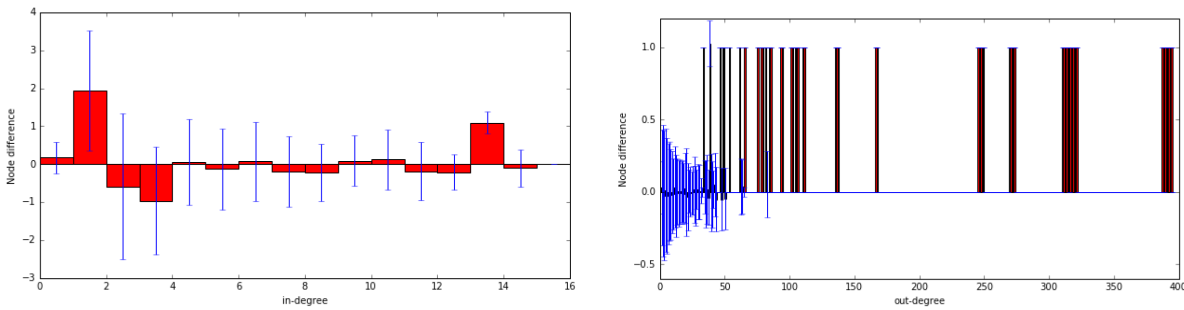
Figure 27: Difference in degree distributions between Park-Newman generated graphs (based on the ownership network) and the ownership network.

Significance Significance of a pattern will be determined using the ratio described in [21]. The ratio R of a pattern i of size k in graph G is defined as seen in Equation 1:



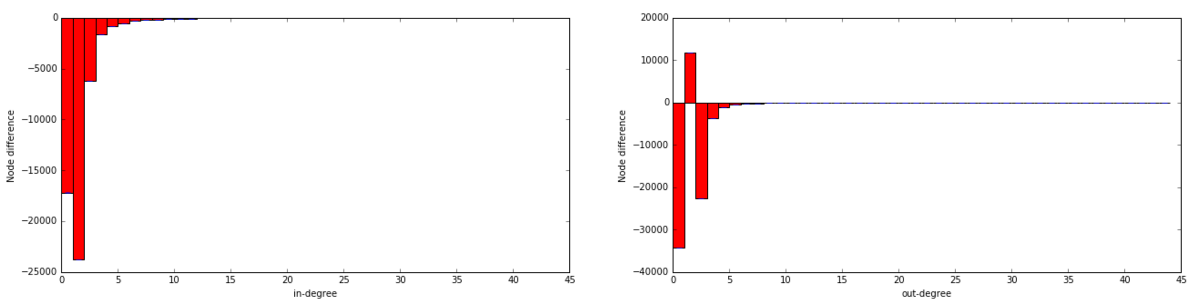
(a) In-degree (b) Out-degree

Figure 28: Difference in degree distributions between Chung-Lu generated graphs (based on the ownership network) and the ownership network.



(a) In-degree (b) Out-degree

Figure 29: Difference in degree distributions between Stub-matching generated graphs (based on the ownership network) and the ownership network.

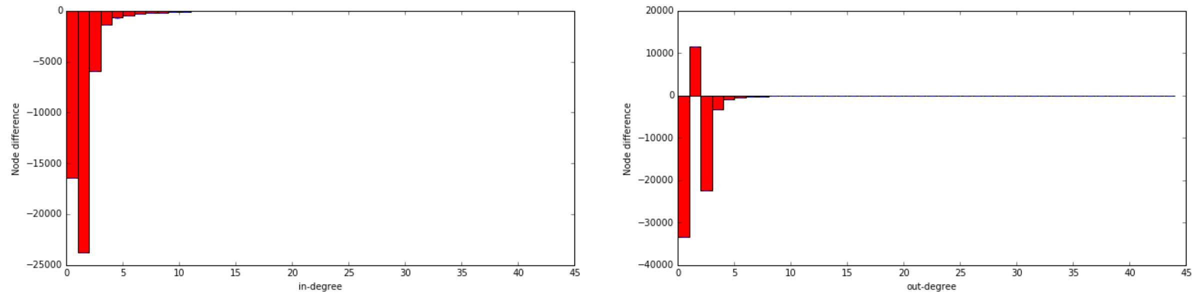


(a) In-degree (b) Out-degree

Figure 30: Difference in degree distributions between Park-Newman generated graphs (based on the interlocks network) and the interlocks network.

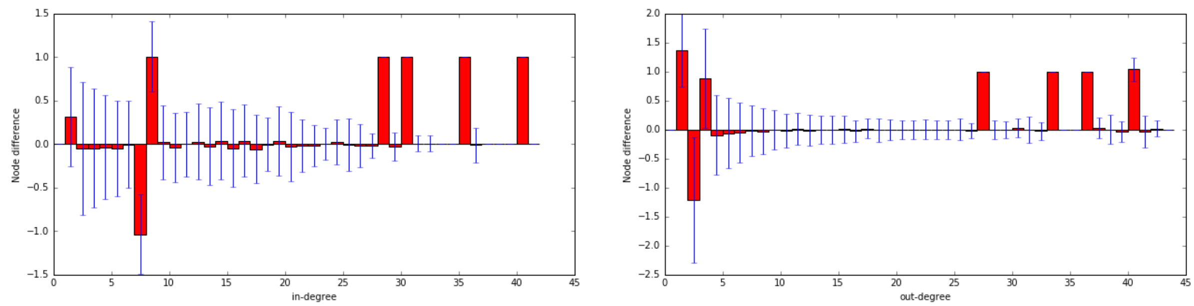
$$R(i,k,G) = |\mathcal{S}_k^i(G)| * \left(\frac{\sum_{G \in \text{Random Graphs}} |\mathcal{S}_k^i(G)|}{|\text{Random Graphs}|} \right)^{-1} \tag{1}$$

When the ratio is larger than 1, the probability of pattern i appearing in a random graph is smaller than the probability of pattern i appearing in the empirical network. To determine which patterns are significant, a cutoff value is used. Only patterns with a ratio larger than a cutoff value are considered significant.



(a) In-degree (b) Out-degree

Figure 31: Difference in degree distributions between Chung-Lu generated graphs (based on the interlocks network) and the interlocks network.



(a) In-degree (b) Out-degree

Figure 32: Difference in degree distributions between Stub-matching generated graphs (based on the interlock network) and the interlocks network.

6.1.2 Estimated Significant Subgraphs

Large networks of the same degree sequence contain similar concentrations of patterns [37, 25]. The concentration of a pattern i of size k is the percentage of pattern i compared to all patterns of size k. With this we can estimate pattern significance based on information of the empirical graph alone. We take the formula for concentration from [37], where the concentration $\mathcal{C}$ of a pattern i of size k is defined as the ratio between its frequency and the sum of all frequencies of patterns of the same size:

$$\mathcal{C}_k^i(\mathcal{G}) = \frac{|\mathcal{S}_k^i(\mathcal{G})|}{\sum_j |\mathcal{S}_k^j(\mathcal{G})|} \tag{2}$$

To determine which patterns are significant, again a cutoff value is used. Only patterns with a concentration larger than a cutoff value are considered significant.

6.1.3 Experts opinion

For the expert's opinion on significant motifs we take textbook examples provided by a study on corporate network structures [38]. This study describes a set of patterns which explain the interaction between different companies. These patterns are:

1. A reciprocal clique: A set of nodes where every node is connected to every other node by a low weight edge.

2. A star: One node connected to many other nodes with high weight edges. The other nodes are not connected with each other.

3. An inverted star: A large group of nodes connected with one node through edges with a low weight. None of the nodes in the large group are connected to each other.

4. A pyramid: A set of nodes with a clear hierarchical structure, where each layer is connected to the layer below by high weight edges.

5. A circle: A set of nodes with a directed path from each node to itself through every other node.

As we do not make use of weights, and therefor we will use these definitions without weight.

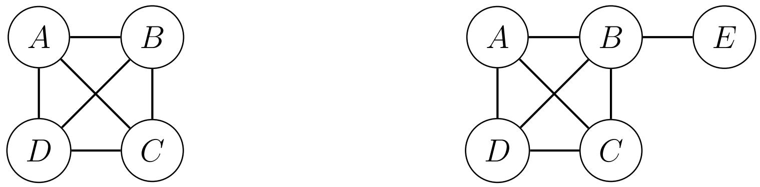
6.2 Comparing Motifs

Our goal is to better understand the information provided by multiplex motifs compared to uniplex motifs. In order to compare uniplex and multiplex motifs we must first define what information we seek, and how to extract it from a set of motifs.

Without looking at the underlying data, we compare motifs based on network structure and properties. Doing so tells us more about what kind of motifs are found. The average *density* of the motifs tells if a network contains structures that are sparse, or if it contains structures that are tightly connected with many edges. Density is defined as the number of relations between nodes in a pattern versus the maximum number of relations possible in a pattern. For this measurement the direction and type of an edge are disregarded. It is only important *if* two nodes are connected, not *how* they are connected. The average density is the average of the density of each motif of the same size.

Using the textbook examples from Section 6.1.3, each motif is categorized into a type of pattern. This tells us if a network has a *preference* for a certain type of pattern, or if it produces motifs that cannot be classified and are more complex. This measurement counts the number of occurrences of textbook example patterns in a set of motifs. As multiplex motifs contain both directed and undirected edges, a multiplex motif will be regarded as undirected when recognizing textbook examples. Many patterns are very similar to textbook example patterns, but are different because they contain a small number of extra edges. To count the textbook examples, all nodes with a degree of one are removed simultaneously from a pattern until a textbook example pattern is recognized, or no more one-degree nodes exist. Figure 33 shows an example of two graphs that are very similar. Figure 33a shows a graph that is a textbook example of a cluster. Figure 33b is almost a cluster, except for the extra node E. For this measurement, both graphs would be counted as a cluster.

The *multiplicity* of motifs is taken into account separately for the multiplex network, as any measure that expresses multiplicity of motifs will result in 0 for uniplex motifs. The multiplicity of a multiplex graph is the division of each edge type. It is calculated by dividing the number of edges of the same type by the total number of edges in the motif. The average multiplicity is expressed as an average over each edge division of all motifs of the same size. Recall that *multiplex* edges are edges that are both interlock and ownership.



(a) Cluster (b) Similar to cluster

Figure 33: Two similarly shaped graphs

When the underlying data is taken into account, we can detect what sort of nodes and edges the motifs contain. For each firm we know their industry sector, number of employees, value of assets, and revenue. By counting the number of occurrences of each company in each motif, we can calculate the relation between the motifs and asset value, employees, or revenue. The *correlation* is expressed both as Pearson correlation and Spearman correlation [11, pp. 177–181]. The Pearson correlation tells us if there is a linear relation between the frequency of a firm in motifs and any of its properties. The Spearman correlation does not check for a linear relation, but rather a *ranking* correlation. A ranking correlation means that if the frequency of firms in motifs decreases or increases, with regards to the other motifs, then so does its property.

Furthermore the division between sectors of a motif becomes apparent by analyzing underlying data. We call this the *company division*. The company division of a single motif is the average over all subgraphs of that motif. The average company division is the average over all motif company divisions. Note that this means that the average company division is an average over averages. This could lead to potentially interesting motifs not becoming apparent right away. A single motif with a company division unlike all other motifs will not change the average company division. We therefore also look at the company division of single motifs.

Just as we can count the occurrence of a certain sector, we can count the number of edges linking different sectors. Each edge can be classified by the companies that it links. This shows if a motif contains information between companies of the same industry, or different industries. An edge between two companies of different industries is called an heterogeneous edge. Likewise an edge between two companies of the same industry is called a homogeneous edge. The percentage of heterogeneous edges is a measurement for motif *complexity*. We say that motifs with a high percentage of heterogeneous edges contain more complex business structures. The complexity of a motif is the average complexity of its subgraphs. The average complexity is the average over all motif complexities. Again, this is an average over averages. Thus we also look at single motifs. For each network we count the number of motifs with a complexity of more than 0.4, and the number of motifs that on average contains no single type of homogeneous edge that takes up more than 60% of all edges. Exploratory experiments show that most companies in a motif are from the same sector and thus have a high share of homogeneous edges from that sector. A cut-off of at least 40% heterogeneous edges captures those motifs that stand out with their complexity. The complexity of the entire network is used as a baseline for complexity.

To understand what all these properties mean, an *overlap* of subgraphs is calculated. Overlap is calculated as the number of subgraphs in a motif from which the nodes also form a subgraph in another network. If there is much overlap in subgraphs between networks, then the motifs are derived from the same nodes from the dataset. Thus the properties provide information on the same structures, viewed from different networks. If there is little overlap, then the motifs from different networks consist of different combinations of nodes. The properties then provide information on different structures originating from the same dataset. A motif M_1 of graph G_1 contains subgraphs $M_1(G_1)$, and each subgraph $g_1 \in M_1(G_1)$ consists of a set of nodes $V(g_1)$. When these nodes form a subgraph for another motif M_2 in another network G_2, we say that motif M_1 of graph G_1 has overlap with graph G_2. Mathematically this is defined as seen in Equation 3.

$$overlap(M_1(G_1), G_2) = \sum_{g_1 \in M_1(G_1)} \sum_{M_2 \in G_2} \sum_{g_2 \in M_2(G_2)} \begin{cases} 1, & \text{if } |V(g_1) \cap V(g_2)| = |V(g_1)| \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

Overlap is computationally a very expensive feature to calculate, as each motif can contain thousands of subgraphs. Except for size 3 motifs, overlap cannot be calculated within our time constraints. Therefore an estimate is used for size 4 motifs. The estimation compares 1% of all subgraphs. This means that for motifs of size 4, 1% of subgraphs is read and each subgraph is compared to 1% of subgraphs of other networks. Overlap is then no longer guaranteed to be detected. Instead the chance of detecting overlap is reduced by a factor 10 000 ($0.01 * 0.01 = 0.0001$). Size 5 motifs contain too many subgraphs for an accurate estimation and thus will not be calculated or estimated.

7 Motif Results

In this section we analyze the patterns found by the motif detection algorithm. First a general overview of all motifs is analyzed. Suitable cut-off values need to found for both ratio and concentration values. A general overview is not sufficient to provide insight in the difference between uniplex and multiplex motifs. Therefore we also take a more in-depth look at the properties discussed in Section 6.2, for each size of motif.

7.1 Overall Results

In Section 6.1 we have discussed three methods to declare a motif significant. We must select which method fits our needs best. Table 34 shows the number of different patterns found for each network. We see that as the possible number of patterns increases, so does the number of encountered patterns. Most of the patterns have a very low ratio and concentration, as seen in Figure 36. This same figure shows that neither ratio nor concentration alone gives a good representation of important patterns. Using only concentration as a measure of significance will cause many *basic* patterns to be seen as motifs. Basic patterns are those patterns which are always likely to frequently appear in any network and are not characteristic for our network. Likewise, using only ratio as a measure of significance will result in some very unique patterns as motifs, that are also not characteristic for the network. These patterns might have a high ratio, but have a very low frequency in the empirical network.

To find those patterns that are the *basic building blocks* of the corporate network, we will need to find those patterns that appear more frequently than in a random network, but also have a high frequency in the empirical network. These patterns are located in the top right corner of Figure 36. For ratio we set the cut-off value to 5, for concentration

	Pattern size			All
	3	**4**	**5**	
Ownership	11	63	391	465
Interlock	2	6	21	29
Multiplex	58	1 132	21 858	23 048

Figure 34: Patterns per network

we set the cut-off value to 0.0001. We also include those patterns which are unique to the empirical data and have a concentration of at least 0.0001. These patterns are shown in Figure 36 with a ratio of 0. Table 35 shows the number of pattens deemed significant by these cut-off values.

	Motif size			All
	3	**4**	**5**	
Ownership	3	4	6	13
Interlock	0	2	10	12
Multiplex	14	48	73	135

Figure 35: Motifs per network

This does not mean that other patterns are not of interest. Patterns with a low concentration but high ratio are for some reason very unique. These patterns might reveal certain business practices, used only by very few companies. Thus these patterns might prove useful for uncovering business structures not used by many companies, such as malicious structures with the intend of tax evasion. Likewise the exact opposite of motifs could also provide valuable information. Patterns which have a very low ratio and concentration, also called *anti-motifs*, indicate structures which are actively avoided by businesses.

We finally look at the complexity of all networks. The definition of complexity for a network is the same as for a subgraph: the percentage of heterogeneous edges in all edges in the network. The complexity of the networks can be seen in Table 37. All networks have identical complexity measures. This is because all networks have an almost identical division of edges when edges are categorized based on which industry sectors they link. If the edge division is seen as a vector, the Euclidean distance between each network is in the order of 10^{-16}. For each network, 74% of all edges link two industrial companies, another 15% of edges link industrial and financial companies, and only 4% of all edges involves a bank. Together with the company division shown in Table 19, this hints that all networks are similar to each other. Motif metrics such as company division and complexity could be influenced by this. If motifs are chosen independently of these properties, then motifs across all networks would have the same values for these metrics.

7.2 Motif Properties

In Section 6.2 several measurements are discussed. Here we report the resulting value per motif size.

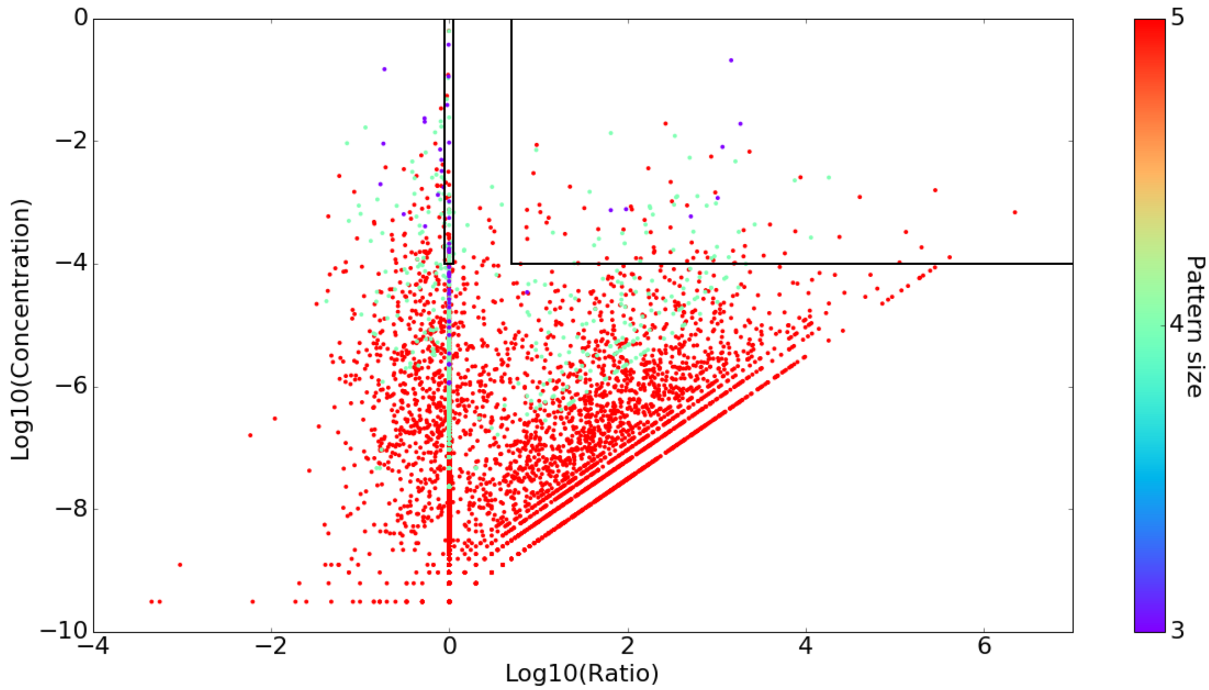


Figure 36: Ratio versus Concentration. Black lines indicate Cut-off values: ratio 5, concentration 0.0001. Non-symmetrical log scale. Patterns that are unique to the empirical data have a ratio of 0.

Network	Complexity (%)
Ownership	19.93
Interlock	19.59
Multiplex	19.88

Figure 37: Network complexity

7.2.1 Size 3

Size 3 is the smallest possible motif size and as such is the fastest to extract from a network. Given the small number of nodes, only very few possible patterns exist. An undirected size 3 pattern only has two variants: a wedge or a triangle. Both of these patterns are relatively easy to create with a bipartite random model and a triangle is even forced into the network when a director works at three companies (Figure 20a). This has caused all size 3 patterns in the interlock network to be deemed insignificant.

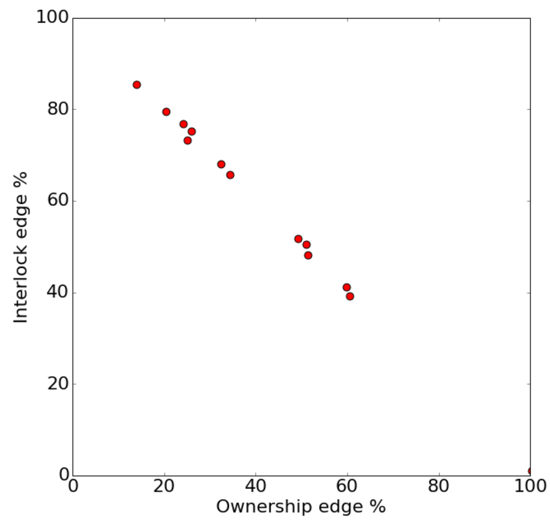
Both ownership and multiplex network do have patterns that are deemed significant. These motifs can be seen in Tables 52, 53 and 54. The ownership network contains a total of 37 724 companies, 7% (2 508) of those companies appear in at least one motif. In effect, the ownership motifs contain only 3.33% of all 75 224 companies in the dataset. For multiplex motifs, this number is much higher. Out of all 75 224 companies, 40% (29 937) appear in at least one motif. The multiplex motifs are also denser than the ownership motifs. The average density of ownership motifs is 0.78. The average density of multiplex motifs is 1.0. Both sets of motifs are very dense, but multiplex motifs score a maximum density. This means that for any multiplex motif, each node has at least one relation with all other nodes.

Although the multiplex network contains much more companies, there is not a lot of overlap with the ownership network. The ownership network has one motif that has 90% overlap with motifs from the multiplex network. This is motif 1 from Table 52, which overlaps with motifs 7, 10, 11 and 12 from Table 53. From the perspective of the multiplex network, each of these motifs has 100% overlap. No other motifs have any overlap. The ownership and multiplex motifs are thus mostly unique sets of subgraphs that provide information on different combinations of companies.

The multiplicity of the multiplex motifs can be seen in Table 38a. On average each motif is a mix of every type of edge. There are only two motifs that contain only one type of edge. Every other motif contains both ownership and interlock edges. For 5 out of 14 motifs, ownership edges appear only in combination with interlock edges, thus forming a multiplex edge. The average multiplicity can be seen in Table 38a. This table shows

Edge type	**Mean**	(σ)
Ownership	27.86%	(29.8%)
Interlock	55.24%	(25.09%)
Multiplex	16.9%	(16.39%)

(a) Multiplicity mean



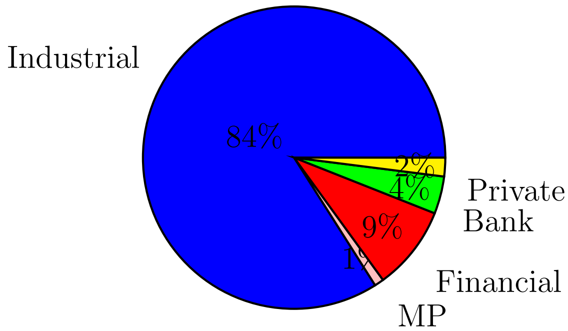
(b) Multiplicity per motif (random jitter of $[-2, 2]\%$ added for visibility)

Figure 38: Size 3: Multiplex motif multiplicity

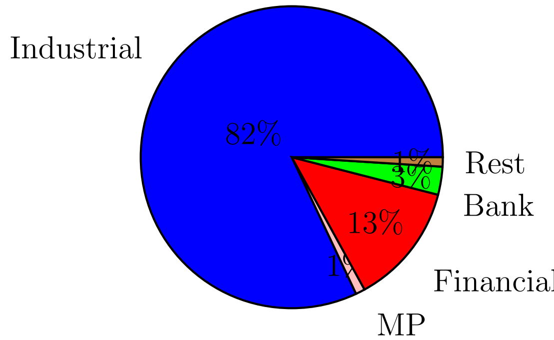
that on average each subgraph contains every type of edge, but that there is a large difference between the subgraphs. Figure 38b shows this difference by plotting the division of ownership versus interlock edges per motif. We can see that, with the exception of the two motifs that consist of only one type of edge, no motif exists for more than 85% of a single edge type. The multiplex motifs are thus focused on patterns that incorporate every type of edge.

The company division in both sets of motifs is almost equal, with the exception of the ownership motifs containing 2% Private Equity firms and 4% less Financial companies. See Figure 39. The average company division of ownership motifs is also very consistent. All motifs contain at least 73% industrial companies and as the percentage of industrial companies rises, the percentage of financial firms drops. Each motif also contains a small percentage of banks. The multiplex motifs show the same trend, but are even more consistent. The standard deviation is at most 4.6% for any sector.

The average complexity of both the ownership and the multiplex network is almost equal. Ownership motifs have an average complexity of 0.202, and multiplex motifs have an average complexity of 0.207. However the multiplex motifs are much more consistent.



(a) Ownership: 3 different patterns



(b) Multiplex: 14 different patterns

Figure 39: Size 3: Company division

No single multiplex motif has more than 40% heterogeneous edges. Multiplex motifs have on average 73% edges between two industrial companies, with only 6% standard deviation. This effectively causes the low complexity of multiplex motifs. The ownership motifs have one motif that can be considered complex. Motif 2 from Table 52 has on average 54% of edges between two industrial companies, while the whole ownership network has 75% of edges between two industrial firms. This motif contains many more edges between financial and industrial companies (23% versus 7%) and between banks and private equity firms (12% versus 0.1%).

Metrics such as the number of textbook examples and the correlation do not show very interesting results. Size 3 patterns can not create many shapes. A circle and a clique are identical, as are a star and a pyramid. A star and pyramid would also be a simple line of three nodes and two edges, a pattern highly unlikely to be deemed significant. The multiplex motif contains 14 (out of 14) motifs that are similar to a clique or circle. The ownership motifs contain no shapes resembling textbook examples. Likewise there is also no significant correlation. Table 40 shows the Pearson and Spearman correlation between employees, revenue and assets and the number of times a company appears in motifs of size 3. Companies that do not appear in any size 3 motif are not taken into account. The correlation values are higher for the ownership network, but not high enough to claim any significant correlation.

	Employees		**Revenue**		**Assets**	
Motifs	**P**	**S** (p)	**P**	**S** (p)	**P**	**S** (p)
Ownership	0.252	0.226 (0.0)	0.362	0.250 (0.0)	0.143	0.349 (0.0)
Multiplex	0.023	0.130 (0.0)	0.026	0.051 (0.0)	0.014	0.107 (0.0)

Figure 40: Size 3: Correlation between appearance in motifs versus several company properties. P = Pearson correlation, S = Spearman correlation with p value.

In general we see that it is difficult to capture interesting information with size 3 motifs as not every network produces motifs, and the multiplex motifs are all similar.

7.2.2 Size 4

With one more node in a subgraph, size 4 motifs offer a much larger variety of possible patterns. Every network contains motifs of size 4, some of which are unique to the empirical data. The motifs can be seen in Tables 55, 56, 57, 58 and 59. Size 4 is also the smallest size where motifs appear in the interlock network. The interlock network contains a total of 61 209 companies, 11% (6 844) of those companies appear in motifs. This is 9% of all companies in the dataset. The ownership motifs contain 15% (5 642) of companies in the ownership network, but this is only 8% of all companies in the dataset. The multiplex motifs contain the most companies, both in percentages and absolute value. A total of 42% (31 938) of all companies appears in multiplex motifs.

Not only do the multiplex motifs contain many more companies, but the companies are also from different sectors. Ownership motifs contain many more financial companies, whereas interlock motifs contain more banks. Both networks contain a small percentage of Mutual & Pension Funds (MP), something that the multiplex motifs do not reflect. Most ownership motifs share the same division of companies: a high percentage of industrial companies (80–95%) and a small percentage of financial companies (2–9%), with the rest of the companies as Mutual & Pension Funds. This is remarkable, as only 0.3% of all companies in the ownership network is a Mutual & Pension Fund. The exception is motif 2 from Table 55. This motif has a much smaller percentage of industrial companies (43%) and a much larger percentage of financial firms (56%). The company division for interlock motifs is mostly identical. The largest difference is that one motif has 15% financial companies and only 2% banks, while the other motif has only 8% financial companies and 8% banks. This division corresponds with the division of the entire interlock network.

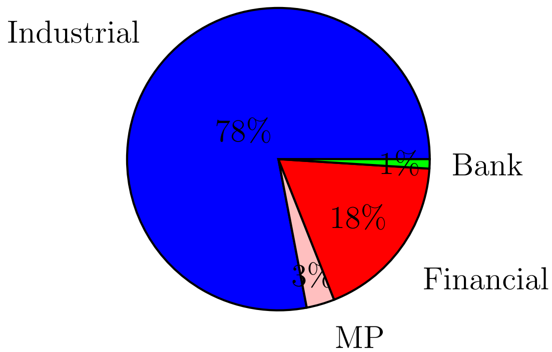
Multiplex motifs have a much more diverse share of industrial companies. The average percentage of industrial companies is 76%, but with a standard deviation of 14%. This average is lower than the average of the entire multiplex network. The financial sector makes up for this difference. For any multiplex motif it holds that industrial and financial companies make up at least 85% of the nodes. The rest of the companies are banks. Thus there is a split in the motifs: those motifs with a high industrial share and low financial share of firms, and those with a low industrial share and high financial share of firms.

As this is the smallest size where interlock motifs are encountered, we can now plot for each company the number of times it appears in interlock motifs and ownership motifs. The plot can be seen in Figure 42. We can see that there is no clear preference of certain sectors for a specific network.

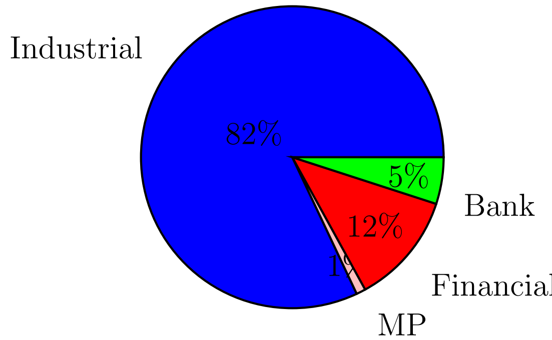
Contrary to size 3 motifs, it is estimated that the interlock and ownership motifs have very little overlap with each other, but that both do have overlap with the multiplex network. Only ownership motif 4 from Table 55 is estimated to have no overlap. When we compare this motif to all multiplex motifs in Tables 57, 58, 59, and 60, there is indeed no pattern that could contain motif 4. The other ownership motifs are likely to overlap almost completely, based on the counted number of overlapping subgraphs and the correction factor of 10 000. The multiplex motifs thus contain information about almost all companies in the ownership motifs.

The interlock motifs have two different patterns. Motif 1 from Table 56 is likely to overlap completely with multiplex motif 7 from Table 57, as this is the same pattern. The other interlock motif has no overlap, as no pattern from the multiplex network can contain this motif. Because interlock motif 1 contains the most subgraphs, indicated by its concentration value in Table 56, most of the information in the interlock motifs is also present in the multiplex motifs.

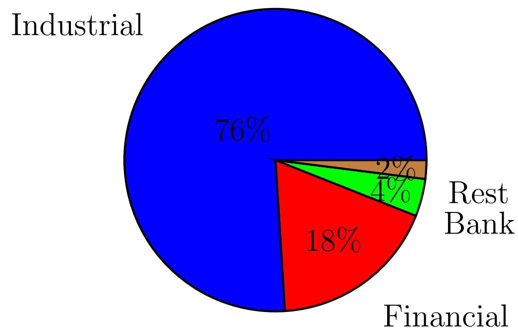
Despite the large overlap, there are considerable differences between the motif sets. Not only is the company division different, the complexity also shows significant contrast



(a) Ownership: 4 different patterns



(b) Interlock: 2 different patterns, with 9.10% of all companies



(c) Multiplex: 48 different patterns

Figure 41: Size 4: Company division

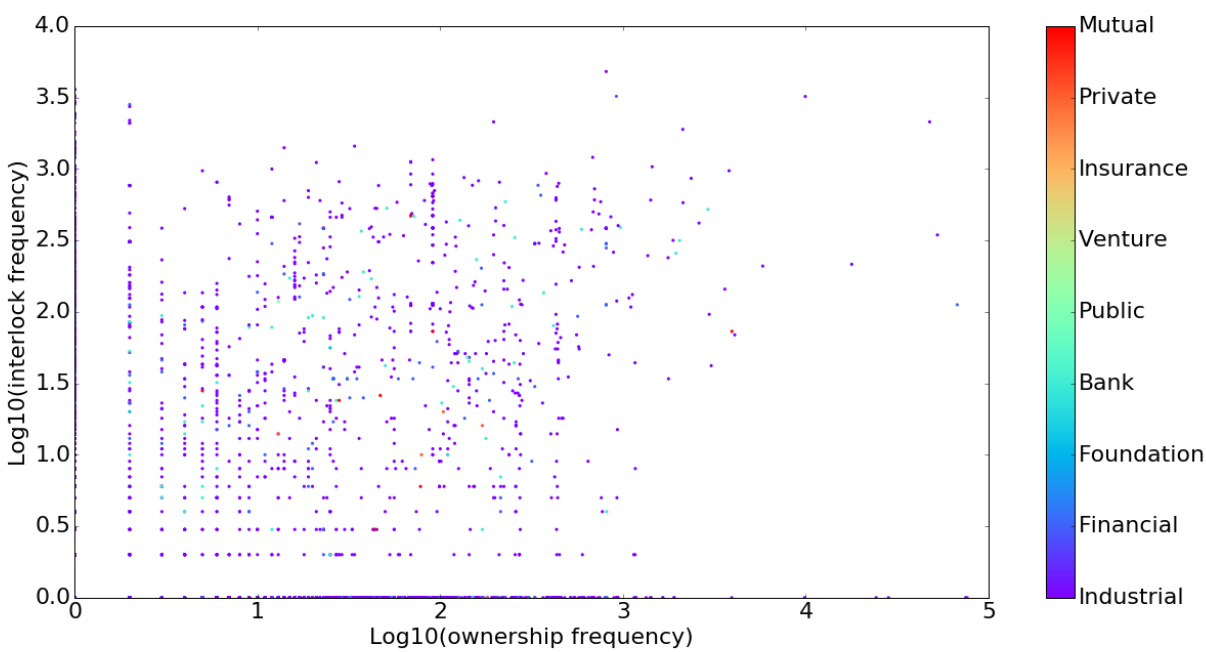


Figure 42: Frequency of firms in size 4 motifs from ownership and interlock networks, that appear in at least one motif. Of these companies 29.91% appears only in the ownership networks, and 48.82% of these companies only appear in the interlock networks.

between the motif sets. Of the two interlock motifs, none is considered to be complex.

Both patterns have 70% of all edges between two industrial companies. However in the ownership motif set there is one motif that is considered complex. Three out of the four ownership motifs have a high percentage of edges between two industrial companies (70–85%). The exception is motif 2 from Table 55, which has on average only 10% of edges between two industrial companies. This motif is much more focused on financial companies, with 65% of edges between a financial firm and an industrial company, and 23% of edges between two financial firms.

The multiplex motifs have a wider distribution of complexity, as the standard deviation of the average complexity is 0.123. Out of all 48 motifs, 14 contain less than 60% of edges between two industrial companies. Out of those 14, only 7 are considered complex. These 7 motifs are more focused on financial firms, as on average for these 7 firms 47% of edges is between a financial firm and an industrial company and 16% is between two financial firms. Two motifs stand out, as they contain many more edges between sectors other than industry and financial. Motif 4 in Table 57 has 30% of edges involving banks, and motif 40 in Table 60 has 15% of edges involving mutual and pension funds.

This shows that the multiplex motif set has several motifs with a remarkably high share of smaller sectors, and several motifs that stipulate the relation between the two largest sectors: industry and finance. The later is also achieved by ownership motifs, but this motif set does not involve any of the smaller sectors.

This difference is not reflected by the average complexity measure. On average the complexity of each set of motifs is very similar. Ownership motifs have an average complexity of 0.277, interlock motifs an average of 0.246, and multiplex motifs an average of 0.263. These values are 0.05 to 0.07 above their respective network complexity shown in Table 37.

The multiplicity of size 4 multiplex motifs can be seen in Table 43a. The size 4 multiplex motifs have on average less multiplex edges than the size 3 motifs, meaning that a smaller share of companies are related by both ownership and interlock. There is a preference for interlock edges, as 73% of motifs contains at least 60% interlock edges. This is not surprising, as there are many more interlock edges than there are ownership edges in the multiplex network.

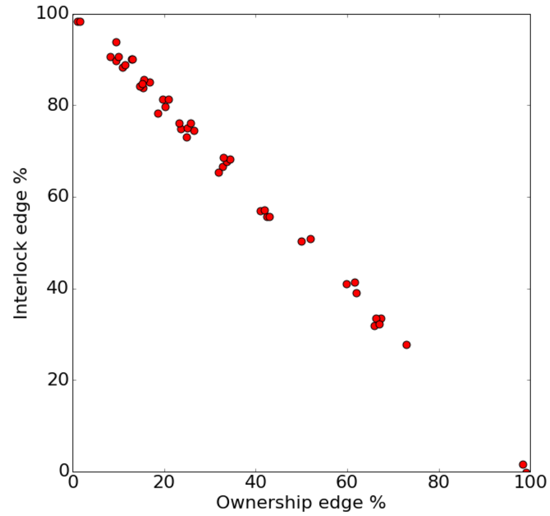
Even though the multiplex motifs contain many more motifs, with a different company division and complexity, some measures fail to express this. The average density of each set of motifs is almost equal. Ownership motifs have a density of 0.71, interlock motifs have a density of 0.75, and multiplex motifs have a density of 0.73. As can be seen in Tables 55, 56, 57, 58, 59 and 60, there is only one motif with a density of below 0.6 (multiplex motif 4, Table 57).

The number of textbook examples also shows little difference between each set. Just as with size 3 motifs, no ownership motif is recognized as a textbook example. Off all 48 multiplex motifs, 38 are similar to the textbook examples. Of those 38, 30 are size 3 cliques with one extra node. These shapes are of little interest, as we have already seen that multiplex motifs contain a lot of *triangles*. The interlock motifs have one motif that is a circle, but as there are only two motifs, we cannot claim that interlock motifs form more textbook example shapes than other motif sets do.

Correlation does not show a large difference between the sets. Table 44 shows the correlation of size 4 motifs. In contrast to the correlation of size 3 motifs, the frequency of companies in multiplex motifs now has the highest correlation with every property. However, no value is high enough to consider any of the aspects correlated.

Edge type	Mean	(σ)
Ownership	27.44%	(29.03%)
Interlock	62.7%	(27.59%)
Multiplex	9.86%	(9.81%)

(a) Multiplicity mean



(b) Multiplicity per motif (random jitter of $[-2, 2]\%$ added for visibility)

Figure 43: Size 4: Multiplex motif multiplicity

Motifs	Employees P	Employees S (p)	Revenue P	Revenue S (p)	Assets P	Assets S (p)
Ownership	0.195	0.051 (0.0)	0.151	-0.025 (0.1)	0.046	-0.037 (0.0)
Interlock	0.076	0.095 (0.0)	0.066	0.019 (0.1)	0.036	0.080 (0.0)
Multiplex	0.211	0.194 (0.0)	0.186	0.104 (0.0)	0.156	0.172 (0.0)

Figure 44: Size 4: Correlation between appearance in motifs versus several company properties. P = Pearson correlation, S = Spearman correlation with p value.

7.2.3 Size 5

This is the largest size motif that can be extracted within a time-span fitting in the scope of this study. Size 5 motifs contain the largest number of different motifs for any network, and they span over the most companies. There are 6 ownership motifs that contain 16% (6 086) of companies of the ownership network. This is 8% of all companies in the dataset. The interlock motifs have surpassed the ownership motifs in numbers, and contain 10 different motifs. This is remarkable as the number of interlock motifs for smaller sizes is extremely small. These 10 motifs provide information on 16% (61 209) of all companies in the interlock network. This is 13% of all companies in the dataset. Just as with the smaller size motifs, the multiplex motif set is the largest set. The 73 motifs cover 35% (26 654) of all companies.

Due to the vast amount of subgraphs, no overlap could be calculated or estimated. However, looking at the the overlap of size 3 and size 4 motifs, we can expect that the ownership and interlock motifs have very little overlap. We can also expect that motifs from either the ownership or interlock network have much overlap with those multiplex motifs that contain the same pattern, as we have also seen with size 3 and 4 motifs. Furthermore, we can state that multiplex motifs that are identical to ownership or interlock motifs, must have 100% overlap with those motifs. The reverse is not necessarily true, an ownership or interlock motif that is identical to a multiplex motif might not have 100% overlap. This is because when multiple edge types are taken into account, the original subgraphs of these ownership or interlock motifs might form different multiplex subgraphs.

Keeping this in mind, it is reasonable to assume that ownership motif 1 (Table 61) has much overlap with multiplex motifs 5 (Table 64), 34 (Table 68), and 54 (Table 71). Ownership motif 2 (Table 61) has overlap with multiplex motif 10 (Table 65) and 17 (Table 66). Motif 3 (Table 61) has overlap with multiplex motifs 13 (Table 65), 14 (Table 65), 46 (Table 70) and 52 (Table 71). Motif 4 (Table 61) has overlap with multiplex motifs 28 (Table 67) and 35 (Table 68). Finally ownership motif 5 (Table 61) has overlap with multiplex motif 61 (Table 72). No multiplex motif matches ownership motif 6 (Table 61), thus ownership motif 6 has no overlap.

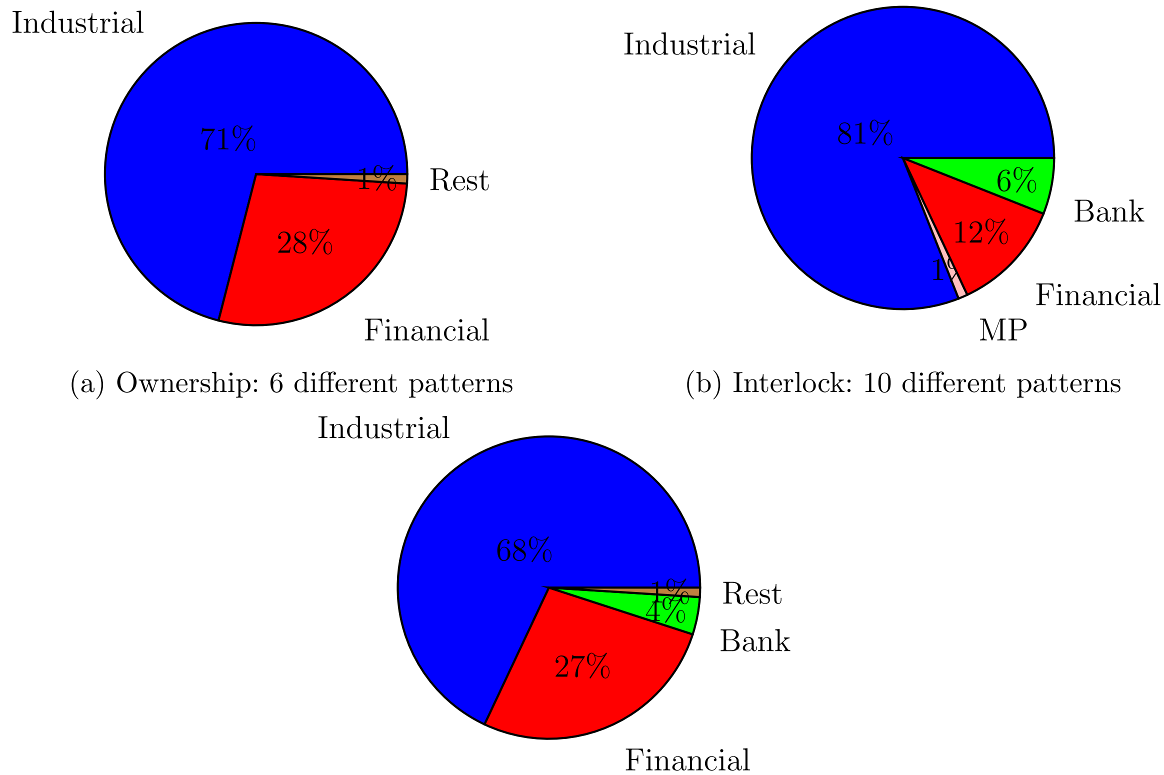
Similarly, the overlap between interlock and multiplex motifs can be determined. Interlock motif 1 (Table 62) is identical to multiplex motif 44 (Table 70), motif 2 (Table 62) is equal to multiplex motif 42 (Table 69), motif 3 (Table 62) is the same as multiplex motif 67 (Table 73), and motif 4 (Table 62) is equal in pattern to multiplex motif 57 (Table 72). The other six interlock motifs have no overlap with multiplex motifs.

The multiplex network contains patterns that exist solely out of interlock edges. These multiplex motifs are motifs 9, 19, 23, 24, 25 and 39 (Table 64 to 69). These patterns are also present in the interlock network, but are not regarded a motif there. This means that these patterns are easily created in a random interlock network, but when ownership data is taken into consideration most of the randomly created subgraphs have at least one ownership edge. A subgraph consisting of only interlock edges becomes much rarer, and is thus considered a motif. The same effect is not observed for multiplex motifs with only ownership edges. Any multiplex motif with only ownership edges has an identical counterpart in the ownership motifs.

Given that it is likely that such a large share of the motifs overlap, it is noteworthy to mention that the company division of the different motif sets does not look alike. The motifs from the ownership network contain mostly industrial and financial companies, as can be seen in Figure 45. They have a very large share of financial companies compared to smaller motifs. This large share of financial companies can also be seen in the multiplex network. The interlock motifs have an average company division with many more industrial companies.

Not surprisingly, the complexity is likewise very different. The ownership motifs have an average complexity of 0.270, very similar to their size 4 counterpart. Out of the 6 ownership motifs, two are complex: motif 1 and 3 from Table 61. Both these motifs have a very low share of edges between two industrial companies: 7% and 10% respectively. Instead these two motifs have a much larger share of edges between financial and industrial companies, and between two financial companies.

Interlock motifs also have similar complexity to that of their size 4 counterparts. With a 0.262 complexity measure, the interlock motifs are on average just as complex as ownership motifs. However the interlock motifs are much more consistent in complexity. No single pattern stands out with a higher or lower percentage of a certain edge. All motifs have between 63% and 75% of edges between industrial companies, and between 10% and 24% of edges between a financial firm and an industrial firm. Thus none of the interlock motifs is considered complex. Even though interlock motifs have a relatively large share of banks (6%), presence of banks in motifs is small. There is one motif that contains on average 20% of edges that involve a bank, be it a homogeneous edge between two banks, or an



(a) Ownership: 6 different patterns

(b) Interlock: 10 different patterns

(c) Multiplex: 73 different patterns

Figure 45: Size 5: Company Division

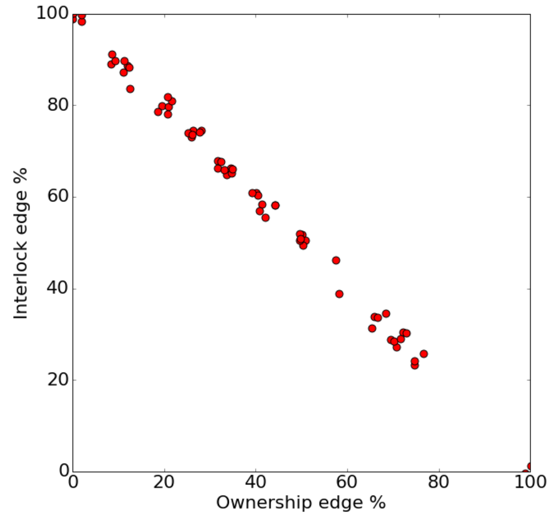
edge between a bank and another sector. This is motif 10 from Table 63.

The multiplex motifs have an average complexity of 0.322. This is the highest average complexity of any motif set of any size. Of the 73 motifs, 22 contain no more than 60% of edges between companies of the same sector and are thus complex. For 37 companies it holds that no single homogeneous edge type makes up 60% of all edges. This suggests that multiplex motifs have a large variety in complexity. Closer inspection of how edges are divided reveals that many motifs have very different divisions. In interlock or ownership motifs edges between two industrial firms often make up either a very large part or a very small part of a motif. This is not the case for multiplex motifs. On average only 54% of edges is between two industrial companies, and there is a standard deviation of 26%. Despite having a smaller share of banks than the interlock motifs, there are more motifs which a large share of edges involving a bank. Motifs 2 and 6 (Table 64), 16 (Table 66), 22 (Table 67), 43 (Table 70) and 58 (Table 72), all have at least 30% of edges involving a bank. Motif 22 stands out with 41% of edges involving a bank. Furthermore motif 70 (Table 73) has 14% of edges involving mutual and pension funds. This is remarkably high compared to the entire multiplex network, which has 0.8% of edges involving mutual and pension funds.

Size 5 motifs have the smallest percentage of edges that are both interlock and ownership, compared to smaller sized multiplex motifs. Instead, they have the largest share of ownership edges of all measured sizes. See Table 46a. These motifs have less preference for interlock edges, compared to size 4 motifs. Of all motifs, 56% have at least 60% interlock edges.

Edge type	Mean	(σ)
Ownership	35.29%	(28.82%)
Interlock	58.32%	(30.09%)
Multiplex	6.4%	(8.06%)

(a) Multiplicity mean



(b) Multiplicity per motif (random jitter of $[-2, 2]\%$ added for visibility)

Figure 46: Size 5: Multiplex motif multiplicity

Just as with size 4 motifs, no industry sector has a preference for either ownership or interlock motifs. This can be seen in Figure 47.

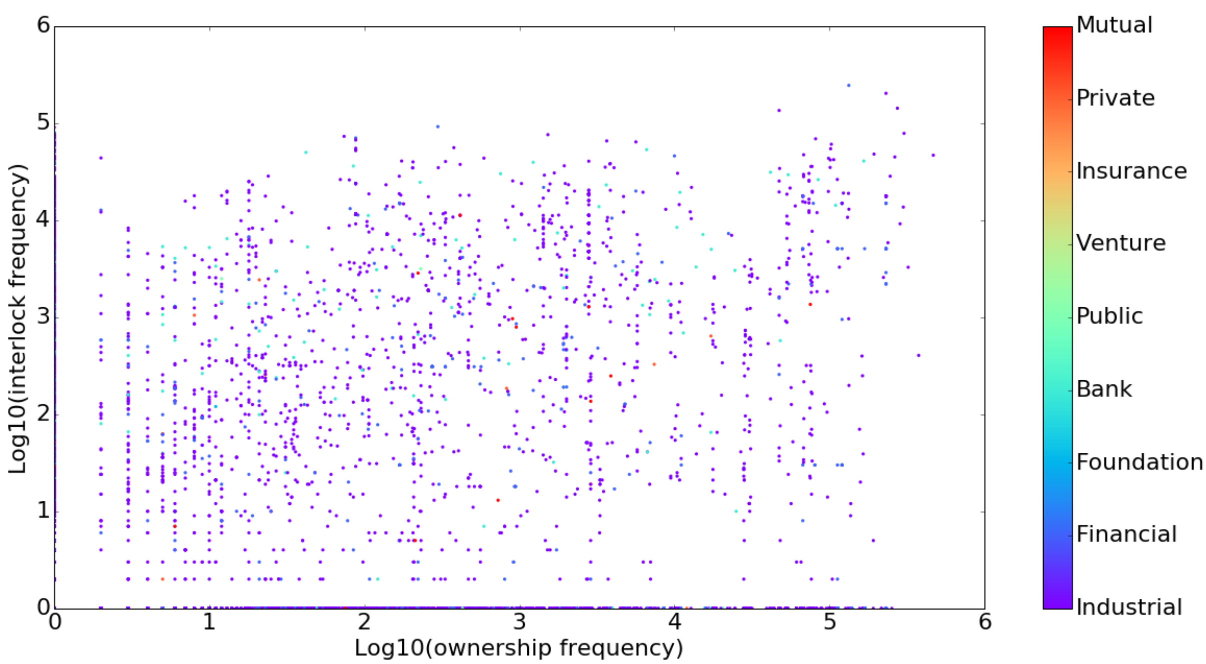


Figure 47: Frequency of firms in size 5 motifs from ownership and interlock networks, that appear in at least one motif. Of these companies 25.63% appear only in the ownership networks, and 56.92% of these companies appear only in the interlock networks.

The other metrics, density, textbook examples and correlation, provide little information. The average motif density for ownership and multiplex networks is almost equal with 0.57 and 0.55 respectively. The density of interlock motifs is higher with 0.66. In respect to the size 3 and 4 motifs, the density of size 5 motifs is much lower, but it does not reveal the difference between motif sets from different networks.

Likewise, and just as with smaller motif sizes, the number of textbook examples expresses little information. Again no ownership motif is recognized as a textbook example. The interlock network contains two motifs (motif 5 Table 62, motif 10 Table 63) that fit the description of a circle. The multiplex motifs have, just as with size 4 multiplex motifs, a large number of patterns that contain 3 nodes that are all connected, with two one-degree nodes extra. These motifs are recognized both as circle and clique. This situation appears 38 times. Apart from these 38 motifs, there are 6 motifs classified as a pyramid, 4 as a circle, and 7 as a clique. There are more different shapes than in smaller motifs, but still a very small number compared to the total 73 different motifs.

Table 48 shows the correlation of size 5 motifs. Just as with size 3 and 4 motifs, no correlation is found between the frequency at which companies appear in motifs, and their attributes.

	Employees		**Revenue**		**Assets**	
Motifs	**P**	**S** (p)	**P**	**S** (p)	**P**	**S** (p)
Ownership	0.166	0.070 (0.0)	0.067	-0.027 (0.0)	0.015	-0.032 (0.0)
Interlock	0.140	0.156 (0.0)	0.151	0.084 (0.1)	0.099	0.170 (0.0)
Multiplex	0.222	0.184 (0.0)	0.148	0.107 (0.0)	0.143	0.170 (0.0)

Figure 48: Size 5: Correlation between appearance in motifs versus several company properties. P = Pearson correlation, S = Spearman correlation with p value.

8 Conclusion and Discussion

In this paper we study network motifs, the basic building block of a network, with the aim to understand the difference between uniplex and multiplex networks. We base our motifs on induced subgraphs, for which we extend the definition to include multiplex networks.

As empirical data we use a corporate dataset with information on companies and their board members. We split the dataset into three different networks: the ownership network with directed edges showing which companies own other companies, the interlock network with undirected edges showing which companies share a board member, and the multiplex network that contains both ownership and interlock edges.

The first step in motif recognition is subgraph enumeration. Subgraph enumeration lists all patterns in a graph, together with the number of times they appear. For this task we use Subenum, an algorithm developed to enumerate induced subgraphs in unweighted directed graphs. To extend subgraph enumeration to multiplex networks, we transform the multiplex network into a weighted graph and extend the algorithm so that it can handle weights.

The second step is to determine which patterns are significant. We take a combination of two measures to determine which patterns are motifs. The first measure is the *ratio* between the frequency of a pattern in a network versus the average frequency of the pattern in a set of random graphs. The random graphs are generated with a stub-matching model. As the interlock network is inherently a bipartite network between board members and companies, this network is also modeled as bipartite. Only those patterns with a ratio of at least 5 can be significant. The second significance measure is the *concentration* of a pattern. Only patterns that make up least 0.01% of all subgraphs, of the same size, can be considered significant. The resulting motifs are compared to textbook examples.

The third step is defining which properties of the motifs must be compared. We compare density, multiplicity, company division, correlation and complexity. To further support the comparison we measure overlap between motif sets of different networks. We extract three sizes of motifs: size 3, 4 and 5. For each size the properties of the motifs of each network are compared.

8.1 Conclusion

In Section 7 we show the properties of each network's motifs, for three different sizes. In this section we discuss results for each size and end with an answer to the research questions posed in Section 1.

There are many different motifs found throughout all networks. We briefly review one motif as an example. The motif is shown in Figure 49. This motif, of size 4, is found in the multiplex network, and depicts a business structure where three companies all share board members. One of the companies owns a share in a company with which it shares a board member, and it downs a share of another company that is not related to the others. This motif appears 1 600 times more often in the empirical data than it does in the random graph ensemble. It also makes up 0.5% of all size 4 subgraphs in the multiplex network.

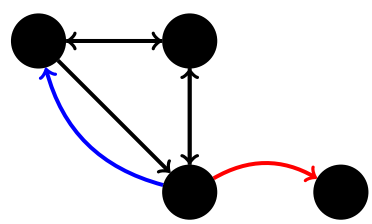


Figure 49: Size 4 multiplex motif 6 from Table 57.

In size 3 motifs we see that there is very little overlap between ownership and multiplex motifs. Thus both sets provide information on different combinations of firms. This is not reflected by the average statistics of both motifs. Density, company division and complexity are all similar, and neither set of motifs appears related to any of the companies' properties. Yet there is one clear distinction between the two sets. The ownership motifs have a greater variety in company division and complexity. There is one ownership motif that contains on average more financial companies and more edges involving a financial company. This same motif also involves much more private equity firms. This is a property that the multiplex motifs do not show. Multiplex motifs however contain many more companies. Where ownership provides information on 7% of the companies of an ownership network, multiplex motifs provide information on 40% of companies in its network. Size 3 multiplex motifs are certainly different from the ownership motifs, but with the given metrics do not provide a clear view of what information they hold.

Size 4 motifs provide a better overview of the difference between uniplex and multiplex networks, as it was possible to extract motifs from both the uniplex networks and the multiplex network. The ownership and interlock motifs contain much less companies than the multiplex motifs. The overlap between ownership and interlock networks is estimated to be negligible, but the overlap between ownership and multiplex is estimated to be large. All except one ownership motif have high overlap. Most of the subgraphs present in the ownership motifs are thus also present in the multiplex motifs. The same holds for the interlock motifs, where the motif with the most subgraphs is estimated to have high overlap with the multiplex motifs. This means that multiplex motifs provide for a large part information on the same combinations of nodes.

Just as with size 3 motifs, we see that size 4 ownership motifs have one pattern that deviates from the average. Ownership motif 2 (Table 55) has many more financial companies, and thus also many more edges involving a financial firm. We can see the same division in several multiplex motifs. There are multiplex motifs with very different company and edge divisions than that the average would indicate. Size 4 multiplex motifs are able to better indicate which motifs are characteristic for certain industry sectors.

The motifs of size 5 show properties similar to size 4 motifs. The multiplex motifs again contain the most companies, and likely overlap with with five out of six ownership motifs and four out of ten interlock motifs. We can only assume that the overlap between both uniplex motif sets is small. The company division and complexity do tell us more about the difference between the motif sets. The motifs from the interlock network are all similar, but motifs from the ownership network have many more motifs involving industrial and financial companies. Just as with size 4 ownership motifs, the size 5 ownership motifs can distinguish motifs that are focused on the financial sector. The multiplex motifs extend upon this, and are able to detect motifs which are more focused on not only financial firms but also banks or mutual and pension funds.

So to answer the first part of our research questions: do multiplex motifs provide a better understanding of the basic building blocks of a corporate network compared to uniplex motifs? Size 4 and 5 multiplex motifs certainly provide different information compared to uniplex motifs. They are able to detect different structures that can be attributed to different industrial sectors. This provides a better overview of a corporate network than ownership or interlock networks show. Ownership is able to find motifs for the two largest sectors, and interlock motifs do not provide any motif characteristic for any sector. Size 3 motifs differ from these results. Size 3 appears to be too small for multiplex motifs to distinguish between different industry sectors.

These results are made apparent when looking at the properties company division and complexity, of individual motifs. These metrics show that different motifs contain different firms from distinct sectors, and that multiplex motifs distinguish the most interesting motifs. However the average values of these properties provide little information. The difference in average complexity is almost equal for any set of motifs of the same size, and the overall company division shows only small differences for each motif set.

Overlap and multiplicity support this result by showing that multiplex motifs contain many of the same companies that the uniplex motif sets contain. For all sizes we also see that multiplicity is high. Only very few multiplex motifs consist of only one type of edge. Multiplex motifs are thus a new set of motifs and not just a union of both uniplex sets. However, overlap shows that many of the same company combinations from uniplex motifs can also be found in the multiplex motifs. The multiplex motif set is also much larger than the union of the two uniplex motif sets. Thus multiplex motifs not only provide a new view on the same set of companies, but also on a larger set of companies.

As for the second part of the question: does this indicate that in general multiplex motifs provide a better view of a network compared to uniplex motifs? We cannot answer this question. The difference in motifs only becomes apparent after taking into account the underlying data. We can not state that for every network it holds that multiplex motifs provide a better, or even different, view of the data.

The metrics that would have shown this are density and the number of textbook examples. Two motif sets with largely different densities indicate a clear preference for different structures in their respective networks. The number of textbook examples would digress on such a result by specifying which structures there are. However, the density of motifs and the number of textbook examples in motifs did not provide any useful insights on the network. The density of every motif set of equal size is very similar. Furthermore, for all sizes of motifs, and every motif set, the number of textbook examples is low. Most textbook examples that are found are both a clique and circle, meaning that the pattern contains an undirected triangle. The other textbook examples are not encountered or only very rarely. From this we must conclude that the textbook examples are not a good indicator of significant motifs, either because the textbook is dated and companies now behave differently and form different patterns, or because the provided examples are not statistically significant but are significant according to a different measure. From these results we must conclude that network properties extracted without knowledge of underlying data, such as density and textbook examples, do not provide a good overview of the characteristics of different motif sets.

Lastly we discuss the correlation. No motif set has shown a significant relation with any of a company's properties. Though motifs certainly express interesting business structures, they do not do so in regards to their number of employees, assets or revenue. This is surprising, as large companies are surely able to construct many business structures, which would result in a high frequency in a motif set.

8.2 Possible limitations of motifs

During this study on multiplex motifs, we have discovered several pitfalls and difficulties with motifs and how to compare them. The first is the selection of motifs. There are many different types of motifs, and selecting the right one for a particular study is more of an art than a science. It is hard to reason which motif type would contain the information that is needed for a specific research.

Second is the time required to enumerate subgraphs. Our results show that as size increases, different motifs appear. It is interesting to see how this trend progresses with larger size motifs, but due to the time required to extract larger size motifs this is not an option. Computation time might be reduced with the use of different algorithms, but these algorithms might not enumerate all desired subgraphs.

Third is the comparison of motifs. Comparing two uniplex motifs is possible with generic network measures. But as the size of a motif is small, many measures will be nearly identical and with multiplex motifs it becomes impossible to use generic measures. Thus comparison must then rely on underlying data of the network. As a motif is a collection of subgraphs, the underlying data can only be expressed as an average of all these subgraphs. Comparison between two motifs is thus a comparison of averages over a set of subgraphs. However, to compare entire sets of motifs, the comparison becomes a comparison between an average over an average, which complicates the interpretation of the result.

8.3 Future Work

For future studies, we first recommend to expand upon certain aspects of this study. This study focuses on unweighted multiplex subgraphs. It would be interesting to see the effect of weights on motifs. Different weights indicate different corporate structures. A structure with high weight ownership edges indicates complete control, while many low weight ownership edges show stock diversity. Weight can also be used to extract network properties that do not rely on the underlying data. Think for example of the average weight per edge type. This could show if a motif is used to tightly connect nodes via the respective edge type. Expanding our proposed algorithm to include weighted graphs could elaborate on the differences between motif sets by providing a more fine-grained motif selection.

Furthermore, the current algorithm enumerates induced subgraphs. These subgraphs have shown to be useful for detecting motifs related to specific industry sectors. Using a different type of subgraph can provide information on other aspects of a network.

The selection of the random graph model used to generate random graphs is also of great importance. A model that is completely random will generate graphs with little similarity to a real-world network. A model too restrictive will generate graphs too similar to a real-world network. Neither of these models is fit for motif recognition. A good middle-ground must be found: one that is sufficiently random to indicate which structures are the result of randomness and which structures are the result of human intervention, yet restrictive enough to filter out generic human behavior in the given setting of the real-world data. In this study we have used the stub-matching model, as it is sufficiently random with restrictions on the number of edges, and keeps in mind the behavior of companies. However, we have not solved the model-selection problem for the general case. To select a model for a network where no underlying data is known, such as a completely abstract network, the techniques describes in [18] may be used to compare the generated random graph with the empirical network.

Bipartite modeling will remain a difficult topic, regardless of model. Creating a bipartite model such that the one-mode projection will be similar to the one-mode projection of an empirical bipartite network is no trivial task. Such a model requires more rules to follow than only a degree-sequence, but not so much rules that the model would no longer be random.

The random models are used to calculate significance of a pattern, but significance is also estimated without the need for random models. In this study we use the concentration of a pattern. As it turns out this measure alone is not enough to indicate significance. A better estimate of importance might indicate significance better and even remove the need for random models. The methods discussed in [6] and [7] could help to estimate the number of times a pattern appears in a graph, and thus can indicate significance without random graph models.

A final point to expand upon this study is an improvement on measuring overlap. Size 4 and 5 motifs contain too many subgraphs to do an all versus all comparison. Overlap is basically the same as Jaccard similarity, and as such a min-hashing technique could provide a better estimate of overlap than the method used in this paper. Furthermore, overlap between a uniplex and multiplex motif set can be calculated faster by first comparing the pattern's adjacency matrices. The uniplex pattern's matrix must be contained in the multiplex matrix, otherwise no overlap can exist. This reduces the number of patterns to compare.

Besides these points that would expand this study, we also suggest a new study. This research has not answered the question whether multiplex motifs provide a better understanding of networks compared to uniplex motifs in the general case. In order to study this, the results of this study should be compared to similar studies on different datasets. How do the results compare to corporate data from different countries, or from different networks types such as a social network?

References

[1] CorpNet. Uncovering Networks of Corporate Control. `http://corpnet.uva.nl/`. Accessed: 2016-09-19.

[2] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74:47–97, 2002.

[3] Federico Battiston, Jacopo Iacovacci, Vincenzo Nicosia, Ginestra Bianconi, and Vito Latora. Emergence of multiplex communities in collaboration networks. *PLOS ONE*, 11(1):1–15, 2016.

[4] Stefano Battiston, Eric Bonabeau, and Gérard Weisbuch. Decision making dynamics in corporate boards. *Physica A: Statistical Mechanics and its Applications*, 322:567–582, 2003.

[5] Stefano Battiston, J. Doyne Farmer, Andreas Flache, Diego Garlaschelli, Andrew G. Haldane, Hans Heesterbeek, Cars Hommes, Carlo Jaeger, Robert May, and Marten Scheffer. Complexity theory and financial regulation. *Science*, 351(6275):818–819, 2016.

[6] Edward A. Bender. The asymptotic number of non-negative integer matrices with given row and column sums. *Discrete Mathematics*, 10(2):217–223, 1974.

[7] Edward A. Bender and E. Rodney Canfield. The asymptotic number of labeled graphs with given degree sequences. *Journal of Combinatorial Theory, Series A*, 24(3):296–307, 1978.

[8] Fan Chung and Linyuan Lu. Connected components in random graphs with given expected degree sequences. *Annals of Combinatorics*, 6(2):125–145, 2002.

[9] Daniel M. Dunlavy, Tamara G. Kolda, and W. Philip Kegelmeyer. Multilinear algebra for analyzing data with multiple linkages. In *Graph Algorithms in the Language of Linear Algebra*, pages 85–114. SIAM, 2010.

[10] Paul Erdős and Alfréd Rényi. On random graphs. I. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[11] Andy Field. *Discovering Statistics Using SPSS*. SAGE, 3rd edition, 2009.

[12] Linton C. Freeman. The development of social network analysis-with an emphasis on recent events. *In The SAGE Handbook of Social Network Analysis*, pages 26–39, 2011.

[13] Shayan Ghazizadeh and Sudarshan S. Chawathe. SEuS: Structure Extraction Using Summaries. In *Proceedings of Discovery Science*, pages 71–85, 2002.

[14] Hu Haiyan, Yan Xifeng, Huang Yu, Han Jiawei, and Zhou Xianghong Jasmine. Mining coherent dense subgraphs across massive biological networks for functional discovery. *Bioinformatics*, 21(1):i213–i221, 2005.

[15] Lawrence B. Holder, Diane J. Cook, and Surnjani Djoko. Substructure discovery in the subdue system. In *Proceedings of the AAAI Workshop on Knowledge Discovery in Databases*, pages 169–180, 1994.

[16] Emőke-Ágnes Horvát and Katharina Anna Zweig. One-mode projection of multiplex bipartite graphs. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*, pages 599–606, 2012.

[17] Emőke-Ágnes Horvát and Katharina Anna Zweig. A fixed degree sequence model for the one-mode projection of multiplex bipartite graphs. *Social Network Analysis and Mining*, 3(4):1209–1224, 2013.

[18] David R. Hunter, Steven M. Goodreau, and Mark S. Handcock. Goodness of fit of social network models. *Journal of the American Statistical Association*, 103(481):248–258, 2008.

[19] Alden S. Klovdahl. Social networks and the spread of infectious diseases: The AIDS example. *Social Science & Medicine*, 21(11):1203–1216, 1985.

[20] Michihiro Kuramochi and George Karypis. Finding frequent patterns in a large sparse graph. *Data Mining and Knowledge Discovery*, 11(3):243–271, 2005.

[21] Sergei Maslov, Kim Sneppen, and Alexei Zaliznyak. Detection of topological patterns in complex networks: correlation profile of the internet. *Physica A: Statistical Mechanics and its Applications*, 333:529–540, 2004.

[22] Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, ii. *Journal of Symbolic Computation*, 60:94–112, 2014.

[23] Giulia Menichetti, Daniel Remondini, Pietro Panzarasa, Raúl J. Mondragón, and Ginestra Bianconi. Weighted multiplex networks. *PLOS ONE*, 9(6):1–8, 2014.

[24] Manuel Middendorf, Etay Ziv, and Chris H. Wiggins. Inferring network mechanisms: The drosophila melanogaster protein interaction network. In *PNA*, volume 102, pages 3192–3197. 2004.

[25] Ron Milo, Shalev Itzkovitz, Nadav Kashtan, Reuven Levitt, Shai Shen-Orr, Inbal Ayzenshtat, Michal Sheffer, and Uri Alon. Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–1542, 2004.

[26] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, D. Chklovskii, and Uri Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[27] Zachary Neal. The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors. *Social Networks*, 39:84–97, 2014.

[28] Mark E. J. Newman and Juyong Park. Why social networks are different from other types of networks. *Phys. Rev. E*, 68:036–122, 2003.

[29] Mark E. J. Newman, Steven H. Strogatz, and Duncan J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64:026–118, 2001.

[30] Takaaki Ohnishi, Hideki Takayasu, and Misako Takayasu. Network motifs in an inter-firm network. *Journal of Economic Interaction and Coordination*, 5(2):171–180, 2010.

[31] Pedro Ribeiro and Fernando Silva. G-tries: An efficient data structure for discovering network motifs. In *Proceedings of the ACM Symposium on Applied Computing*, SAC '10, pages 1559–1566, 2010.

[32] Matthew Rocklin and Ali Pinar. On clustering on graphs with multiple edge types. *Internet Mathematics*, 9:82–112, 2013.

[33] Lotte Romijn, Breanndán Ó. Nualláin, and Leen Torenvliet. Discovering motifs in real-world social networks. In *Proceedings of SOFSEM on Theory and Practice of Computer Science*, pages 463–474, 2015.

[34] Shahrivari Saeed and Jalili Saeed. Fast parallel all-subgraph enumeration using multicore machines. *Scientific Programming*, 2015:901321:1–901321:11, 2015.

[35] David Stark and Balázs Vedres. Social Times of Network Spaces: Network Sequences and Foreign Investment in Hungary. *American Journal of Sociology*, 111(5):1367–1412, 2006.

[36] Stefania Vitali, James B. Glattfelder, and Stefano Battiston. The network of global corporate control. *PLOS ONE*, 6(10):1–6, 2011.

[37] Sebastian Wernicke. A faster algorithm for detecting network motifs. In *Proceedings of Algorithms in Bioinformatics*, pages 165–177, 2005.

[38] Paul Windolf and Jürgen Beyer. Co-operative Capitalism: Corporate Networks in Germany and Britain. *The British Journal of Sociology*, 47(2):205–231, 1996.

[39] Wayne W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.

A Apendix

A.1 Graph models

The pseudo-code for the graph models discussed in Section 6.1.1.

A.1.1 Chung-Lu models

Algorithm 1 Directed Chung-Lu

INITIALIZE L_u = number of edges in the original graph
INITIALIZE in-degree = in-degree of each node in the original graph
INITIALIZE out-degree = out-degree of each node in the original graph
INITIALIZE edges = empty list
for $i \in$ all nodes **do**
  for $j \in$ all nodes **do**
    add edge(i, j) to edges with probability: $\dfrac{\text{out-degree}(i) * \text{in-degree}(j)}{L_u}$
  end for
end for
return edges

A.1.2 Park-Newman models

We choose adjuster $\alpha = 33500$, as this fits our empirical data best.

Algorithm 2 Directed Park-Newman

INITIALIZE in-degree = in-degree of each node in the original graph
INITIALIZE out-degree = out-degree of each node in the original graph
INITIALIZE adjuster = α
INITIALIZE edges = empty list
for $i \in$ all nodes **do**
  for $j \in$ all nodes **do**
    add edge(i, j) to edges with probability: $\dfrac{\text{out-degree}(i) * \text{in-degree}(j) * \text{adjuster}^{-1}}{\text{out-degree}(i) * \text{in-degree}(j) * \text{adjuster}^{-1} + 1}$
  end for
end for
return edges

A.1.3 Stub-matching model

The stub matching model is implemented by the Python package *networkx*, in Python 3.5.

Algorithm 3 Directed Stub-matching

INITIALIZE in-degree = in-degree of each node in the original graph
INITIALIZE out-degree = out-degree of each node in the original graph
D =networkx.directed_configuration_model(in-degree,out-degree)
Remove parallel edges from D
Remove self loops from D
Return edges(D)

A.1.4 One-mode projection

If the model produces a bipartite graph, a one-mode projection must be made. The following algorithm creates this projection. It assumes two sets of nodes, one of which has only outgoing edges, and one has only incoming edges. The one-mode projection is made of the node set that has incoming edges.

Algorithm 4 One mode projection

```
INITIALIZE G = bipartite graph, as networkx graph object
INITIALIZE bipart-set = empty list
INITIALIZE edges = empty list
for edge ∈ G do
   bs = edge[0]
   if bs in bipart-set then
      continue
   end if
   bipart-set.add(bs)
   for source in G[bs] do
      for target in G[bs] do
         if source ≠ target and source < target then
            add edge(source, target) to edges
         end if
      end for
   end for
end for
return edges
```

A.1.5 Multiplex Model

The multiplex stub-matching algorithm creates a separate network for each type of edge, and then merges all networks into a multiplex network. However this creates a problem with the interlock edges. When two nodes have both an interlock and an ownership relation, the relation becomes a directed multiplex relation. This leaves one directed interlock relation instead of an undirected interlock edge because half of the undirected relation is merged into a multiplex relation. An example is given in Figure 50. Figure 50a shows two companies, A and B, where A owns B (blue directed edge), and both share board members (two black directed edges). Figure 50b shows the same two companies A and B, but now the ownership and interlock relations from A to B have been merged into one combined relation (red directed edge).

(a) As multiple relations:
Black: Interlock, undirected relation.
Red: Ownership, directed relation.

(b) As multiplex relations:
Black: Interlock, undirected relation.
Blue: Combined, directed relation.

Figure 50: Two companies with interlock and ownership ties

Furthermore a choice must be made on how to model a combination edge. Part of the multiplex edge (ownership) is a direct relation between two companies. The other part (interlock) is a bipartite relation between one or more board members and two companies. Since a combined edge can not exist without an interlock edge, it must follow the rules of an interlock edge. Therefore multiplex edges are also modeled bipartite. This means that a board member can be seen as responsible for the multiplex relation. If two companies A and B have a multiplex relation from A to B, we select a single board member P to be responsible for the combined relation. P loses its interlock edges with A and B, and gains a multiplex relation with both A and B. To keep track of the direction of the original multiplex edge from A to B, we split the combined relation into two parts: one part is a directed edge from a company to a board member (from A to P), and one part is a directed edge from a board member to a company (from P to B). A also loses its ownership out-edge, and B loses its ownership in-edge. This edge is already taken into account with the combination edge. See Figure 51 for an example.

(a) As multiple relations:
Black: Interlock.
Red: Ownership.

(b) As multiplex relations:
Blue: Company to board member.
Green: Board member to company.

Figure 51: Bipartite projection of Figure 50

The degree sequence of the graph from Figure 51 thus makes many changes before the modeling is done. These changes are as follows:

- The multiplex network of node A and B has the following degree sequences:
  - $D_{\text{in-ownership}} = [0, 1]$
  - $D_{\text{out-ownership}} = [1, 0]$
  - $D_{\text{interlock}} = [1, 1]$
- The interlock part is made bipartite with the introduction of node P (Figure 51a). The degree sequences of nodes A, B and P are now:
  - $D_{\text{in-ownership}} = [0, 1, 0]$
  - $D_{\text{out-ownership}} = [1, 0, 0]$
  - $D_{\text{in-interlock}} = [1, 1, 0]$

- $D_{\text{out-interlock}} = [0, 0, 2]$

- As the relation between A and B is multiplex, the ownership and interlock edge are merged. The degree sequences of nodes A, B and P are now:
  - $D_{\text{in-ownership}} = [0, 0, 0]$
  - $D_{\text{out-ownership}} = [0, 0, 0]$
  - $D_{\text{in-interlock}} = [0, 0, 0]$
  - $D_{\text{out-interlock}} = [0, 0, 0]$
  - $D_{\text{in-multiplex1}} = [0, 0, 1]$
  - $D_{\text{out-multiplex1}} = [1, 0, 0]$
  - $D_{\text{in-multiplex2}} = [0, 1, 0]$
  - $D_{\text{out-multiplex2}} = [0, 0, 1]$

This means that after merging ownership and interlock into multiplex before modeling, the degree sequence of the leftover interlock network does not represent an undirected network. Therefore only interlock edges are considered where the reverse edge is not condensed into a multiplex edge.

When all edge types have been modeled separately, the edges are then merged into one edge-list which resembles the multiplex graph. Each multiplex edge is given a reverse interlock edge, to correct for the removal of interlock edges that where condensed into a multiplex edge.

Algorithm 5 Multiplex Stub-matching

```
INITIALIZE G = original graph
INITIALIZE o-in-degree = in-degree of type1 edges for each node in G
INITIALIZE o-out-degree = out-degree of type1 edges for each node in G
INITIALIZE i-in-degree = in-degree of type2 edges for each node in G
INITIALIZE i-out-degree = out-degree of type2 edges for each node in G
INITIALIZE m-edges = all multiplex edges between nodes
INITIALIZE m-in-degree = in-degree of multiplex edges for each node in G
INITIALIZE m-out-degree = out-degree of multiplex edges for each node in G

for edge in m-edges do
  pick random node connected to this edge, with non-zero i-out-degree
  adjust degrees of this node and source and target of edge
end for

O =networkx.directed_configuration_model(o-in-degree,o-out-degree)
Remove parallel edges and self loops of O
I =networkx.configuration_model(i-degree)
Remove parallel edges and self loops of I

M = empty list
for bm in rand-bm do
  for i = 1 to size(m-in-degree[bm]) do
    pick source at random out of all nodes with a m-out-degree
    decrease m-out-degree[source] by 1
  end for
  for i = 1 to size(m-out-degree[bm]) do
    pick target at random out of all nodes with a m-out-degree
    decrease m-in-degree[target] by 1
  end for
  create an edge between every source and target, store in M
end for
all-edges= empty list
for edge in O do
  all-edges[edge] = 1
end for
for edge in I do
  all-edges[edge] = 2
end for
for edge in M do
  all-edges[edge] = 3
  if all-edges[reverse(edge)] ≠ 3 then
    all-edges[reverse(edge)] = 2
  end if
end for
return all-edges
```

A.2 Motif Combined Result Tables

A.2.1 Size 3

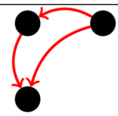
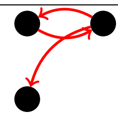
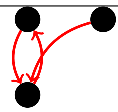
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$		278	0.418%	
2	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$		26	0.026%	
3	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$		-	0.01%	

Figure 52: Ratio and concentration of size 3 motifs in the ownership network

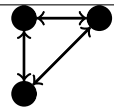
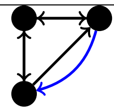
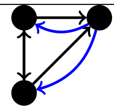
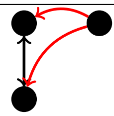
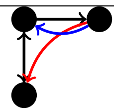
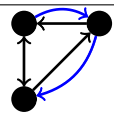
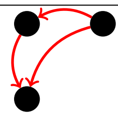
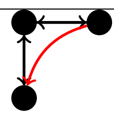
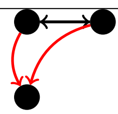
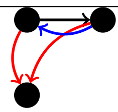
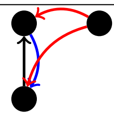
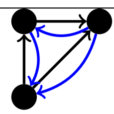
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 2 \\ 2 & 2 & 0 \end{bmatrix}$		1462	20.791%	Circle,Clique
2	$\begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 2 \\ 3 & 2 & 0 \end{bmatrix}$		1858	1.918%	Circle,Clique
3	$\begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 2 \\ 3 & 3 & 0 \end{bmatrix}$		-	0.945%	Circle,Clique
4	$\begin{bmatrix} 0 & 2 & 0 \\ 2 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$		1170	0.805%	Circle,Clique
5	$\begin{bmatrix} 0 & 2 & 0 \\ 2 & 0 & 2 \\ 1 & 3 & 0 \end{bmatrix}$		1032	0.118%	Circle,Clique
6	$\begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 3 \\ 3 & 2 & 0 \end{bmatrix}$		-	0.104%	Circle,Clique
7	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$		97	0.078%	Circle,Clique
8	$\begin{bmatrix} 0 & 2 & 0 \\ 2 & 0 & 2 \\ 1 & 2 & 0 \end{bmatrix}$		65	0.075%	Circle,Clique
9	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 2 \\ 1 & 2 & 0 \end{bmatrix}$		516	0.059%	Circle,Clique
10	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 2 \\ 1 & 3 & 0 \end{bmatrix}$		-	0.056%	Circle,Clique
11	$\begin{bmatrix} 0 & 2 & 0 \\ 3 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}$		-	0.021%	Circle,Clique
12	$\begin{bmatrix} 0 & 2 & 2 \\ 3 & 0 & 2 \\ 3 & 3 & 0 \end{bmatrix}$		-	0.018%	Circle,Clique

Figure 53: Ratio and concentration of size 3 motifs in the multiplex network (part 1)

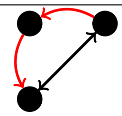
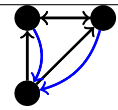
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
13	$\begin{bmatrix} 0 & 0 & 2 \\ 1 & 0 & 0 \\ 2 & 1 & 0 \end{bmatrix}$		-	0.017%	Circle,Clique
14	$\begin{bmatrix} 0 & 2 & 2 \\ 3 & 0 & 2 \\ 3 & 2 & 0 \end{bmatrix}$		-	0.016%	Circle,Clique

Figure 54: Ratio and concentration of size 3 motifs in the multiplex network (part 2)

A.2.2 Size 4

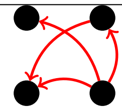
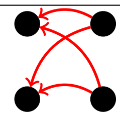
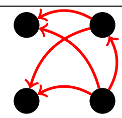
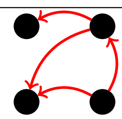
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$		100	0.436%	
2	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$		2024	0.351%	
3	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$		-	0.068%	
4	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}$		29	0.016%	

Figure 55: Ratio and concentration of size 4 motifs in the ownership network

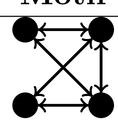
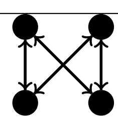
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$		171	3.802%	
2	$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$		42	0.104%	Circle

Figure 56: Ratio and concentration of size 4 motifs in the interlocks network

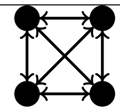
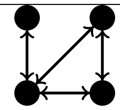
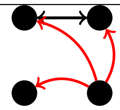
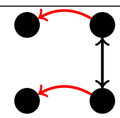
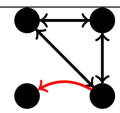
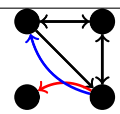
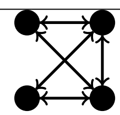
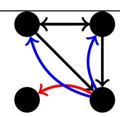
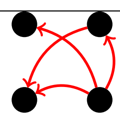
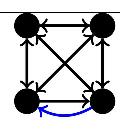
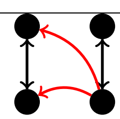
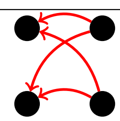
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0&2&2&2 \\ 2&0&2&2 \\ 2&2&0&2 \\ 2&2&2&0 \end{bmatrix}$		-	2.43%	Clique
2	$\begin{bmatrix} 0&2&2&2 \\ 2&0&0&0 \\ 2&0&0&2 \\ 2&0&2&0 \end{bmatrix}$		65	1.353%	Circle,Clique
3	$\begin{bmatrix} 0&0&0&0 \\ 0&0&2&0 \\ 0&2&0&0 \\ 1&1&1&0 \end{bmatrix}$		341	1.216%	Circle,Clique
4	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 0&1&0&2 \\ 1&0&2&0 \end{bmatrix}$		9	0.717%	Pyramid
5	$\begin{bmatrix} 0&0&0&0 \\ 0&0&2&2 \\ 0&2&0&2 \\ 1&2&2&0 \end{bmatrix}$		500	0.536%	Circle,Clique
6	$\begin{bmatrix} 0&0&0&0 \\ 0&0&2&2 \\ 0&2&0&2 \\ 1&3&2&0 \end{bmatrix}$		1632	0.468%	Circle,Clique
7	$\begin{bmatrix} 0&0&2&2 \\ 0&0&2&2 \\ 2&2&0&2 \\ 2&2&2&0 \end{bmatrix}$		18241	0.255%	
8	$\begin{bmatrix} 0&0&0&0 \\ 0&0&2&2 \\ 0&2&0&2 \\ 1&3&3&0 \end{bmatrix}$		7556	0.229%	Circle,Clique
9	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 1&0&0&0 \\ 1&1&1&0 \end{bmatrix}$		38	0.149%	Circle,Clique
10	$\begin{bmatrix} 0&2&2&2 \\ 2&0&2&2 \\ 2&2&0&2 \\ 3&2&2&0 \end{bmatrix}$		-	0.134%	Clique
11	$\begin{bmatrix} 0&2&0&0 \\ 2&0&0&0 \\ 0&0&0&2 \\ 1&1&2&0 \end{bmatrix}$		611	0.127%	Circle,Clique
12	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 1&1&0&0 \\ 1&1&0&0 \end{bmatrix}$		363	0.126%	Circle

Figure 57: Ratio and concentration of size 4 motifs in the multiplex network (part 1)

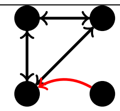
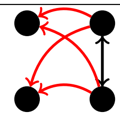
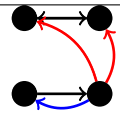
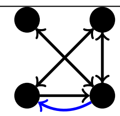
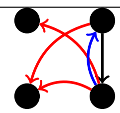
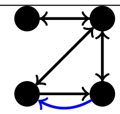
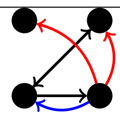
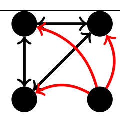
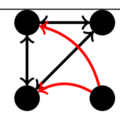
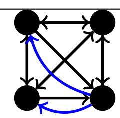
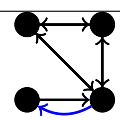
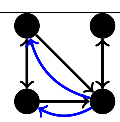
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
13	$\begin{bmatrix} 0&2&2&0 \\ 2&0&2&0 \\ 2&2&0&0 \\ 1&0&0&0 \end{bmatrix}$		352	0.121%	Circle,Clique
14	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 1&1&0&2 \\ 1&1&2&0 \end{bmatrix}$		-	0.1%	
15	$\begin{bmatrix} 0&0&0&2 \\ 0&0&2&0 \\ 0&2&0&0 \\ 3&1&1&0 \end{bmatrix}$		986	0.096%	Circle,Clique
16	$\begin{bmatrix} 0&0&2&2 \\ 0&0&0&2 \\ 2&0&0&2 \\ 3&2&2&0 \end{bmatrix}$		334	0.092%	Circle,Clique
17	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 1&0&0&2 \\ 1&1&3&0 \end{bmatrix}$		776	0.089%	Circle,Clique
18	$\begin{bmatrix} 0&0&2&2 \\ 0&0&2&0 \\ 2&2&0&2 \\ 3&0&2&0 \end{bmatrix}$		178	0.087%	Circle,Clique
19	$\begin{bmatrix} 0&0&2&2 \\ 0&0&0&0 \\ 2&0&0&0 \\ 3&1&1&0 \end{bmatrix}$		181	0.081%	Circle,Clique
20	$\begin{bmatrix} 0&2&2&0 \\ 2&0&2&0 \\ 2&2&0&0 \\ 1&1&1&0 \end{bmatrix}$		-	0.077%	Clique
21	$\begin{bmatrix} 0&2&2&0 \\ 2&0&2&0 \\ 2&2&0&0 \\ 1&1&0&0 \end{bmatrix}$		-	0.07%	
22	$\begin{bmatrix} 0&2&2&2 \\ 2&0&2&2 \\ 2&2&0&2 \\ 3&3&2&0 \end{bmatrix}$		-	0.067%	Clique
23	$\begin{bmatrix} 0&0&0&2 \\ 0&0&2&2 \\ 0&2&0&2 \\ 3&2&2&0 \end{bmatrix}$		193	0.063%	Circle,Clique
24	$\begin{bmatrix} 0&2&0&2 \\ 2&0&0&2 \\ 0&0&0&2 \\ 3&3&2&0 \end{bmatrix}$		2683	0.044%	Circle,Clique

Figure 58: Ratio and concentration of size 4 motifs in the multiplex network (part 2)

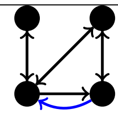
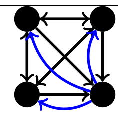
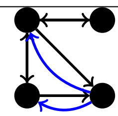
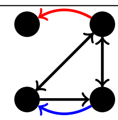
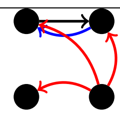
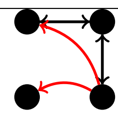
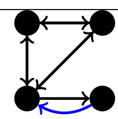
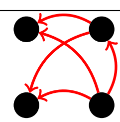
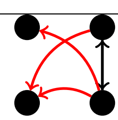
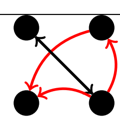
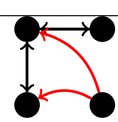
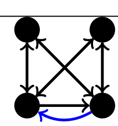
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
25	$\begin{bmatrix} 0&2&2&2 \\ 2&0&0&0 \\ 2&0&0&2 \\ 3&0&2&0 \end{bmatrix}$		160	0.044%	Circle,Clique
26	$\begin{bmatrix} 0&2&2&2 \\ 2&0&2&2 \\ 2&2&0&2 \\ 3&3&3&0 \end{bmatrix}$		-	0.043%	Clique
27	$\begin{bmatrix} 0&2&0&2 \\ 2&0&2&2 \\ 0&2&0&0 \\ 3&3&0&0 \end{bmatrix}$		551	0.033%	Circle,Clique
28	$\begin{bmatrix} 0&0&2&2 \\ 0&0&0&0 \\ 2&1&0&2 \\ 3&0&2&0 \end{bmatrix}$		1778	0.033%	Circle,Clique
29	$\begin{bmatrix} 0&0&0&0 \\ 0&0&2&0 \\ 0&3&0&0 \\ 1&1&1&0 \end{bmatrix}$		163	0.032%	Circle,Clique
30	$\begin{bmatrix} 0&0&0&0 \\ 0&0&2&0 \\ 0&2&0&2 \\ 1&1&2&0 \end{bmatrix}$		15	0.032%	Circle,Clique
31	$\begin{bmatrix} 0&2&2&2 \\ 2&0&2&0 \\ 2&2&0&0 \\ 3&0&0&0 \end{bmatrix}$		88	0.031%	Circle,Clique
32	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 1&1&0&0 \\ 1&1&1&0 \end{bmatrix}$		11490	0.027%	
33	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 1&0&0&2 \\ 1&1&2&0 \end{bmatrix}$		48	0.026%	Circle,Clique
34	$\begin{bmatrix} 0&0&0&0 \\ 0&0&0&2 \\ 1&0&0&0 \\ 1&2&1&0 \end{bmatrix}$		93	0.022%	Circle,Clique
35	$\begin{bmatrix} 0&2&0&0 \\ 2&0&2&0 \\ 0&2&0&0 \\ 1&1&0&0 \end{bmatrix}$		42	0.02%	Circle,Clique
36	$\begin{bmatrix} 0&2&2&2 \\ 2&0&0&2 \\ 2&0&0&2 \\ 3&2&2&0 \end{bmatrix}$		-	0.019%	

Figure 59: Ratio and concentration of size 4 motifs in the multiplex network (part 3)

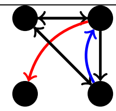
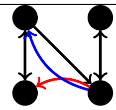
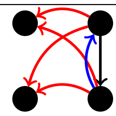
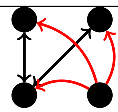
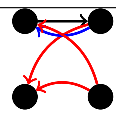
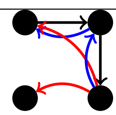
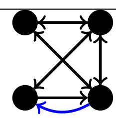
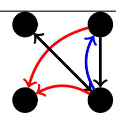
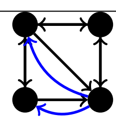
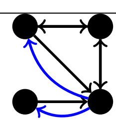
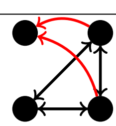
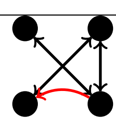
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
37	$\begin{bmatrix} 0&0&0&0\\ 0&0&2&2\\ 1&2&0&2\\ 0&2&3&0 \end{bmatrix}$		1013	0.019%	Circle,Clique
38	$\begin{bmatrix} 0&2&0&0\\ 2&0&0&2\\ 0&0&0&2\\ 1&3&2&0 \end{bmatrix}$		328	0.017%	Circle,Clique
39	$\begin{bmatrix} 0&0&0&0\\ 0&0&0&0\\ 1&1&0&2\\ 1&1&3&0 \end{bmatrix}$		-	0.016%	
40	$\begin{bmatrix} 0&2&2&0\\ 2&0&0&0\\ 2&0&0&0\\ 1&1&1&0 \end{bmatrix}$		-	0.014%	
41	$\begin{bmatrix} 0&0&0&0\\ 0&0&2&0\\ 1&3&0&0\\ 1&1&0&0 \end{bmatrix}$		949	0.013%	Circle
42	$\begin{bmatrix} 0&0&0&0\\ 0&0&2&0\\ 0&3&0&2\\ 1&1&3&0 \end{bmatrix}$		1374	0.013%	Circle,Clique
43	$\begin{bmatrix} 0&0&2&2\\ 0&0&2&2\\ 2&2&0&2\\ 3&2&2&0 \end{bmatrix}$		-	0.013%	
44	$\begin{bmatrix} 0&0&0&0\\ 0&0&0&2\\ 1&0&0&2\\ 1&2&3&0 \end{bmatrix}$		881	0.012%	Circle,Clique
45	$\begin{bmatrix} 0&2&0&2\\ 2&0&2&2\\ 0&2&0&2\\ 3&3&2&0 \end{bmatrix}$		-	0.011%	
46	$\begin{bmatrix} 0&0&0&2\\ 0&0&2&2\\ 0&2&0&2\\ 3&3&2&0 \end{bmatrix}$		139	0.011%	Circle,Clique
47	$\begin{bmatrix} 0&0&2&2\\ 0&0&0&0\\ 2&1&0&2\\ 2&1&2&0 \end{bmatrix}$		-	0.01%	
48	$\begin{bmatrix} 0&0&2&0\\ 0&0&0&2\\ 2&0&0&2\\ 1&2&2&0 \end{bmatrix}$		25	0.01%	Circle,Clique

Figure 60: Ratio and concentration of size 4 motifs in the multiplex network (part 4)

A.2.3 Size 5

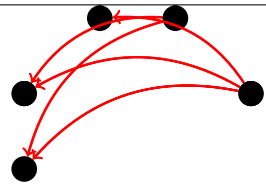
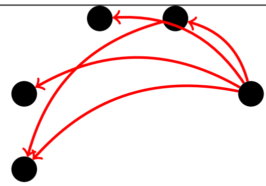
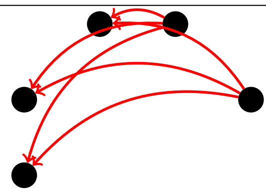
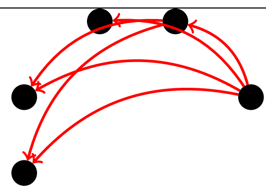
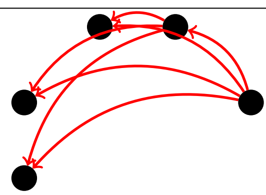
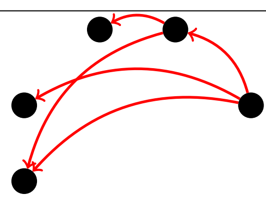
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&0&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		642	0.623%	
2	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&0&0&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		57	0.488%	
3	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&1&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		584907	0.391%	
4	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&0&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		28619	0.114%	
5	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&1&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		-	0.029%	
6	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&0&1&0&0 \\ 1&1&0&1&0 \end{bmatrix}$		10	0.015%	

Figure 61: Ratio and concentration of size 5 motifs in the ownership network

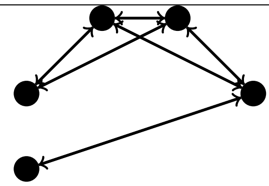
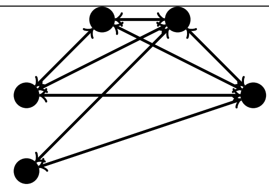
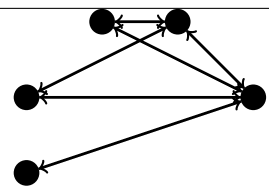
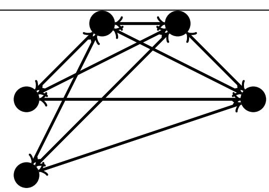
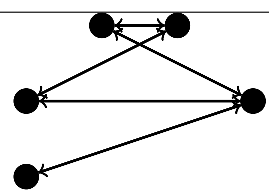
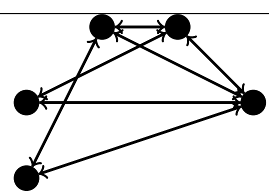
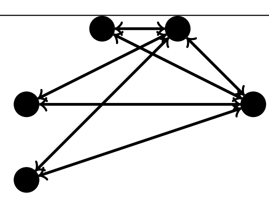
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0&0&0&0&1 \\ 0&0&1&1&0 \\ 0&1&0&1&1 \\ 0&1&1&0&1 \\ 1&0&1&1&0 \end{bmatrix}$		66	2.133%	
2	$\begin{bmatrix} 0&0&0&1&1 \\ 0&0&1&1&1 \\ 0&1&0&1&1 \\ 1&1&1&0&1 \\ 1&1&1&1&0 \end{bmatrix}$		82	2.039%	
3	$\begin{bmatrix} 0&0&0&0&1 \\ 0&0&0&1&1 \\ 0&0&0&1&1 \\ 0&1&1&0&1 \\ 1&1&1&1&0 \end{bmatrix}$		20	1.362%	
4	$\begin{bmatrix} 0&0&1&1&1 \\ 0&0&1&1&1 \\ 1&1&0&1&1 \\ 1&1&1&0&1 \\ 1&1&1&1&0 \end{bmatrix}$		63435	1.007%	
5	$\begin{bmatrix} 0&0&0&0&1 \\ 0&0&0&1&1 \\ 0&0&0&1&1 \\ 0&1&1&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		20	0.301%	Circle
6	$\begin{bmatrix} 0&0&1&0&1 \\ 0&0&0&1&1 \\ 1&0&0&1&1 \\ 0&1&1&0&1 \\ 1&1&1&1&0 \end{bmatrix}$		43	0.294%	
7	$\begin{bmatrix} 0&0&0&1&1 \\ 0&0&0&1&1 \\ 0&0&0&1&1 \\ 1&1&1&0&1 \\ 1&1&1&1&0 \end{bmatrix}$		11421	0.159%	

Figure 62: Ratio and concentration of size 5 motifs in the interlocks network (part 1)

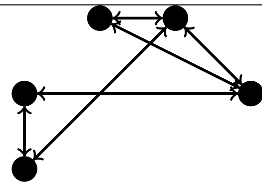
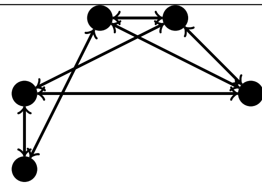
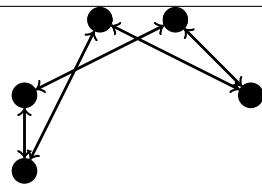
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
8	$\begin{bmatrix} 0&1&0&1&0 \\ 1&0&0&0&1 \\ 0&0&0&1&1 \\ 1&0&1&0&1 \\ 0&1&1&1&0 \end{bmatrix}$		18	0.091%	
9	$\begin{bmatrix} 0&1&1&0&0 \\ 1&0&0&1&1 \\ 1&0&0&1&1 \\ 0&1&1&0&1 \\ 0&1&1&1&0 \end{bmatrix}$		-	0.021%	
10	$\begin{bmatrix} 0&1&1&0&0 \\ 1&0&0&1&0 \\ 1&0&0&0&1 \\ 0&1&0&0&1 \\ 0&0&1&1&0 \end{bmatrix}$		24	0.018%	Circle

Figure 63: Ratio and concentration of size 5 motifs in the interlocks network (part 2)

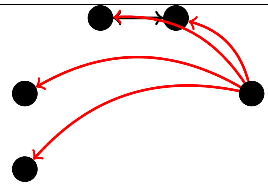
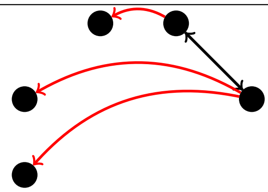
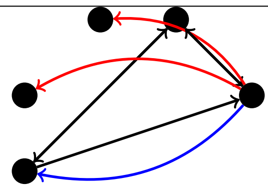
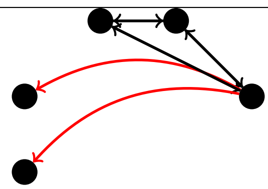
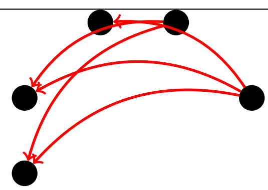
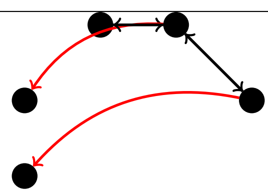
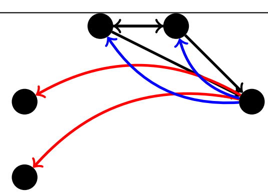
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
1	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&0 \\ 0&0&2&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		269	1.932%	Circle,Clique
2	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&1&0&2 \\ 1&1&0&2&0 \end{bmatrix}$		10	0.868%	Pyramid
3	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 2&0&0&0&2 \\ 3&1&1&2&0 \end{bmatrix}$		2346	0.673%	Circle,Clique
4	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&2 \\ 0&0&2&0&2 \\ 1&1&2&2&0 \end{bmatrix}$		873	0.559%	Circle,Clique
5	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&0&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		172	0.361%	Circle
6	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&0 \\ 0&1&2&0&2 \\ 1&0&0&2&0 \end{bmatrix}$		9	0.3%	Pyramid
7	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&2 \\ 0&0&2&0&2 \\ 1&1&3&3&0 \end{bmatrix}$		8770	0.257%	Circle,Clique

Figure 64: Ratio and concentration of size 5 motifs in the multiplex network (part 1)

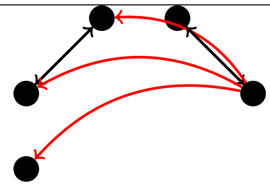
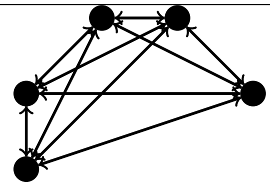
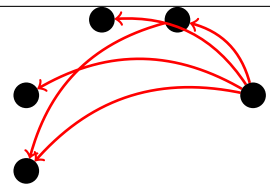
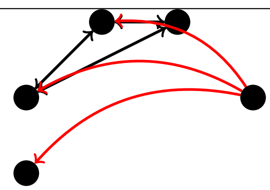
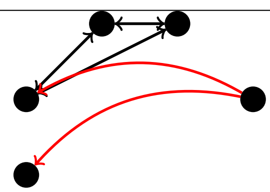
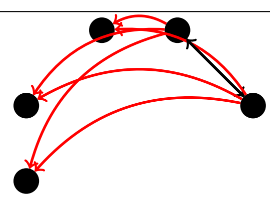
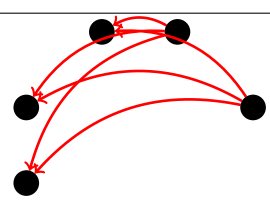
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
8	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&0&0 \\ 0&2&0&0&0 \\ 0&0&0&0&2 \\ 1&1&1&2&0 \end{bmatrix}$		306	0.216%	Circle,Clique
9	$\begin{bmatrix} 0&2&2&2&2 \\ 2&0&2&2&2 \\ 2&2&0&2&2 \\ 2&2&2&0&2 \\ 2&2&2&2&0 \end{bmatrix}$		-	0.194%	Clique
10	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&0&0&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		23	0.181%	Circle,Clique
11	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&0 \\ 0&2&0&2&0 \\ 0&2&2&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		285266	0.158%	
12	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&0 \\ 0&2&0&2&0 \\ 0&2&2&0&0 \\ 1&1&0&0&0 \end{bmatrix}$		969	0.146%	Circle,Clique
13	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&1&0&2 \\ 1&1&1&2&0 \end{bmatrix}$		-	0.126%	
14	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&1&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		40572	0.123%	

Figure 65: Ratio and concentration of size 5 motifs in the multiplex network (part 2)

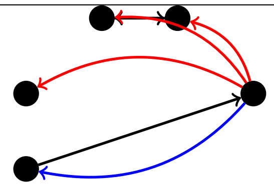
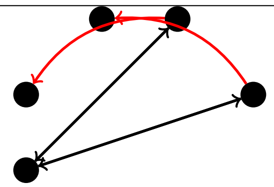
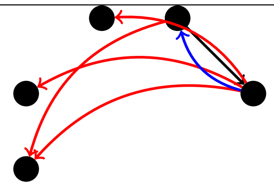
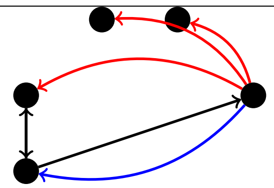
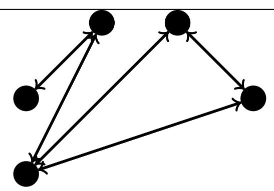
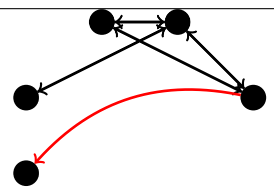
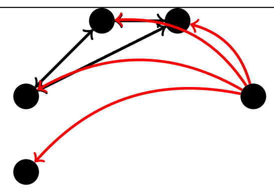
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
15	$\begin{bmatrix} 0&0&0&0&2 \\ 0&0&0&0&0 \\ 0&0&0&2&0 \\ 0&0&2&0&0 \\ 3&1&1&1&0 \end{bmatrix}$		316	0.104%	Circle,Clique
16	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 2&1&0&0&0 \\ 2&0&1&0&0 \end{bmatrix}$		10	0.092%	Pyramid
17	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&0&0&0&2 \\ 1&1&1&3&0 \end{bmatrix}$		468	0.089%	Circle,Clique
18	$\begin{bmatrix} 0&2&0&0&2 \\ 2&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 3&1&1&1&0 \end{bmatrix}$		110	0.086%	Circle,Clique
19	$\begin{bmatrix} 0&0&2&2&2 \\ 0&0&2&0&0 \\ 2&2&0&0&0 \\ 2&0&0&0&2 \\ 2&0&0&2&0 \end{bmatrix}$		33	0.081%	Circle,Clique
20	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&2&0 \\ 0&0&0&2&2 \\ 0&2&2&0&2 \\ 1&0&2&2&0 \end{bmatrix}$		112	0.077%	Circle,Clique
21	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&0 \\ 0&2&0&2&0 \\ 0&2&2&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		-	0.077%	Clique

Figure 66: Ratio and concentration of size 5 motifs in the multiplex network (part 3)

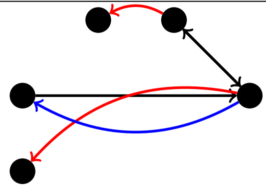
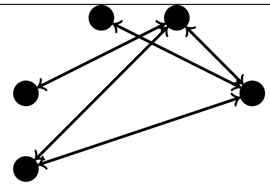
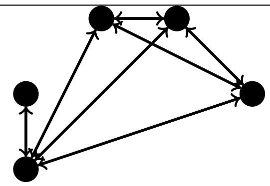
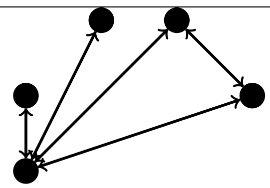
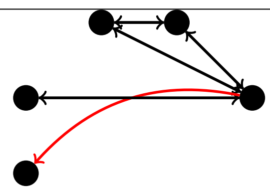
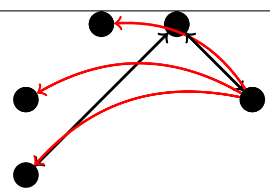
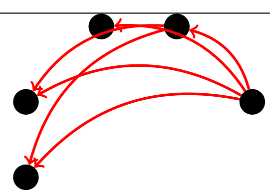
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
22	$\begin{bmatrix}0&0&0&0&0\\0&0&0&0&2\\0&0&0&0&0\\0&0&1&0&2\\1&3&0&2&0\end{bmatrix}$		7	0.076%	Pyramid
23	$\begin{bmatrix}0&0&0&2&2\\0&0&0&2&0\\0&0&0&0&2\\2&2&0&0&2\\2&0&2&2&0\end{bmatrix}$		11	0.069%	Circle,Clique
24	$\begin{bmatrix}0&2&2&2&2\\2&0&0&0&0\\2&0&0&2&2\\2&0&2&0&2\\2&0&2&2&0\end{bmatrix}$		2235624	0.069%	Clique
25	$\begin{bmatrix}0&2&2&2&2\\2&0&0&0&0\\2&0&0&0&0\\2&0&0&0&2\\2&0&0&2&0\end{bmatrix}$		13	0.056%	Circle,Clique
26	$\begin{bmatrix}0&0&0&0&0\\0&0&0&0&2\\0&0&0&2&2\\0&0&2&0&2\\1&2&2&2&0\end{bmatrix}$		156	0.052%	Circle,Clique
27	$\begin{bmatrix}0&0&0&2&0\\0&0&0&0&0\\0&0&0&0&0\\2&0&0&0&2\\1&1&1&2&0\end{bmatrix}$		12	0.047%	Circle,Clique
28	$\begin{bmatrix}0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\1&1&0&0&0\\1&1&1&1&0\end{bmatrix}$		3987	0.044%	

Figure 67: Ratio and concentration of size 5 motifs in the multiplex network (part 4)

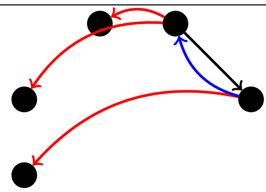
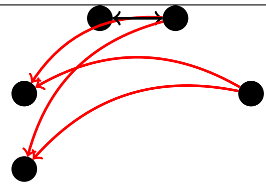
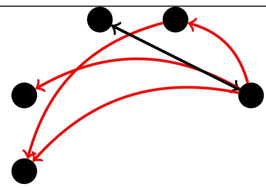
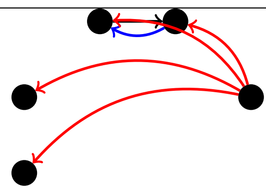
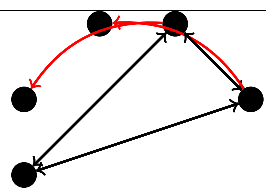
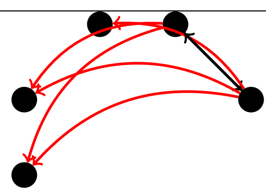
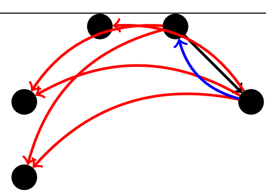
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
29	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&1&1&0&2 \\ 1&0&0&3&0 \end{bmatrix}$		7	0.039%	Pyramid
30	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&0 \\ 1&1&2&0&0 \\ 1&1&0&0&0 \end{bmatrix}$		305	0.039%	Circle
31	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&2 \\ 1&0&0&0&0 \\ 1&1&2&1&0 \end{bmatrix}$		47	0.037%	Circle,Clique
32	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&0 \\ 0&0&3&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		91	0.036%	Circle,Clique
33	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 2&1&0&0&2 \\ 2&0&1&2&0 \end{bmatrix}$		2019	0.035%	Circle,Clique
34	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&0&0&2 \\ 1&1&1&2&0 \end{bmatrix}$		7664	0.034%	
35	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&0&0&2 \\ 1&1&1&3&0 \end{bmatrix}$		133710	0.033%	

Figure 68: Ratio and concentration of size 5 motifs in the multiplex network (part 5)

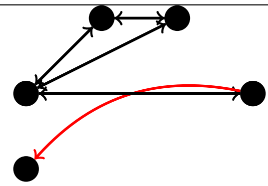
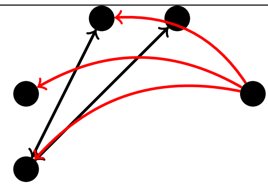
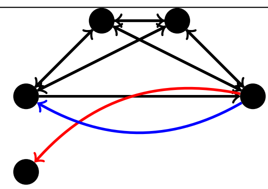
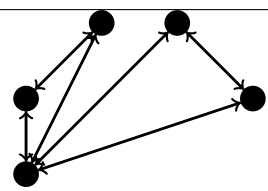
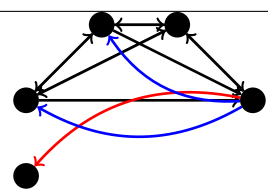
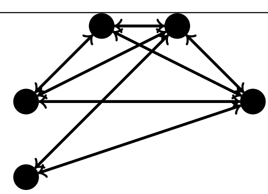
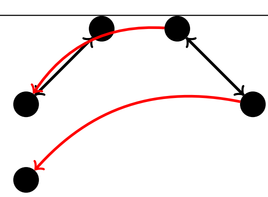
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
36	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&2 \\ 0&2&0&2&0 \\ 0&2&2&0&0 \\ 1&2&0&0&0 \end{bmatrix}$		183	0.033%	Circle,Clique
37	$\begin{bmatrix} 0&0&2&2&0 \\ 0&0&0&0&0 \\ 2&0&0&0&0 \\ 2&0&0&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		20	0.031%	Circle,Clique
38	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&2 \\ 0&2&0&2&2 \\ 0&2&2&0&2 \\ 1&3&2&2&0 \end{bmatrix}$		-	0.03%	Clique
39	$\begin{bmatrix} 0&2&2&2&2 \\ 2&0&2&0&0 \\ 2&2&0&0&0 \\ 2&0&0&0&2 \\ 2&0&0&2&0 \end{bmatrix}$		23978	0.027%	
40	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&2 \\ 0&2&0&2&2 \\ 0&2&2&0&2 \\ 1&3&3&2&0 \end{bmatrix}$		-	0.027%	Clique
41	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&2&2&2 \\ 0&2&0&2&2 \\ 2&2&2&0&2 \\ 2&2&2&2&0 \end{bmatrix}$		-	0.026%	
42	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&0&0 \\ 0&2&0&0&0 \\ 0&1&0&0&2 \\ 1&0&0&2&0 \end{bmatrix}$		8	0.025%	Pyramid

Figure 69: Ratio and concentration of size 5 motifs in the multiplex network (part 6)

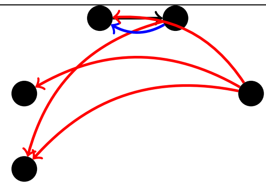
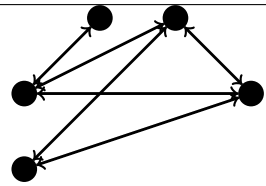
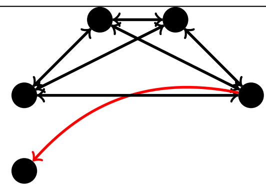
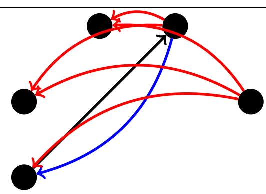
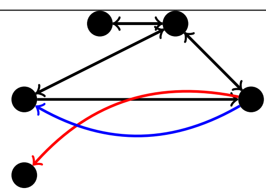
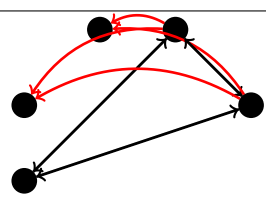
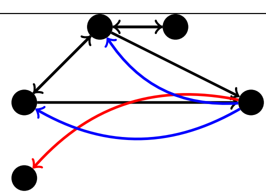
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
43	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&2&0 \\ 1&0&3&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		527	0.022%	Circle
44	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&2&2&2 \\ 0&2&0&0&0 \\ 2&2&0&0&2 \\ 2&2&0&2&0 \end{bmatrix}$		2087	0.02%	
45	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&2 \\ 0&2&0&2&2 \\ 0&2&2&0&2 \\ 1&2&2&2&0 \end{bmatrix}$		-	0.019%	Clique
46	$\begin{bmatrix} 0&0&0&2&0 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 3&1&1&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		200136	0.019%	
47	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&2&2 \\ 0&0&0&2&0 \\ 0&2&2&0&2 \\ 1&3&0&2&0 \end{bmatrix}$		198	0.019%	Circle,Clique
48	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 2&1&1&0&2 \\ 2&1&1&2&0 \end{bmatrix}$		-	0.017%	
49	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&0&2 \\ 0&2&0&2&2 \\ 0&0&2&0&0 \\ 1&3&3&0&0 \end{bmatrix}$		1534	0.016%	Circle,Clique

Figure 70: Ratio and concentration of size 5 motifs in the multiplex network (part 7)

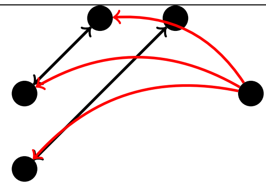
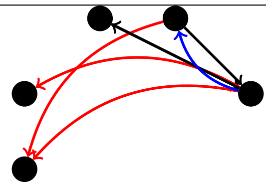
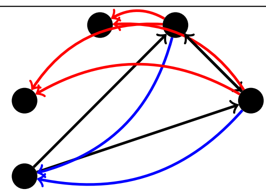
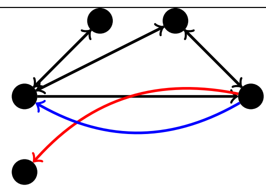
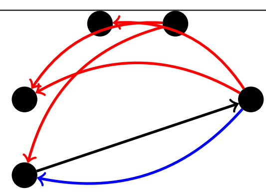
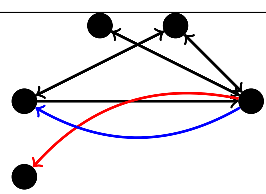
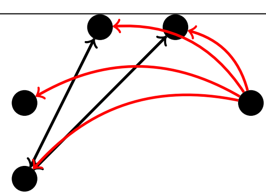
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
50	$\begin{bmatrix} 0&0&0&2&0 \\ 0&0&2&0&0 \\ 0&2&0&0&0 \\ 2&0&0&0&0 \\ 1&1&1&0&0 \end{bmatrix}$		202	0.016%	Circle,Clique
51	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&0&0&0&2 \\ 1&0&0&0&2 \\ 1&1&2&3&0 \end{bmatrix}$		606	0.016%	Circle,Clique
52	$\begin{bmatrix} 0&0&0&2&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 3&1&1&0&2 \\ 3&1&1&2&0 \end{bmatrix}$		-	0.016%	
53	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&2&2 \\ 0&2&0&0&0 \\ 0&2&0&0&2 \\ 1&3&0&2&0 \end{bmatrix}$		305	0.016%	Circle,Clique
54	$\begin{bmatrix} 0&0&0&0&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 1&1&0&0&0 \\ 3&1&1&0&0 \end{bmatrix}$		755	0.015%	Circle
55	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&2&2 \\ 0&0&0&0&2 \\ 0&2&0&0&2 \\ 1&3&2&2&0 \end{bmatrix}$		198	0.015%	Circle,Clique
56	$\begin{bmatrix} 0&0&2&2&0 \\ 0&0&0&0&0 \\ 2&0&0&0&0 \\ 2&0&0&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		5851	0.015%	

Figure 71: Ratio and concentration of size 5 motifs in the multiplex network (part 8)

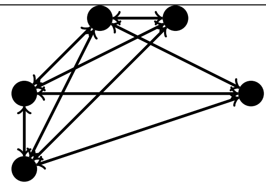
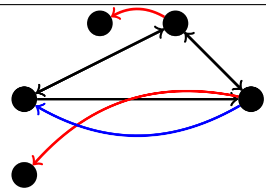
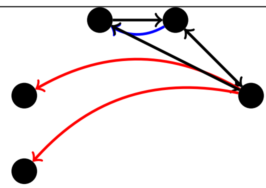
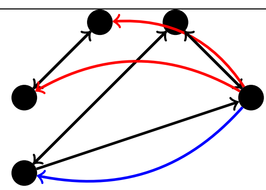
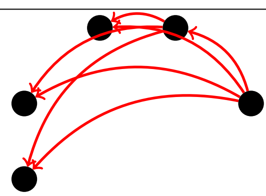
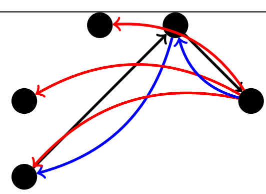
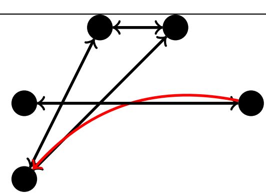
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
57	$\begin{bmatrix} 0 & 2 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 & 2 \\ 2 & 2 & 0 & 2 & 2 \\ 2 & 2 & 2 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \end{bmatrix}$		-	0.014%	
58	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 2 \\ 1 & 3 & 0 & 2 & 0 \end{bmatrix}$		3386	0.014%	Circle,Clique
59	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 3 & 0 & 2 \\ 1 & 1 & 2 & 2 & 0 \end{bmatrix}$		2011	0.013%	Circle,Clique
60	$\begin{bmatrix} 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 2 \\ 3 & 1 & 1 & 2 & 0 \end{bmatrix}$		-	0.013%	
61	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$		413857	0.013%	
62	$\begin{bmatrix} 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 2 \\ 1 & 1 & 1 & 3 & 0 \end{bmatrix}$		796	0.013%	Circle,Clique
63	$\begin{bmatrix} 0 & 0 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \\ 2 & 0 & 0 & 2 & 0 \\ 2 & 0 & 2 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 \end{bmatrix}$		550	0.012%	Circle,Clique

Figure 72: Ratio and concentration of size 5 motifs in the multiplex network (part 9)

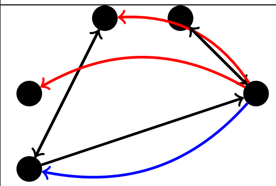
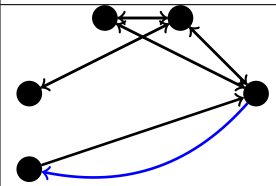
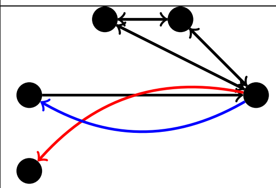
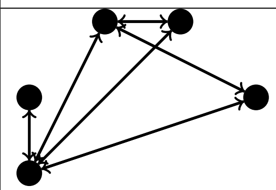
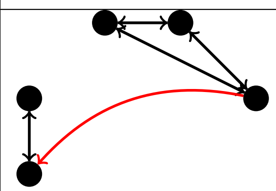
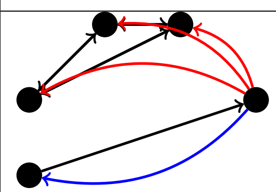
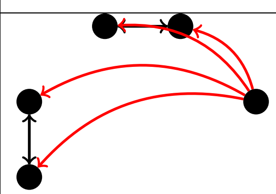
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
64	$\begin{bmatrix} 0&0&2&0&2 \\ 0&0&0&0&0 \\ 2&0&0&0&0 \\ 0&0&0&0&2 \\ 3&1&1&2&0 \end{bmatrix}$		125	0.012%	Circle,Clique
65	$\begin{bmatrix} 0&0&0&0&2 \\ 0&0&0&2&0 \\ 0&0&0&2&2 \\ 0&2&2&0&2 \\ 3&0&2&2&0 \end{bmatrix}$		59	0.012%	Circle,Clique
66	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&0&0&2 \\ 0&0&0&2&2 \\ 0&0&2&0&2 \\ 1&3&2&2&0 \end{bmatrix}$		141	0.012%	Circle,Clique
67	$\begin{bmatrix} 0&2&2&2&2 \\ 2&0&0&0&0 \\ 2&0&0&2&2 \\ 2&0&2&0&0 \\ 2&0&2&0&0 \end{bmatrix}$		922	0.011%	
68	$\begin{bmatrix} 0&2&0&0&0 \\ 2&0&0&0&0 \\ 0&0&0&2&2 \\ 0&0&2&0&2 \\ 1&0&2&2&0 \end{bmatrix}$		469	0.011%	Circle,Clique
69	$\begin{bmatrix} 0&0&0&0&2 \\ 0&0&2&2&0 \\ 0&2&0&2&0 \\ 0&2&2&0&0 \\ 3&1&1&1&0 \end{bmatrix}$		-	0.011%	Clique
70	$\begin{bmatrix} 0&2&0&0&0 \\ 2&0&0&0&0 \\ 0&0&0&2&0 \\ 0&0&2&0&0 \\ 1&1&1&1&0 \end{bmatrix}$		113400	0.01%	

Figure 73: Ratio and concentration of size 5 motifs in the multiplex network (part 10)

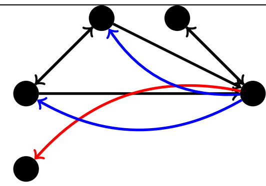
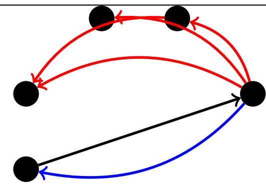
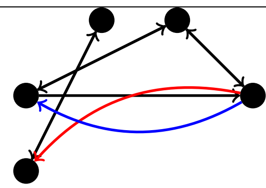
ID	Adjacency Matrix	Motif	Ratio	Concentration	Type
71	$\begin{bmatrix} 0&0&0&0&0 \\ 0&0&2&0&2 \\ 0&2&0&0&2 \\ 0&0&0&0&2 \\ 1&3&3&2&0 \end{bmatrix}$		1615	0.01%	Circle,Clique
72	$\begin{bmatrix} 0&0&0&0&2 \\ 0&0&0&0&0 \\ 0&0&0&0&0 \\ 0&1&0&0&0 \\ 3&1&1&1&0 \end{bmatrix}$		28	0.01%	Circle,Clique
73	$\begin{bmatrix} 0&0&2&0&0 \\ 0&0&0&2&2 \\ 2&0&0&0&0 \\ 0&2&0&0&2 \\ 1&3&0&2&0 \end{bmatrix}$		1571	0.01%	Circle,Clique

Figure 74: Ratio and concentration of size 5 motifs in the multiplex network (part 11)