NET Institute*

www.NETinst.org

Working Paper #04-17

October 2004

Highly Interconnected Subsystems of the Stock Market

John Idicula

Netz Informatics

* The Networks, Electronic Commerce, and Telecommunications ("NET") Institute, http://www.NETinst.org, is a non-profit institution devoted to research on network industries, electronic commerce, telecommunications, the Internet, "virtual networks" comprised of computers that share the same technical standard or operating system, and on network issues in general.

Highly Interconnected Subsystems of the Stock Market[1]

John Idicula[2]
Netz Informatics[3]

Draft revised December 13, 2004

Abstract

The stock market is a complex system that affects economic and financial activities around the world. Analysis of stock price data can improve our understanding of the past price movements of stocks. In this work, we develop a method to determine the highly interconnected subsystems of the stock market. Our method relies on a k-core decomposition scheme to analyze large networks. Our approach illustrates that the stock market is a *nearly decomposable* system which comprises hierarchic subsystems. This work also presents results from the analysis of a network derived from a large data set of stock prices. This network analysis technique is a new promising approach to analyze and classify stocks based on price interactions and to decompose the complex system embodied in the stock market.

1 Introduction

Massive data sets arise in various scientific, engineering, and commercial application domains, and they often contain information on complex systems. Some of those systems which are of great importance to people all over the world include economic and financial systems. With the help of advances in information technologies, several companies capture vast amount of data on systems and make them accessible for general public use. Analysis of such massive data sets can improve our understanding of the behavior and structure of the systems they represent. In this paper, we will undertake an effort to analyze stock market data.

Many data sets on complex systems have subsets that exhibit *harmonious* behavior and appear as clusters of densely populated regions in space. Data points of anomalously high local density are pervasive in massive data sets and their identification is a challenging of practical and theoretical interest. A common approach to identify data points of high local density in a data set is to categorize them into clusters based on notions of *proximity* or *similarity* among data points. However, in this paper we present an alternative approach using a graph algorithm.

[1]Draft research paper submitted to The NET Institute, USA on September 30, 2004.
[2]Email: idicula@eastlink.ca
[3]Box 6667, Cornwall, PE, Canada C0A 1H0

In many practical cases, it is possible to model a data set as a graph or a network. Using domain knowledge, data analysts associate special attributes of the data points with the vertices and edges of the representation graphs. Furthermore, preserving the structural attributes of the data, they can define representation graphs to suit the goals of data analysis. For example, if a pair of data points share a property, an edge is added between the vertices that represent those data points. Analysis of the resulting graph is essential to efficiently manage the data set, to understand characteristics of the data set, and to identify patterns of potential interest to a data miner.

Systems that evolve seem to gravitate towards *quasi-stable* states in which parts of their subsystems behave in concert carrying out their roles. A similar phenomenon occurs during the evolution of a graph. As a graph evolves, components form, grow, merge, and divide, and the local transformations change the structure of affected subgraphs. And, when a graph is generated or modified by iteratively adding and removing vertices and edges based on some rules, subgraphs with interesting properties would emerge.

Given a representation graph, data analysts are keen to determine its specific subgraphs. The class of dense subgraphs deserves their interest as the identification of dense subgraphs in the given graph can help to determine subsets of data points of high local density in the original data set. We can measure the density of a subgraph by the ratio of the number of its edges to its maximum possible number of edges. A subgraph with the maximum possible number of edges is called a *clique*. Determination of the *maximum clique* in a graph is a classic NP-hard problem. Moreover, even a polynomial time scheme to closely approximate the maximal clique is unlikely to exist [3]. Considering the difficulty of identifying dense subgraphs, we will investigate the problem of determining another closely related class of subgraphs, namely the *highly interconnected subgraphs*.

We loosely define a graph to be highly interconnected if its vertices are connected by multiple disjoint paths. The highly interconnected subgraphs of a given representation graph form a rich base to identify subsets of data points that directly or indirectly affect each other. They are also good targets to search for smaller dense subgraphs and cliques. Based on recent results on graphs by Pittel and Wormald [5], we will employ an efficient algorithm to identify highly interconnected subgraphs of large graphs.

The application domain for this work is the stock market. Huge amounts of data on stock market transactions are generated daily all around the world. Those transactions reflect the price movements of securities. Unlike the common approach to analyze data on stock price movements using time series analysis techniques, we apply a method based on network analysis. Literature search on the Internet showed a few references to studies that use large scale network analysis of stock market data. Some researchers study stock market data by using *distance measures* between each pair of stocks [1, 4]. They form complete graphs whose vertices represent stocks and edges have distance measures as weights. Then, they choose subgraphs whose edge weights are bounded by various threshold values to conduct detailed analysis of the stock market data. We will also follow similar steps in this work.

The main focus of this work is to identify highly interconnected subgraphs of a large graph that models the stock market. The highly interconnected subsystems of the stock market are composed of stocks that affect each other, directly or indirectly. If the edges of the input graph are restricted to those that represent similar price movements, the prices of stocks in the highly interconnected subsystem fluctuate simultaneously in a similar manner. The stocks in such a subsystem form a portfolio of stocks whose prices affect each other. Moreover, the highly interconnected subsystems determined from the complement of that input graph reflect what financial analysts usually refer to as a diversified portfolio.

Although stock price data are rich in economic information, it is impossible to predict the future trend for any particular stock. But, from a systems point of view, the price of a stock may affect prices of other stocks, and vice versa. An interaction between a pair of stocks exists if their prices affect each other. The interactions of a given stock with stocks of one group may be weak while those with stocks of another group may be strong. Thus, interactions among stocks can also cause the stock prices to fluctuate.

The stock market is a complex system of stocks that has subsystems, and we say it is a *hierarchic system*, following the seminal work of Herbert Simon on complex systems [6]. A set of given stocks can be decomposed into interacting subsystems of stocks based on analysis of stock price data. Moreover, the stock market appears to be a *nearly decomposable system*, as it has groups of stocks with interactions within and among them. Using a large data set of stock prices and analysis techniques from graph theory, we will show the existence of a hierarchy of subsystems in the stock market and establish that the stock market is a *nearly decomposable system*.

Our approach to systems analysis has four major steps. The first step is to collect data on the system. The second step is to form a representation graph to model the system using the collected data. The third step is to extract interesting subgraphs of this graph. Finally, in the fourth step we decompose the graph (and/or its subgraphs) to aid the analysis. This approach is useful when the analysis concerns system entities, their relationships, and their individual and collective behaviors.

In this paper, we present an analysis of a graph generated from stock market data, using our systems analysis approach. Viewing the stock market as a complex system, we will seek to identify highly interconnected subsystems of the stock market. Moreover, we will determine a hierarchy among subsystems of the stock market, and show evidence that the stock market is a *nearly decomposable system*. We will also present some results related to the stock graph we study in this work. The remainder of this paper is organized as follows. Section 2 gives a few definitions. The stock graph we study in this paper is described in Section 3. The network analysis technique in our research method is reviewed in Section 4. In Section 5, we discuss the structure of the stock market. Finally, we present our conclusions in Section 6.

2 Preliminaries

Let us define some terminologies before we proceed. They are given here to explain how we construct the representation graph to model the stock market.

Definition 1 Stock Graph

A *stock graph* $G = (V, E, W)$ consists of a set V of n vertices, each of which represents a stock, a set E of m edges connecting the vertices, where $m = \frac{n(n-1)}{2}$, and a weight function $W\colon E \to \mathbb{R}$ defined on E to assign a weight to each edge. We will analyze subgraphs of the stock graph with bounded edge weights, and we will use θ to denote edge weights.

Definition 2 Stock Price

We define *stock price* P as the daily closing price of a stock. We will use $P_i(t)$ to denote the closing price of stock i on day t.

Definition 3 Stock Yield

The *stock yield* Y is the natural logarithm of the daily return of a stock. We will use $Y_i(t)$ to denote the yield of stock i on day t, and it is calculated as follows:

$$Y_i(t) = \ln P_i(t) - \ln P_i(t-1)$$

Definition 4 Edge Weight Function

The value of the weight function W_{ij}, the weight of the edge between vertices i and j, is the cross-correlation between the yields of the pair of stocks i and j. It is calculated as follows:

$$W_{ij} = \frac{\langle Y_i Y_j \rangle - \langle Y_i \rangle \langle Y_j \rangle}{\sqrt{(\langle Y_i^2 \rangle - \langle Y_i \rangle^2)(\langle Y_j^2 \rangle - \langle Y_j \rangle^2)}}$$

The averages in the formula above are calculated over the period covered by the analysis.

3 Stock Graph

The stock graph can model the stock market as a dynamic system. However, as the empirical part of this work is limited to a snapshot of the stock market taken from a fixed period, the massive network we analyze is static. The edges of the stock graph we analyze represents price interactions among selected stocks traded on three major stock exchanges in the USA, namely, AMEX, NASDAQ, and NYSE.[4] The initial data set for this study consists of daily closing prices for these stocks during a four year period. Using this data set, we constructed the stock graph to study stock price interactions and to determine highly interconnected subsystems in the stock market.

3.1 Construction

We constructed our stock graph from a list of 7,670 stocks and their daily closing prices from January 2000 to December 2003. A vertex in the stock graph represents each stock in the list. As there may be a price interaction between each pair of stocks, we added an edge in the stock graph to represent each potential price interaction. Using the edge weight function, we assigned each edge a weight that measures the cross-correlation coefficient of the yields of the stocks represented by its end vertices. Hence, our stock graph has 7,670 vertices and 29,410,615 weighted edges.

The stock price data set we use is part of a massive volume of data downloaded from a popular Internet web site which shares stock market information. Some data points were missing in the downloaded data, and we inserted values in the place of the missing data points.

Missing values can be due to stocks that did not begin trading until some later day in the period. For such a stock, we inserted the closing price of its first trading day in the place of missing data for all the days before its first trading day.

Several stocks in our list had some missing data points for dates after their first trading days in the chosen period. In such a case, we inserted the closing price of the most recent trading day before the days of missing closing prices. For example, consider that the closing prices of stock X on October 14 and October 15 are missing, but we have its closing price for October 13. In the place of closing prices for October 14 and October 15 for stock X, we inserted its closing price on October 13.

From the modified stock price data, we calculated the stock yields first, and then computed the cross-correlations between the yields of every pair of stocks to assign weights for the edges of our stock graph.

3.2 Distribution of Edge Weights

The edge weights of our stock graph have a range from -1.0 to 1.0 as defined in the edge weight function. To understand the distribution of the edge weights in our data, we have grouped the edges of the stock graph into different intervals of the edge weight, θ, as shown in Table 1.

[4] The number of stocks selected for this work was limited by the size of the random access memory of the computer that we used to analyze the network.

Table 1: Edge Weight Frequency Distribution.

Edge Weight, θ	Number of Edges
$\theta =$ -1	0
-1.0 $\leq \theta <$ -0.9	5
-0.9 $\leq \theta <$ -0.8	18
-0.8 $\leq \theta <$ -0.7	60
-0.7 $\leq \theta <$ -0.6	83
-0.6 $\leq \theta <$ -0.5	136
-0.5 $\leq \theta <$ -0.4	523
-0.4 $\leq \theta <$ -0.3	2,977
-0.3 $\leq \theta <$ -0.2	14,808
-0.2 $\leq \theta <$ -0.1	84,986
-0.1 $\leq \theta <$ 0.0	7,220,708
$\theta = 0$	658
0.0 $< \theta <$ 0.1	16,758,731
0.1 $\leq \theta <$ 0.2	3,958,880
0.2 $\leq \theta <$ 0.3	1,035,959
0.3 $\leq \theta <$ 0.4	245,411
0.4 $\leq \theta <$ 0.5	63,065
0.5 $\leq \theta <$ 0.6	17,013
0.6 $\leq \theta <$ 0.7	4,658
0.7 $\leq \theta <$ 0.8	1,392
0.8 $\leq \theta <$ 0.9	444
0.9 $\leq \theta <$ 1.0	100
$\theta = 1$	0
-1.0 $\leq \theta \leq$ 1.0	**29,410,615**

The edge set of our stock graph contains only five edges with weights below -0.9, but it has 100 edges with weights greater than or equal to 0.9. Furthermore, 658 edges have weight equal to zero. Therefore, the number of pairs of stocks with extremely low or high, or no price interaction is much smaller than the number of the other pairs of stocks. Over 99% of the edges have weights in the interval [-0.2, 0.5). The proportion of edges with weights in the interval [-0.1, 0.1) is quite high, and it decreases quickly as the value of θ moves away from zero.

A rough graphical representation of the proportion of edges plotted against edge weight intervals is shown in Figure 1. In this figure, we can see that the proportion of edges with positive weights exceeds the number of edges with negative weights. Figure 1 and Table 1 show how the number of edges are distributed for values of θ around zero and at the tails of the curve. In the next two sections we will explore a few subgraphs of the stock graph, generated from edges with different values of θ.

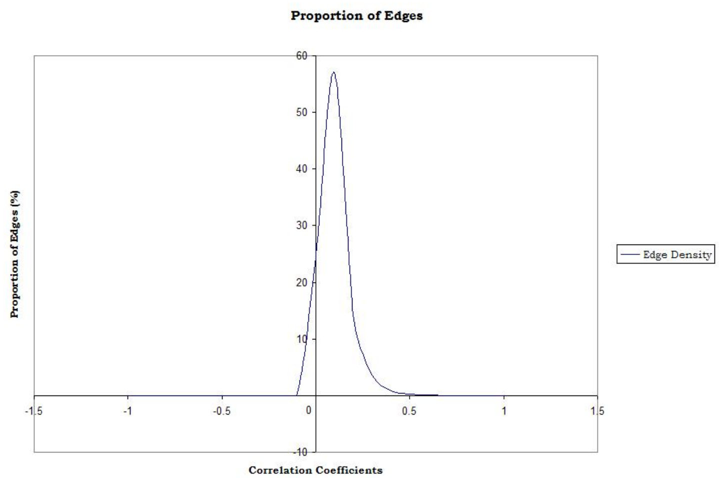

Figure 1: Edge Density.

4 Network Analysis

The topic of network analysis receives great attention because of theoretical interests and its practical significance [1, 2, 3, 4, 5]. An interested reader can find a wealth of related literature from the aforementioned references to appreciate the research efforts in a wide variety of areas such as biology, ecology, economics, epidemiology, finance, internetworking, and of course, random graphs. Fascinating results from experiments conducted to study structural properties of networks constructed from stock market data are given in the papers by the Mantegna and Pardalos groups [1, 2, 4].

The application of network analysis techniques to study stock price data may reveal new information that will improve our understanding of the stock market. In this work, we apply a technique to identify highly interconnected subsystems of the stock market, and we determine them by an analysis of the edges in the stock graph.

4.1 k-Cores

An algorithm that exploits connectivity properties of a graph lies at the heart of the network analysis component of this work. It partitions vertices and edges, based on their *connectedness*. The way it works is by determining all the k-cores of the graph. During the determination of k-cores, all the vertices are assigned *core numbers* and the vertices are partitioned. As the vertices get core numbers, the edges fall into their partitions based on the core numbers of their end vertices.

Determination of *good* partitions of vertices and edges in a graph is, in general, a hard problem [3]. The algorithm we use in our work is one of a few known efficient tools that are applicable to the analysis of large networks. The vertex and edge partitions that result from the application of this algorithm to the subgraphs of the stock graph will help us to understand the structure of stock market, its subsystems, and determine the membership of each stock in its subsystems.

The *k-core* of a graph is its maximal subgraph with vertices of degree at least k. The *core number* of a vertex is the maximum k for which there exists a k-core containing that vertex. The $(k+1)$-core of a graph, if it exists, is contained in its k-core.

Determining the k-core of a given graph is easy. All that needs to be done is to repeatedly peel layers of vertices from that graph until we have the k-core. During each iteration, the vertices of degree less than k are deleted. When we delete a vertex, we also remove all of its incident edges. This *peeling* process ends when all the remaining vertices have degree at least k. The vertices and edges that survive form the k-core of the given graph.

4.2 Highly Interconnected Subsystems

Pittel and Wormald [5] have recently shown that the k-core of a graph is highly likely to be k-connected when k as well as the numbers of vertices and edges in the graph are large. A graph is said to be k-connected if there are k disjoint paths connecting any pair of its vertices. Therefore, the k-cores of networks that model complex systems represent highly interconnected subsystems of those systems.

Let us consider a graph whose k-core and $(k+1)$-core are non-empty, for a fixed k. Assume that $(k+1)$-core is a proper subset of k-core. Using the peeling process described earlier, let us determine the k-core of this graph first, and then its $(k+1)$-core. Denote the vertices with core number k by set X and the vertices with core number $(k+1)$ by set Y. By definition of the k-core, the vertices of X and Y are present in the $k-$core, but the vertices of X are absent in the $(k+1)$-core. The interactions between the sets X and Y occur through the edges of the original graph with end vertices in both X and Y, and they are removed during the peeling process of determining the $(k+1)$-core from the k-core, as the vertices of X are deleted. In contrast, the interactions internal to the k-core are due to the edges whose end vertices are in X or Y. Similarly, the interactions internal to the $(k+1)$-core are solely due to the edges whose end vertices are in Y. Furthermore, it is the number of internal interactions incident with a vertex that contributes to the degree of that vertex in the k-core, and hence, to its core number. When k is large, the internal interactions of a k-core define highly interconnected subsystems.

4.3 Interactions with Bounded Weights

Armed with information on the distribution of edge weights, we extracted subgraphs with bounded edge weights to analyze the interactions of the stock graph and we found some interesting preliminary results. Boginsky et al. [1] note that in large subgraphs of the stock graph constructed from restricted intervals of θ, the degree distribution of vertices does not reveal any well-defined structure. They also report that graphs with low edge density have a pattern in the degree distribution. We can validate their finding in a small subgraph of our stock graph.

4.3.1 Degree Distribution

We inspected the degree distribution in a small graph formed with edges of the stock graph with bounded weights. With a small proportion ($< 1\%$) of edges in the stock graph, the subgraph with $\theta \geq 0.5$ was chosen for analysis. This subgraph has 1,709 vertices and 23,607 edges. The distribution of degrees of the vertices is presented in Table 2. Interestingly, 1,012 of its 1,709 vertices ($> 59\%$) have degrees in the interval $[1, \ 10]$, and the highest degree is 419. After classifying the vertex degrees for this small graph into class intervals of size 50, the frequency distribution appears almost like a straight line in the log-log scale.

Table 2: Distribution of Degrees in the Subgraph with $\theta \geq 0.5$.

Degrees	Frequency
1 - 10	1012
11 - 50	413
51 - 100	168
101 - 150	58
151 - 200	22
201 - 250	11
251 - 300	12
301 - 350	9
351 - 400	3
401 - 419	1

4.3.2 Core Number Distribution

We determined the core numbers for each vertex in the subgraph with $\theta \geq 0.5$, and they fall in the interval $[1, \ 71]$. The frequency of vertices for each core number is given in Table 3 and it varies from 0 to 435. The 25 vertices with the highest degrees among the 81 vertices in the 71-core of this subgraph are given in Table 4. These

Table 3: Frequency of Core Numbers in the Subgraph with $\theta \geq 0.5$.

CN	Frequency	CN	Frequency	CN	Frequency	CN	Frequency
1	435	19	10	37	4	55	2
2	185	20	34	38	3	56	8
3	101	21	15	39	7	57	5
4	98	22	9	40	0	58	59
5	46	23	5	41	4	59	0
6	35	24	7	42	3	60	4
7	34	25	9	43	3	61	4
8	67	26	32	44	2	62	0
9	40	27	10	45	0	63	1
10	27	28	6	46	6	64	1
11	26	29	8	47	4	65	2
12	7	30	10	48	2	66	1
13	16	31	4	49	34	67	3
14	31	32	5	50	4	68	11
15	6	33	9	51	7	69	1
16	10	34	9	52	7	70	0
17	10	35	56	53	5	71	81
18	9	36	3	54	7		

Table 4: 25 stocks with highest degrees in 71-core of the Subgraph with $\theta \geq 0.5$.

Stock	Degree	Stock	Degree	Stock	Degree	Stock	Degree	Stock	Degree
ASML	187	IJK	266	IWF	284	IYY	340	SMH	209
BDH	207	IJR	302	IWM	283	MDY	419	SPY	399
IAH	271	IVV	337	IWO	282	MTK	216	STM	192
IIH	204	IVW	254	IWV	358	PHG	200	SWH	197
IJH	343	IWB	331	IYW	280	QQQ	289	XLK	357

vertices have degree values from 187 to 419, and the stocks that they represent include some *heavy weight* items such as the NASDAQ-100 tracking stock, QQQ.

4.4 Components in the k-Core

Connectedness of k-cores cannot be guaranteed; that is, a k-core may have more than one component. As the k-core is a maximal subgraph of the original graph that satisfies the degree constraint, the connected components of the k-core are also maximal and they too satisfy the degree constraint. We will show the drawing of an example of a disconnected k-core whose components are highly interconnected.

In the analysis of the subgraph with $\theta \geq 0.9$ we observed that its 5-core is disconnected. This subgraph has 77 vertices and 100 edges. Its edges represent the strongest 100 interactions in our stock graph. The core numbers of the vertices in this subgraph vary from 1 to 5. The 5-core of this graph, shown in Figure 2, has two components, 12 vertices, and 35 edges. Each component in the 5-core is highly connected.

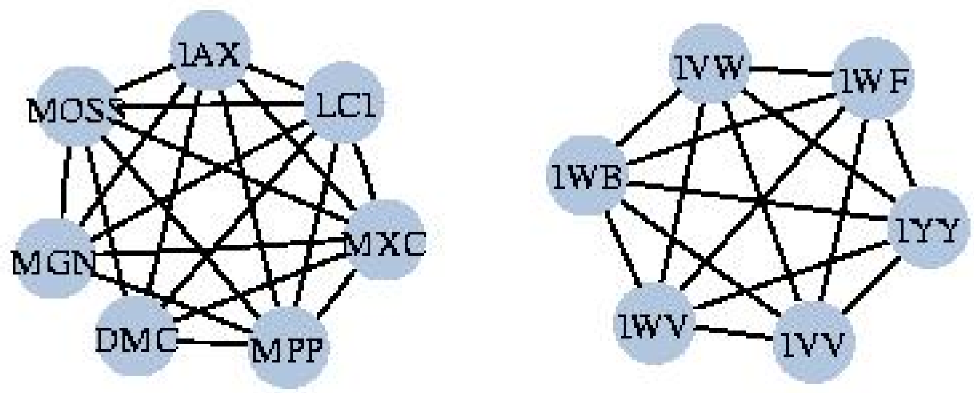

Figure 2: 5-core of the Subgraph with $\theta \geq 0.9$.

5 Structure of the Stock Market

The stock market is a highly dynamic system, with a multitude of factors that influence its behavior. Although our model of the stock market is quite simplistic, it serves to improve our understanding of the stock market behavior. In our model, the stocks are the system parts, and their historic price movements help us to study the past interactions between each pair of stocks.

Applying our network analysis technique, we can identify hierarchies in the stock market due to past price movements of the stocks and show that the stock market is indeed a hierarchic system. Using our research method, we also establish the existence of subsystems formed by the stocks due to their price interactions. Furthermore, the presence of interactions among these subsystems show evidence of the *near decomposability* of the stock market.

5.1 Hierarchy in the Stock Market

Identification of hierarchies among the stocks may reveal new information on the evolution of the stock market. Applying our network decomposition scheme to proper subgraphs of the stock graph formed by edges with bounded weights, we will show the existence of a hierarchy among the stocks.

During the k-core decomposition of a graph, its vertices are partitioned as they are assigned core numbers. In this scheme, the vertices that share a core number belong to the same partition. Thus, we have a natural hierarchic decomposition of the graph based on the vertex core numbers. The vertices with the highest core number are the most dominant vertices in this hierarchy, and the dominance of the other vertices decrease with their core numbers.

The vertices with the highest core number k and the edges that belong to the innermost core of the system are the most dominant ones in this hierarchy. Let k' be the next core number less than k. The set of the vertices with the core number k' and the set of edges whose end vertices have core number k' or k share the next level of dominance. Similarly, each vertex and edge gets a proper assignment in this hierarchy.

We can describe a vertex layout of this hierarchy using a set of concentric layers. The innermost layer in this hierarchy contains the set of vertices with the highest core number k. Each subsequent layer contains the set of vertices with the next core number. The vertices with the least core number are in the outermost layer.

The k-core decomposition scheme can be used to identify hierarchies in complex systems such as the stock market. The assignment of core numbers to vertices of massive graphs during k-core decomposition results in the identification of the hierarchies in the systems they represent. In Section 5.3, we will discuss the identification of a hierarchy from a large subgraph of the stock graph using the k-core decomposition scheme, with a slight modification.

5.2 Near Decomposability of the Stock Market

Identification of subsystems in the stock market may reveal new information on the behavior of the stock market. As we described earlier, the k-core decomposition of a graph results in a partition of the vertices. From the vertices in each partition, we can identify subgraphs of the input graph induced by those vertices, and then we can determine subsystems of the stock market. The connected components of these subgraphs represent the subsystems that are worthy of the attention of quantitative analysts and fund managers.

The interactions internal to a subsystem represent the price interactions among the stocks represented in that subsystem. Recall that the vertices in a subsystem have the same core number. If the core number of the vertices in a subsystem S is

k_S, then each vertex in S has at least k_S interactions with the other vertices in S. The connectivity of the system S is likely to be very high, as shown by Pittel and Wormald [5]. Let us qualify each each interaction internal to a subsystem as *strong*.

The external interactions of a subsystem represent its interactions with vertices of different core numbers, and they are the interactions among subsystems. Let us qualify each of those interactions as *weak*. With the presence of the strong interactions within the subsystems and the weak interactions among the subsystems, we establish the *near decomposability* of the stock market.

5.3 Innermost k-Core

We chose a large subgraph that has all the vertices of the stock graph to understand the hierarchy in the stock market. This subgraph was formed by edges from the stock graph with weights $|\theta| \leq 0.1$. It has 7,670 vertices and 23,980,157 edges.

Instead of following the straightforward k-core decomposition approach, we apply a slight modification. In the modified scheme, we iteratively determine and prune the innermost[5] k-core of the subgraph. During each iteration, we first determine the innermost k-core. Then we prune the vertices in the innermost k-core and their incident edges. This process stops when all the edges in the subgraph vanish.

Table 5: Changes to the Innermost Core During the Evolution of the Subgraph with $|\theta| \leq 0.1$.

Iteration Stage	Core Number[1]	Core Vertices [2]	Core Edges[3]	Connections to Core[4]	Subgraph Edges[5]
1	4,820	5,654	15,172,796	8,717,853	23,980,157
2	89	423	26,373	52,250	89,508
3	17	372	5,313	4,775	10,885
4	3	165	397	243	797
5	1	214	157	0	157

1 - Highest core number
2 - Number of vertices in the innermost core
3 - Number of edges in the innermost core
4 - Number of edges between the innermost core and the other vertices
5 - Number of edges in the subgraph

We applied the modified decomposition algorithm to determine a hierarchy in the subgraph described above. Changes to the innermost core of the subgraph as it evolves through the iteration stages are shown in Table 5.

[5]The k-core for which k is maximum.

The decomposition process took only five iterations. The innermost core of this subgraph is a *giant 4,820-core!* It has 5,654 vertices and 15,172,796 edges. The percentage of the number of vertices in the 4,820-core is greater than 73% and that of the number of edges is greater than 63%. Moreover, the innermost core has 8,717,853 connections to the other subsystems represented in this subgraph. The number of edges incident with the vertices in the 4,820-core forms a very large proportion (> 99%) of the subgraph edges. After the deletion of the 4,820-core and its incident edges in Stage 1, only 2,016 vertices and 89,508 edges remain in the subgraph. Another notable aspect of this table is that all the numbers related to vertices and edges in the table quickly decrease as the subgraph evolves through the iterations, with the exception of the number of vertices in 1-core in Stage 5.

The most dominant vertices in the identified hierarchy are the vertices of the 4,820-core of Stage 1. The next level of dominance belongs to the vertices of the 89-core of Stage 2. Similarly, we can assign decreasing levels of dominance to vertices of innermost core of each iteration stage. After the deletion of all the edges, some vertices with degree zero remain in the subgraph. As they remain with core number 0 after Stage 5, let us arbitrarily assign them the least dominance in this hierarchy.

6 Conclusion

In this paper, we describe a research method to analyze massive data sets on complex systems and to improve our understanding of the structure and behavior of such systems. We summarize our approach in four steps: 1) collect data on the system, 2) model the system as a weighted graph, 3) identify subgraphs, and 4) perform k-core decomposition on selected subgraphs to aid the analysis. We use this approach to identify highly interconnected subsystems of the stock market from an analysis of stock price data.

The stock market affects the common people. Massive amounts of stock price data are generated daily, and they are available for general public use. Modeling the stock market as a weighted graph generated from historic stock price data, we apply a network analysis technique to improve our understanding of the past price movements of stocks.

In addition to the brief description of our research method to study systems, this work contributes the following:

• First, we describe a new approach to analyze stock market data, and gave preliminary results from our study.

• Second, we present a method to identify highly interconnected subsystems of the stock market. After generating subgraphs of interest based on edge weights, we assign core numbers to vertices using the k-core decomposition scheme. For large values of k, the connected components of the k-cores represent the highly interconnected subsystems of the stock market.

• Third, we present a hierarchy among the stocks. The k-core decomposition scheme specifies the position of each stock in this hierarchy, as the core numbers define a natural hierarchic partitioning. Thus, we show that the stock market is a hierarchic system.

• Finally, we also show evidence that the stock market is a *nearly decomposable* system by establishing the presence of the highly interconnected subsystems formed by the stocks as well as the interactions among those subsystems.

The preliminary results obtained from our analysis encourage us to classify stocks into groups based on different levels of price interactions. In addition to being an approach to analyze stock market data, the research method of this paper serves to improve our understanding of the structure and behavior of the stock market. We anticipate refinements to this approach and applications of this method in the financial industry.

7 Acknowledgement

Partial financial support from the NET Institute (http://www.NETinst.org) for this research is gratefully acknowledged.

References

[1] V. Boginski, S. Butenko, and P.M. Pardalos, On structural properties of the market graph. In A. Nagurney, editor, Innovations in Financial and Economic Networks, Edward Elgar Publishers, London, 2003, 29-45.

[2] G. Bonanno, G. Caldarelli, F. Lillo, S. Micciché, N. Vandewalle, and R. N. Mantegna, The European Physical Journal B, 38 (2004), 363-371.

[3] Håstad, J. Clique Is Hard to Approximate Within $n^{1-\epsilon}$, Acta Mathematica 182 (1999), 105-142.

[4] R. N. Mantegna, Information and hierarchical structure in financial markets, Computer Physics Communications, 121-122 (1999), 153-156.

[5] B. Pittel and N. C. Wormald, Asymptotic enumeration of sparse graphs with a minimum degree constraint. Combinatorial Theory, Series A, 101 (2003), 249-263.

[6] Simon, H. A. *The Sciences of the Artificial*, MIT Press, Cambridge, MA, 2nd edition (1981).