University of Jyväskylä

Modeling Concept Maps Done by Physics Students

Author:
Toni Purontaka

Supervisors:
Ismo T. Koponen
Juha Merikoski

October 22, 2013

Abstract

Understanding the structure of scientific knowledge is to large degree based on understanding what the key concepts are and how and why these concepts are connected. Recent cognitively oriented research on learning has suggested that procedures of knowledge construction and processing which generate the complex knowledge structures may be simple ones, reducible to simple basic patterns. In this study we concentrate on modeling concept maps that are produced by physics students. We study seven different models to describe physics students knowledge structure utilizing the basic patterns as modeling parameters. The models ranged from straightforward calculation of triangles and edges to more complex models that take in account the transitivity as well as cycles and triangles. According to the research documented in this paper it seems possible that at least highly hierarchical information in learner's mind can be modeled by triangles and other simple patterns formed by the key concepts and meaningful links between them.

Contents

1 Introduction

The knowledge in physics has been shown to be highly structural by nature [1]. This means that new concepts - whether they describe a physical phenomena, more general principle or a physical law are connected to previously defined concepts. These connections are not loosely or arbitrarily defined. Two concepts can be seen connected if there exists either a direct theoretical or an empirical connection between the new concept and the body of knowledge already defined. Physical phenomena can thus be described by concepts and relations connecting them [2]. Also, knowledge processing and acquisition has recently been described in the framework of network theory, and such an approach seems to be well adapted for description of relational aspects of conceptual knowledge and aspects related to retrieval of knowledge [3, 4]. These are the main reasons why there has been interest in developing learning and teaching theories that capitalize on concept maps [5].

Understanding the structure of scientific knowledge is to large degree based on understanding what the key concepts are and how and why these concepts are connected. The structural relations between the concepts play also an essential role in establishing the meaning of concepts [6, 7]. Structure also affects how concepts are introduced in teaching scientific knowledge, how concepts are acquired through teaching, and how the conceptual knowledge can be represented and transferred forward [8, 9, 10]. Furthermore, recent cognitively oriented research on learning has suggested that procedures of knowledge construction and processing which generate the complex knowledge structures may be simple ones, reducible to simple basic patterns. Of particular importance seems to be different types of hierarchies, cliques, transitive patterns and cycles [11, 12]. These notions have encouraged the idea that such patterns may help in understanding the cognitive processes behind knowledge construction and may lead also to the development of computational models for cognitive processes [11, 12, 13].

Whenever a new concept is introduced to the body of knowledge in physics it links to already existing concepts. These connections are needed to justify the introduction of the new concept. An arbitrary (and legitimate) link between two concepts has it's justification through a theoretical model or an experiment. In both cases there are distinct sets of concepts of the established body of knowledge that are connected to the newly introduced concept.

It is the hierarchical nature of knowledge in physics that makes the use of concept maps such a useful tool [14]. In this study we follow Novak's definition for a concept map [8]. Concept maps consists of a set of concepts (also called nodes in graph theoretical applications) and links (or edges) connecting them. Each link has a label describing the meaning of the conncetion. This definition leads us to note that concept maps are graphs with added labels. Graphs are commonly definited as combination of a set of nodes and a set of edges. In the most common case, we can write $G = (V, E)$, where G is the set of graphs, V is the set of nodes and E is the set of edges.

We see that concept maps are a justified approach to model knowledge in physics. There is also another useful feature of using concept maps as the tool to model knowledge. In physics there are often many different paths to introduce a new concept. In planning of teaching the order of concepts can vary and different strategies can still be equally effective. That can be illustrated easily by concept maps as there are usually many paths linking node A to node B in a well-connected map. Concept maps can serve teachers in building their own understanding of the structure of the knowledge (of a topic of interest) as well as help them plan teaching in a way that introduces knowledge in a coherent way.

As concept maps are essentially graphs the study of them can leverage of the recent graph theoretical advancements. The interest in the networks and their properties has surged over the past two decades. A lot of effort has gone into studies of real-world graphs such as various kinds of social networks, the Internet and world wide web as well as into creation of analysis algorithms and models to predict and explain the behavior of the real world system [15]. There is a sophisticated mathematical system that provides powerful tools to analyze complex networks. This system is called *exponential random graphs* (ERG). The foundations of the theory were laid over thirty years ago by Holland and Leinhardt [16] although it took some time before the scientific community at large understood the power of ERGs. They were developed after it became widely accepted fact that mere random graphs were not good models for real-world networks. After their introduction the exponential random graphs turned out to be highly effective in handling of complex networks. Exponential random graphs have a solid mathematical foundation, and equally importantly, they can also be derived from equilibrium statistical mechanics [17]. This aspect is illustrated in an article by Park and Newman as in it they arrive at analytical solution for the 2-star model by utilizing statistical mechanics theory [18]. The 2-star model is the most simple, yet not trivial exponential random graph model. They derive the model from maximum entropy assumptions. This result shows how solid is the mathematical foundation under the ERGMs.

The rise of computing power enabled researchers to calculate more robust variables and run more sophisticated algorithms on larger (and usually more realistic) networks. In recent years the wide spread use of computers across the fields of research has lead to automation of data collection. This has enabled scientists to work on large and diverse real world networks.

1.1 Modeling parameters

In this study we concentrate on modeling concept maps that are produced by physics students. Our aim is to calculate certain characteristic values of the concept maps and use these as modeling parameters, or in some cases boundary conditions. We hypothesize that by carefully choosing the parameters can yield better models that describe the physical substance better than random graphs. Modeling is carried out using the Statnet package of R [19]. It provides a well-established framework for modeling concept maps. Statnet and it's features will be discussed in greater detail in chapter 4.

It is obvious that some simple modeling parameters like the number of edges in the graph are insufficient alone. The problem is that they don't capture the hierarchical structure of physical knowledge [20]. There are more advanced modeling parameters available, such as clustering or k-stars that capture the topology of the graphs better. In the context of this study we have a constant number of nodes since the concepts used by physics students to construct their concept maps were predetermined. It is reasonable to limit the modeling process to networks with (the same) constant number of nodes. There are also other similar boundary conditions that can be applied into modeling such as forcing at least one edge for all nodes or disallowing self-connections.

All the models will be simulated in order to get insight whether they produce networks that can be interpreted as realistic concept maps that are distinguishable from random graphs or not. We consider a graph realistic when the calculated values of the statistical variables are close to the values calculated of human-made graphs. As statistical variables to be measured we chose the number of edges, triangles, k-stars, n-cycles and transitivity. In order for a simulated network to be realistic it should imitate the real-world networks. It is a well known fact that mere random graphs don't have realistic link distributions as their link grades follow Poisson rather than power-law distribution [21]. It became clear that many times poor models yield to networks with unrealistic number of links for a node. We consider simulated networks that facilitate nodes with zero links unrealistic as there are no isolate concepts in a developed physics theory such as electromagnetism (which is the theme of the concept maps in this study).

1.2 Research questions

It has been shown that the knowledge processing and acquisition happens through set a basic motifs (or patterns) [2]. The goal of this study is to take a concentrated look at these motifs and use them then as building blocks to create models. These models are then tested by simulating them. Similar measurements to original, real world concept maps are then done to these simulated networks to see how much, if any of the hierarchical and topological features are sustained. This goal can be compressed into two research questions:

1. Can the maps made by the physics students[1] be modeled using the basic patterns so that the simulated graphs preserve some of the topological information found in the original concept maps?

2. What is the set of basic patterns that, when used as modeling parameters, yield to most accurate model?

The ideas discussed in this paper can be applied into a variety of areas of interest. One of the most promising is to use the methods introduced here to help teachers plan their teaching. By automating parts of the analysis done here a (web-based) software could be developed to quickly give a teacher a good general view of the level of understanding of the students, especially related to their ability to understand relatedness of key concepts in the subject matter. In science, that kind of information about the topology of the knowledge structure could play an important role in addition to the information gain from regular exams.

[1] A computer-drawn reconstructions of two original concept maps can be seen in figures 3 and 4.

2 Modeling and simulation of concept maps

In the heart of modeling concept maps are the topological features. We will begin by looking at the basic motifs based on the recent advancements in cognitive sciences [22]. We will continue on to discuss the development of *exponential random graphs* and the *Markov chain Monte Carlo* method.

2.1 Structure and basic motifs

New concepts connect to the established body of knowledge in the context of experiments and theoretical models. Experiments and models therefore produce the basic structural patterns found in complex networks of structured scientific knowledge [23, 2]. In (laboratory) experiments the concepts are connected through designing the experiment and through interpretation of the experimental data. In quantitative experiments (usually a small) number of concepts are used to make a concept measurable. In the simplest case this leads to triangular pattern. Triangle is formed when connected concepts A and B are used to make concept C measurable. This forms connections A → C and B → C closing the triangle (since there already exits a connection A → B). This is illustrated in the figure 1. As an example we can take a look at a well known experiments on photoelectric effect. One of the key conclusions of the experiments was that above the threshold frequency, the maximum kinetic energy of the emitted photoelectron depends on the frequency (but not on the intensity) of the incident light [24]. Intensity and frequency are connected in the theory of light. This experiment connects intensity and (threshold) frequency to the energy of the emitted photons.

There are cases where three or more concepts are explicitly needed to make another concept measurable. The graph theoretical patterns that are then formed are called *k-stars*, where k is the number of concepts that are connected to the concept that is wanted to make measurable. In figure 1 (b) there is a 3-star motif which is a result of three nodes all linked to a fourth node. Important patterns that arise from similar arguments are *small cycles*. As an example an (undirected) 4-cycle can be a result of the following procedure. A concept A can be used in defining two other concepts B and C. These concepts are then used to define another concept D. As a result we have a small cycle with four concepts. The definition of D requires explicitly concepts B and C and implicitly concept A. A *twopath* connects concepts A and C via concept B. This is illustrated in the figure 2. *Transitivity* is a motif

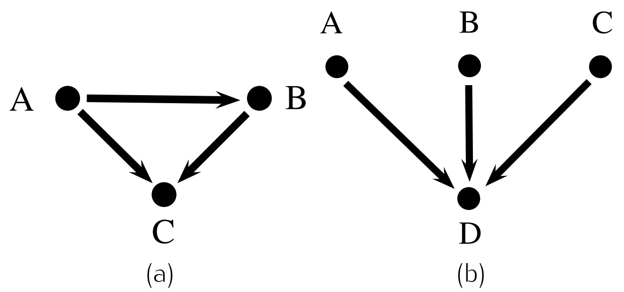


Figure 1: (a) A triangle, the basic motif of interest (b) A 3-star, an example of the k-star class of network motifs.

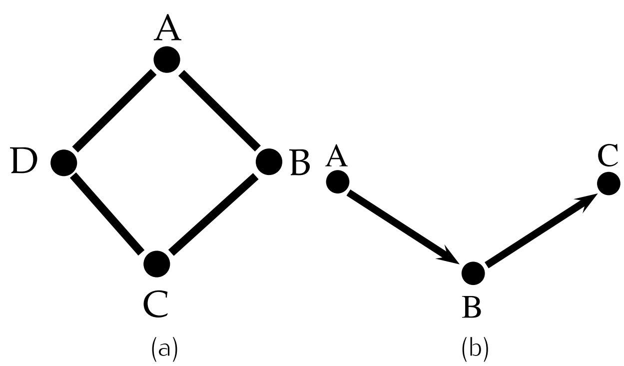


Figure 2: (a) A 4-cycle, an example of the cycle class of network motifs (b) A transitivity motif. We consider a pattern to be a transitivity motif when there is a connection from node A to node C via a third node B.

where nodes A and C are linked via a third node B. An example of this can be seen in figure 2.

The same basic patterns or motifs are also found in the context of theoretical models. This is due to the requirement of the conceptual coherence of a theoretical model. A Conceptually coherent model integrates various concepts into a solid system that describes real phenomena. This leads to the basic knowledge-ordering patterns of mutual dependencies between concepts [23]. The same basic patterns discussed above are then expected to be found in conceptual systems that describe well the physical knowledge, regardless of the context where the connections between concepts are formed.

2.2 From random graphs to ERGMs

Random graphs are products of some random procedure. In the most straightforward case consider a set of nodes n that forms edges with the probability p. The resulting graph is a random graph with fixed number of nodes. This is the Erdös-Rényi (ER) random graph model[2] [26]. The probability distribution p_k of this kind of random graph model is given by [27]

$$p_k = \binom{N}{k} p^k (1-p)^{N-k} \tag{1}$$

where N is the number of vertices in the graph and k is the degree of the vertex. The degree of the vertex means the number of connections it has to other vertices.

Random graphs have been deployed in many areas of interest. In addition to their obvious use in mathematics and computer science they have also been used in sociology and epidemiology [28] among other. Because of the discrete nature of the network and mutually independent probability to form edges between any nodes the grade distribution of the random graph is a Poisson distribution [21]. It was soon discovered however that the random graphs did not accurately predict or explain the behavior of neural networks, networks of friendships, the internet or other real-world phenomena. These real-world networks very commonly have heavy tails in their grade distribution. Heavy tails indicate nodes with remarkably high grades. These nodes are commonly called hubs and they are found throughout the real-world networks. Heavy tails do not fit into the Poisson distribution but are rather an indication of the power law distribution. It is common that in many networks the grades of hubs are several orders of magnitude higher than average grade of the network. The other major downfall of the ER model (as defined originally) is that the random graphs are undirected. This means that information between two nodes that are connected can travel in both directions. This is again not true for many practical situations or applications. Consider for example a network where nodes are web sites and links are hyperlinks in the web sites. These links are clearly directed. For these reasons it imperative for researchers to move beyond the ER random graphs. [27]

Network theory has seen development into multiple directions after the introduction of ER models. Many new models have been proposed such as

[2]An equivalent model was developed independently at the same time by Gilbert [25]

the small-world model [29]. The name of the model comes from the famous proposition that any two people are connected through no more than six other people. The idea behind the small-world model is to consider the nodes forming a one-dimensional lattice with N nodes. We then allow the links to be formed between nodes that are not more than k nodes apart. The constant $k < N$ is a threshold. In the next phase some long distance links are formed. This completes the small-world model. This model has the advantage of high transitivity over the original ER models. The small-world model is an example of many other models studied, such as configuration models. [30]

Another important new model was the scale-free model by Barabási and Albert [31]. They propose in their paper that the many complex networks such as the world wide web follow the scale-free power-law as the link grade distribution. The power-law for probability $P(k)$ can written for a large k as

$$P(k) \sim k^{-\gamma} \tag{2}$$

The model is based on two observations of the real world networks that are absent in the ER model. Those are growth and preferential attachment. The growth means that real-world networks rarely stay stationary with respect to their nodes. Many complex networks are open so that new nodes are introduced to the network all the time. The preferential attachment comes from the notion that there are hubs in most of the real-world networks. In the world wide web this means that an arbitrary web page is more likely to have links to popular web sites than non-popular. These two features lead to the scale-free power-law model.

It became clear that these models that are analytically solvable all fall short in one way or the other. Researchers then turned their attention to a different approach. In particular the transitivity can only be included in very few solvable models. The exponential random graph model was developed using the analogy of the Boltzmann ensemble. To define exponential random graphs we follow the presentation of Strauss [32, 30]. We begin by considering the measurable properties of the graph $\{\epsilon_i\}$. These include in our study such variables as number of edges, clustering coefficient, communicability betweenness centrality and transitivity. These variables are discussed in more detail in chapter 2.4. As these properties are related to the connections between the nodes we can use the analogy of the energy function in statistical mechanics. Let the set $\{\beta_i\}$ be the field parameters. Exponential random graph model is the set of all possible graphs G. The probability distribution for these graphs will be

$$P(G) = \frac{1}{Z}\exp\left(-\sum_i \beta_i \epsilon_i\right) \tag{3}$$

where Z is the partition function given by

$$Z = \sum_G \exp\left(-\sum_i \beta_i \epsilon_i\right) \tag{4}$$

Furthermore, the average of a graph observable ϵ_i is obtained through the use of free energy f. Thus,

$$< \epsilon_i >= \frac{\partial f}{\partial \beta_i} \tag{5}$$

This setup allows us to perform calculations on the models. However, there are currently no general analytical solutions for ERGMs, so we have to rely on Monte Carlo simulations.

2.3 Markov Chain Monte Carlo

The foundation and beginning of the Markov Chain Monte Carlo (MCMC) method were laid by Metropolis et al. [33] and an important generalizations were made by Hastings [34]. The method has it's roots in statistical physics, but it took nearly 40 years for them to become part of the mainstream in statistics. It has become an essential tool in many areas of application, including especially multi-dimensional numerical integration. One of it's advantages is it's versatility. The same method (and thus the same software) can be applied to variety of problems [35].

The MCMC is essentially a Monte Carlo method using Markov chains. The MCMC algorithms sample from probability distributions by means of simulation. It has the desired distribution as it's equilibrium distribution. A Markov chain is a sequence of random variables where the distribution of each random variable depends only on the value of the previous random variable. This leads to a system, where configuration k_{n+1} is created from previous configuration k_n stochastically [36]. The information about the previous configuration is not needed. The Markov process operates by a (often a very large) transition matrix W that contains the transition probabilities. In the Metropolis-Hastings algorithm, the process begins from an initial state with an evaluation of the desired distribution at this state. A proposal is generated, evaluated and then either accepted or rejected. Good proposals are accepted with higher probability than proposals that lead away from the equilibrium distribution. When a proposed state is rejected then the resulting state $k_{n+1} = k_n$. The sampled distributions tend to equilibrium over a large number of steps. There are also other methods, such as Gibbs sampling and slice sampling but they are not used in this study. Increasing the number of steps usually increases the accuracy of the simulation process, but there are always some residuals from the initial state. This is also a major difference between the MCMC and the standard Monte Carlo algorithms, where each step is statistically independent of the other steps.

2.4 Mathematical definition of the observabels

There are essentially two different *clustering coefficients* C_k that are frequently used [37]. We here use the following definition

$$C_k = \frac{3N_\Delta}{N_3} = 3 \sum_{k>j>i} a_{ij} a_{ik} a_{jk} / \sum_{k>j>i} \left(a_{ij} a_{ik} + a_{ji} a_{jk} + a_{ki} a_{kj} \right) \tag{6}$$

The term N_Δ takes into account the number of triangles found in a network. Note that the product $a_{ij} a_{ik} a_{jk} = 1$ if and only if there exists all three links between the three nodes i, j and k. Otherwise the product equals zero. This comes from the fact that we are not dealing with weighted networks. In another words, terms a_{xy} can only have value 1 if there exists a link and 0 if there does not exist a link between two given nodes.

The *communicability betweenness centrality* (CBC) for a network is defined as an average of communicability between any two nodes in the network. Communicability between two nodes G_{AB} is commonly defined as the ability of an arbitrary piece of information to travel from node A to node B. We use the form proposed by Estrada [38]

$$G_{AB} = \sum_{k=0}^{\infty} \frac{\left(A^k \right)_{AB}}{k!} \tag{7}$$

where A be the adjacency matrix. In adjancecy matrix you have nodes represented as rows and columns. The element $a_{ij} = 1$ if there is an edge from node i to j. Otherwise $a_{ij} = 0$.

This definition of communicability considers not only the shortest path from node A to node B, but extends it's reach to other non-shortest paths. As in real world arbitrary particles of information are more likely to use shorter than longer paths to get from a node to another, the relevance of longer walks is scaled down by the term $(k!)^{-1}$, variable k being the length of the path in consideration. It has been shown [39] that the element $\left(A^k\right)_{AB}$ gives the number of walks from node A to node B that have the length of k.

A *link grade distribution* shows how many links there are connecting each node to other nodes. It helps a researcher to determine the if the model is degenerate or not. In this study we plot the *traces* of different variables as they undergo MCMC simulation processes. A trace shows the evolution of the value of the variable during the simulation.

3 The empirical data

The concept maps considered in this work were done by the physics teacher students in their third or fourth year. The subject of the concept maps was electromagnetism. By this time the students should have developed a fair understanding of the subject. They had also received teaching on how to develop a concept map. A total of 46 concept maps were analyzed in this research. There were 20 teacher students making concept maps from 2008 to 2010. Each student created two concept maps. Some authors created maps both in 2009 and 2010. (Which explains how 20 students created 46 maps.) A list of concepts to be used in the maps they were to produce were given to them. This means that the number of concepts will be constant which is something that is important to keep in mind while modeling these networks. Students were required to justify the links made between any two nodes. They were to choose the direction of the connection as well as declare whether the connection is due an experimental procedure or theoretical relation between the two concepts. For a concept map to be included in this research the justifications and directions of the links needed to be logical for the majority of the links.

The data from the concept maps made by the teacher students can now be included in an adjacency matrix. An element a_{ij} of the adjacency matrix A has the binary value of 1 if there is a link going from node (or concept) i to node j. The value 0 means there is no connection in that direction. For an undirected network we have a symmetrical matrix with elements$a_{ij} = a_{ji}$ for all $i \neq j$. Loops a_{ii} are not allowed as they are not relevant in physics. The adjacency matrix is then converted to a network object by Statnet's build-in macros.The network object is mathematically a graph. Networks can then be modeled by giving the algorithm the desired modeling parameters [40]. The methods of the analysis is discussed in more detail in chapter 4.

Figure 3 shows an example of a concept map done by a physics student. The map has been chosen to be displayed here because it is a typical example of the maps researched in this study. Some of it's features are visible even without any analysis. There are some triangles in the graph, but also 4 concepts with only one link connecting them to other concepts. Many of the concepts are connected to others with very few links. This leads to chain-like formations. One of these chains can be observed going counterclockwise from below to right-hand side of the graph. These notions indicate that the topological features of the map are not optimal. The exact analysis of the

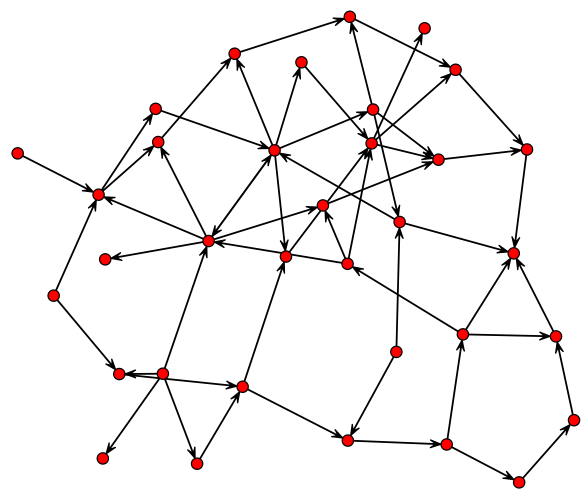


Figure 3: A reconstruction of a decent concept map done by a physics student. The data of the links from the original maps were included in an adjacency matrix. That data was then used to reconstruct this map.

map is done in chapter 5.

Some of the concept maps studied in this research turned out to be relatively poorly constructed. An example of those graphs can be seen in figure 4. The difference the between earlier example and this one is remarkable even to the naked eye. The concept map is lacking much from the hierarchical point of view. First of all, there are very few triangles, 4-cycles and other basic motifs. The map is not well-connected as 50 % of concepts have one or two links connecting them to other concepts. This leads to a shallow and chain-like hierarchical structure. There are number of reasons that could have led to this kind of a map. One of the probable reasons arise from the fact that the students were required to justify each link they made

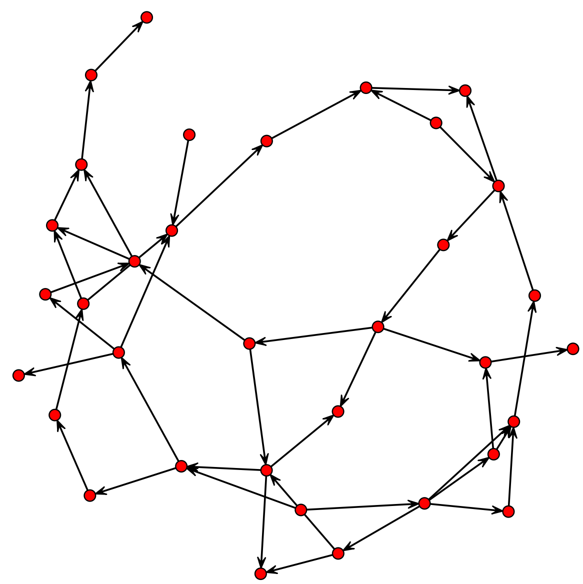


Figure 4: An example of a poorly constructed concept map.

between two concepts. If a student doesn't have a strong understanding of the definition and application of a concept in physics, it is hard to justify it's connections to other concepts. To justify a link by experiment requires specific understanding of the experimental procedure and of the conditions and boundary rules involved.

4 Method of analysis

The original data was first processed so that the analysis software can make a use of it. Data was first taken in as an adjacency matrix. The concept maps were directional so adjacency matrices were not symmetrical. Some of the analysis steps require the symmetry of the adjacency matrix. Therefore corresponding symmetrical matrices were obtained using a simple (pseudo) algorithm:

$$\texttt{if}\ a_{ij} = 1\ \texttt{set}\ a_{ji} = 1\ \forall\ i, j \in \{1, 2, \ldots, 34\} \tag{8}$$

Note that element $a_{ij} = 1$ means there is link going from vertex i to vertex j. Indexes get integer values from 1 to 34 as there were 34 (given) concepts in maps. Matrices were then converted into Statnet's network objects. This concludes the data preparation phase.

4.1 Models

Initially both directed and undirected networks were considered. Eventually only models 6 and 7 make use of the directness of the graphs. Note that the number of nodes was kept constant all the time since the real world concept maps had specific set of concepts. This information together with the fact that the networks are non-dynamic helps the modeling procedure to yield more accurate results. There are even more boundary conditions that were applied into the models. Model 7 is the most constrained model in this study and as such it stands out as an important experiment. The model specifications are listed in table 3. Each model is discussed in brief in the following text.

We begin by some of the most basic modeling ideas. We first turn attention to counting edges and triangles. The original network has 4 triangles and 55 edges. If the model isn't degenerate, these parameter values should vary only to some extend while going through the MCMC steps. Triangles have shown to be important motifs to describe the information processing of a hierarchical knowledge. Counting triangles is important in modeling also because it provides information about the topology of the network. By counting the number of edges the simulated networks will have realistic number of connections between nodes. Both 3-stars and triangles are used widely in the theory of knowledge production, acquisition and processing. That is why model 2 has some potential in describing the the hierarchical

Table 1: Model specifications

Model	Parameters
Model 1	triangles + edges
Model 2	triangles + 3-stars
Model 3	twopaths + 4-cycles + edges
Model 4	3-stars + 4-cycles + edges
Model 5	triangles + 3-stars + edges
Model 6	transitivity + 4-cycles + triangles + edges
Model 7	transitivity + 4-cycles + triangles + edges + constrains

properties of the students concept maps. As it turns out simulated graphs produced by model 2 do share some topological properties with the real world graphs. But this comes with the cost of an increasing number of edges in the simulated graphs.

In model 3 a parameter linked to transitivity and small cycles were taken in consideration. A knowledge structure in physics features triangles and k-stars rather than cycles. So this model is not expected to produce accurate and realistic results, but stands as an important reference. Model 4 combines ideas from models 2 and 3. It can be considered (more so than others) experimental, the reason being that there is no clear physical justification of the model besides the fact that it combines parameters of the earlier models. Adding counting of edges to model 2 gives us model 5. The purpose of this is to guide the simulation process to more realistic edge numbers.

The motivation for the next model came from building on the ideas of model 3. By intuition transitivity seems to be a very important factor in forming of the knowledge structure. Multitude of different models including transitivity were examined here. Besides the actual model 6 the most promising were models: transitivity + edges and transitivity + triangles. Still both of them failed to produce much of value and are thus neglected here. The final definition of model 6 pulls together many relevant motif counting variables. One more model was defined. Here a parameter was added to model 6 to force at least one edge to all nodes as there are no isolate concepts in electromagnetism. Constraining the modeling procedure can have fatal effects when ran through the MCMC steps. This nevertheless might guide the process to more realistic outcome as measured by how they relate to the original data.

4.2 Simulation process

The values for statistical network variables such as density of edges and number of triangles were measured from the original empirical data. For example the density of edges of given network "`net`" can be calculated (among other information) using Statnet's command `summary(net)`. The most useful Statnet and R commands in this research are listed in the table 2. These commands are listed here so that an interested reader can quickly try out modeling and simulation. A lot more commands were actually needed to run the complete modeling and simulation of this research including many custom developed R functions that are listed in the appendix. For a much more complete selection of commands and their specifications interested readers are advised to consult Morris et al. [41].

Table 2: A small summary of the most vital Statnet commands

Command	Use
matrix()	Scans raw link data into an adjacency matrix
net()	Creates a Statnet network object of a matrix
ergm()	Creates a Statnet model based on the given specifications and a Statnet network object
simulate()	Simulates a model until a given limit of simulations is reached

A graph with typical statistical and topological properties was chosen to be modeled. A single graph is needed because of the way Statnet handles and creates models. The values of the observables were then used as modeling parameters for Statnet algorithms. The number of vertices was taken in as a boundary rule for the modeling. Different sets of modeling variables were chosen to construct different kinds of models as was discussed in chapter 4.1. The obtained models were then simulated using Statnet's simulation capabilities. Besides the simple edge, triangle and small circle counting we looked at two important variables. The *clustering coefficient* measures the tendency to form closed triangles. It can be calculated with relative ease [30].

A custom R algorithm was developed to calculate this. Another important parameter to look at is communicability betweenness centrality. Communicability between nodes p and q in the network represents the probability that an arbitrary particle of piece of information of some kind travels from node p to node q. *Communicability betweenness centrality* (CBC) for a node r is defined as the weighted sum over walks involving node r [38]. Custom algorithms were developed to calculate these values. Exact mathematical definitions can be found in chapter 2.4.

4.3 Software environment

R is an open source programming language and software environment for statistical computing. It provides tools for statistical analysis, including modeling, classification and simulation. R is highly extensible through the use of user-submitted packages [42]. Statnet package of R is one these extensions. It is being developed by many contributors from several universities [43]. Statnet was used in this study solely in modeling and simulation of graphs. Statnet was chosen as a software environment because of it's extensive library of build-in functions to model graphs. It supports directed as well as non-directed networks and thus fits for our purposes. Statnet benefits from the recent advancements in statistical analysis, such as random networks, which are also relevant for our study. It provides wide variety of tools for model estimation and evaluation. As an important feature it also provides simulation capabilities. This allows us to simulate the models and see whether they produce real-world like graphs or not. [19]

The functions (relevant to this study) that have been coded into Statnet make use of the Markov Chain Monte Carlo (MCMC) algorithm. The MCMC algorithm is more know from it's use in numerical integration [44]. It samples from probability distributions based on constructing a Markov chain that has the desired distribution as its equilibrium distribution. The state of the chain after a large number of steps is then used as a sample of the desired distribution. The quality of the sample improves as the number of steps increases. In our study the number of steps was $5-7 \cdot 10^5$. Statnet is modeling networks based on models called exponential-family random graph models (ERGMs) or p-star models. The purpose of it is to help quantify local effects that shape the global structure of the network [45]. Statnet also provides tools to evaluate the goodness of a fit through graphical representations. The MCMC estimation can sometimes lead to undesired results. These include irrational topological properties of sample networks and far from average values for measured attributes. In that case the model is likely to be a degenerate one. Degenerate models cannot be used to describe any real world phenomena, but sometimes they can be fixed by adjusting the model specifications.

5 Empirical results

1000 networks were produced in simulation of each model. Simulated graphs were used to determine the values for clustering coefficient and communicability betweenness centrality (CBC). For demonstration one of them was chosen and plotted for each model (except model 3). The network based on real world concept map created by a physics student is displayed in figure 5. The communicability distribution of the concept map is plotted in figure 6 to serve as a reference. Three nodes with communicability greater than 0.2 can be observed. These three nodes can thus be considered hubs in the network. The network is not very dense with respect to links as there are 55 edges and 34 nodes. This usually leads to long walks that involve nodes with only two or three links. This can be seen in the figure as well. The clustering coefficient of the network was 0.08 as can be seen in table 3. This is realistic and expected result as similar values have been observed for numerous real world networks such as the Internet, power grids or protein interactions [30]. The same can be said about the average communicability betweenness centrality over the nodes. $< CBC >= 0.010$ is realistic for sparse networks.

As can be seen in the parameters table (table 3) the characteristical values of model 1 do fall in line with the real world graphs. But mere averages do not paint the whole picture. Looking at how the links were

Table 3: Characteristical parameters of the models. C_k is the average clustering coefficient and $<$ CBC $>$ the average communicability betweenness. Their respective stadard deviations are provided as well.

Model	C_k	$\sigma(C_k)$	$<$ CBC $>$	$\sigma(<$ CBC $>)$
Real World	0.08	-	0.010	-
Model 1	0.07	0.04	0.010	0.005
Model 2	0.04	0.03	0.091	0.003
Model 3	0.08	0.04	0.103	0.009
Model 4	0.13	0.03	0.109	0.004
Model 5	0.09	0.04	0.097	0.003
Model 6	0.08	0.04	0.098	0.004
Model 7	0.04	0.03	0.093	0.003

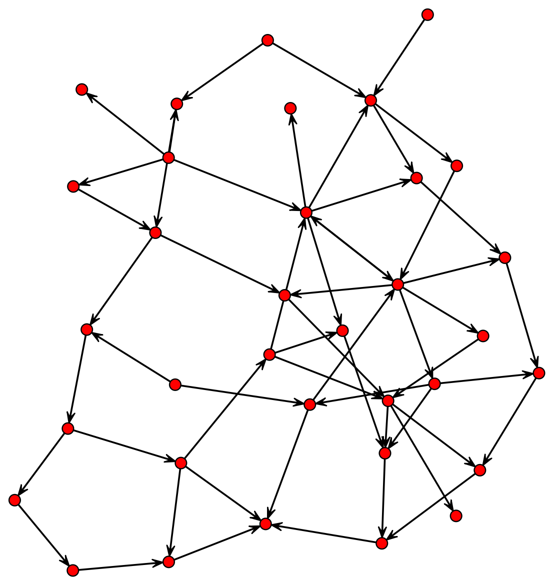


Figure 5: A reconstruction of real world graph drawn by teacher student.

distributed over the nodes (which is called link grade distribution) you can find nodes with as many as 9 edges in the simulated graphs. This indicates that the model seems to capture the generating process poorly. Intuition tells us that model 2 should describe the hierarchy better than simply calculating the edges and triangles. The MCMC diagnostics look promising for model 2 as they did for model 1. Densities vary around the observed values. After simulation triangle count matches observations. The simulated graph seems well connected and realistic. The average number of links in the graphs increased by 12,7%. This is not a big surprise as edge count was not a parameter in this model.

The averages of edge and 4-cycle density distributions of model 3 are not close to the values of the modeled network. Means are marked as dashed lines in figure 8. If a model is not degenerate then the densitiy distribu-

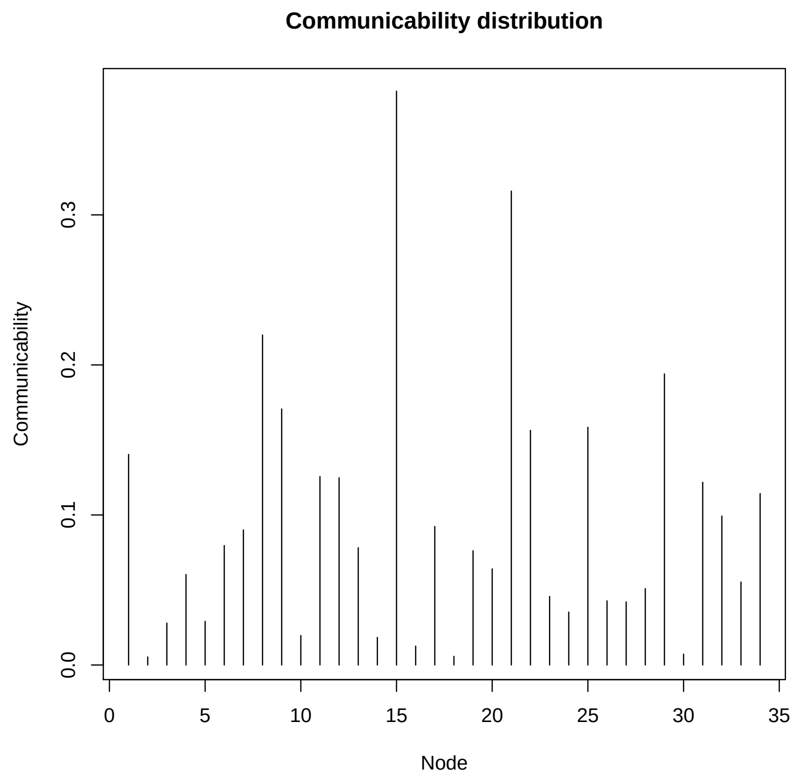
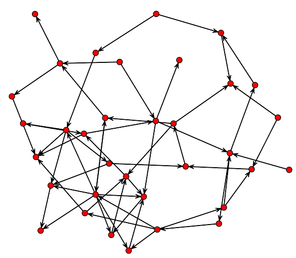
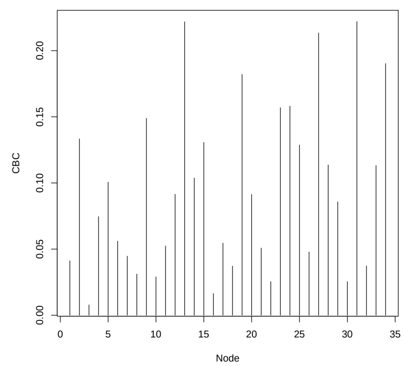


Figure 6: The communicability distribution of a concept map of figure 5. The nodes with higher communicability can be considered more important when linking concepts. That is to say that a node with higher communicability is more likely to be involved when a new concept is introduced to the system.

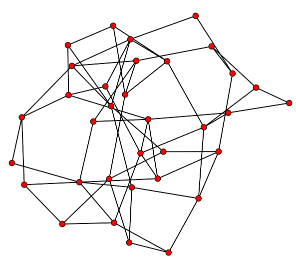
tions drawn in the same figures should be close to symmetrical around the mean values. This is not the case for model 3 simulations so it indicates a degenerate model. The density traces didn't converge during simulations. Interestingly this model seems to produce only very few triangles. Triangles are vital motifs so this has to be considered a yet another downside of this model. Simulation did not produce enough acceptable graphs so that the communicability distribution could be determined. This is further evidence of model degeneration. The clustering coefficient and average value of communicability betweenness centrality is in line with the real world network. It is normal even for a degenerate model to produce realistic average values.



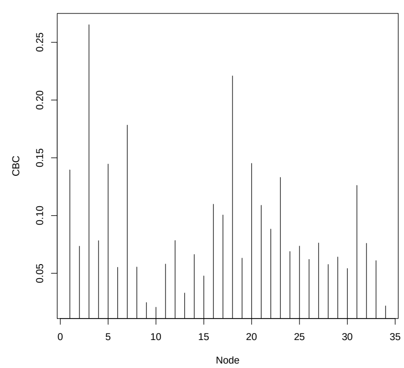
(a)



(b)



(c)



(d)

Figure 7: (a) A graph generated by simulating model 1 and (b) communicability distribution of the simulation. (c) A graph generated by simulating model 2 and (d) communicability distribution of the simulation.

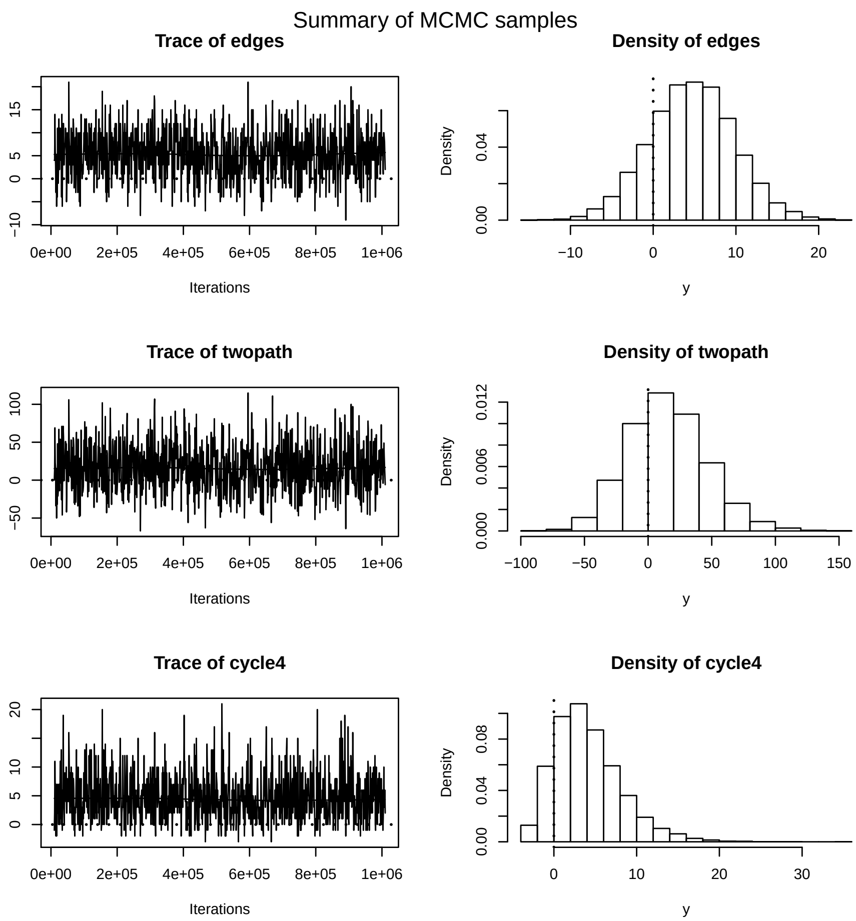
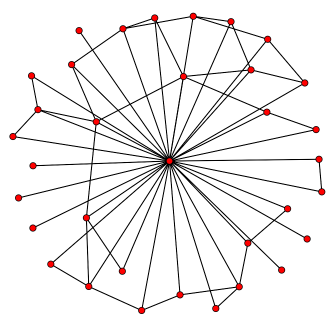
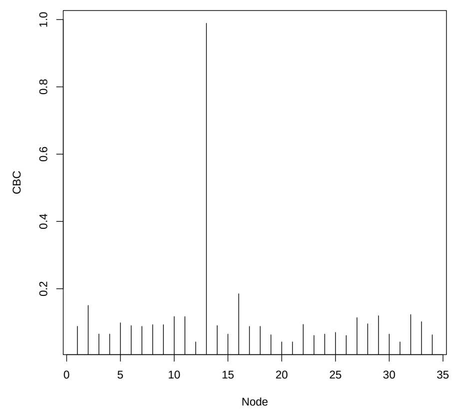


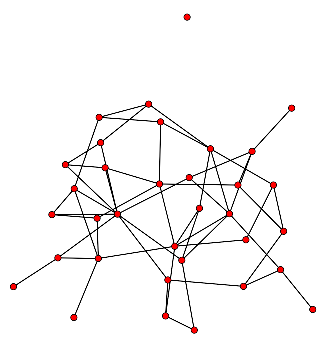
Figure 8: MCMC diagnostics of the model 3. The dashed lines in the right-hand side figures are the mean values for the densities. If a model is not degenerate then the densitiy distributions drawn in the same figures should be close to symmetrical around the mean values.



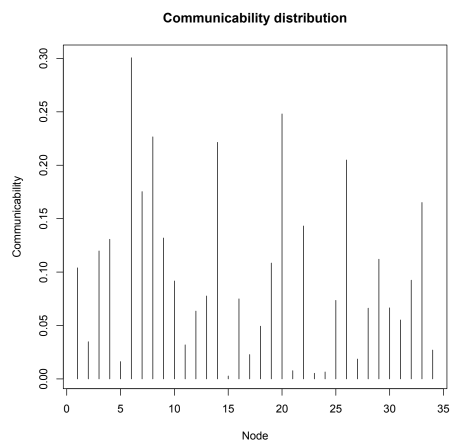
(a)



(b)



(c)

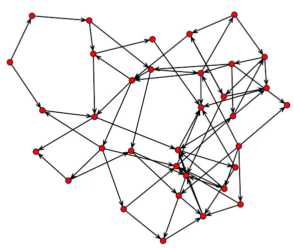


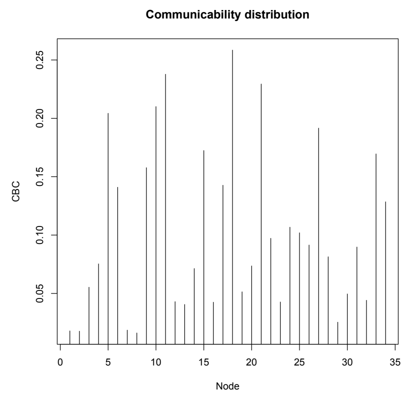
(d)

Figure 9: (a) A graph generated by simulating model 4 and (b) communicability distribution of the simulation. (c) A graph generated by simulating model 5 and (d) communicability distribution of the simulation.

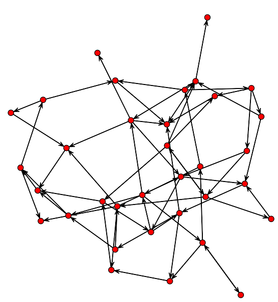
Looking at the one of the graphs produced in the simulation of the model 4 (in figure 9) we notice alarming topology. The communicability betweenness centrality distribution has a spike which is consistent with the topology of the simulated graph. The clustering coefficient is on average notably higher than the original graph. Looking at the MCMC diagnostics of the simulation it becomes clear that the model is degenerate. All the means are very much off the desired, real world network values. This is good news since this model was not based on any vision on the network structure. The value for the 3-star count is off by huge margin. This is consistent with observations on the clustering coefficient. All this is evidence that the model indeed is degenerate and thus can be considered irrelevant for us. The model diagnostics for model 5 are somewhat surprising. Characteristic parameters look promising as the means of clustering coefficient and communicability betweenness centrality are in line with the original real world network. Still all parameter value distributions signal degeneracy of the model. The simulated graph also doesn't look satisfactory as there is an isolated node, 4 nodes with only 1 edge and 53% of the nodes having exactly 3 edges.

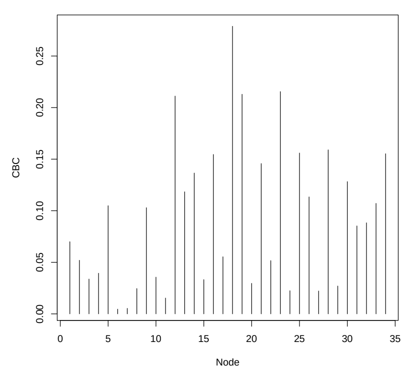
Model 6 takes transitivity, 4-cycles, triangles and edges in as model parameters. Note that transitivity is applied for directed network. There is no clear evidence of degeneracy. Simulation produced graphs that look satisfactory except for the isolates found in many simulations. The featured graph in figure 10 has 13 triangles and 71 edges compared to the 4 triangles and 55 edges of the real world network. The link grade distribution of the model is realistic on average. As there are no isolate concept definitions in physics the number of isolates in the models should be limited to 0 by some boundary rule. That is done in model 7 by setting boundary rules that force minimum of one edge for each node. It turned out that adding boundary conditions tends to provoke minor increase in the number of edges in the modeling. The sample graph generated by the simulation of model 7 has 63 edges and 3 triangles. That is in line with the real world graph. The communicability betweenness centrality distribution is plotted in figure 10. Distribution (d) is calculated for a simulated network based on the model 7. It is fairly similar with the real world graph. Both have few nodes with minimal communicability betweenness centrality. The simulated graph doesn't yield to any nodes having CBC over 0.3. On the other hand the number of nodes with CBC over 0.2 is nearly identical (3 in real world graph to 4 in simulated). In light of CBC simulating the model 7 produces realistic networks. The clustering coefficient is below the real world value.





(a) (b)





(c) (d)

Figure 10: (a) A graph generated by simulating model 6 and (b) communicability distribution of the simulation. (c) A graph generated by simulating model 7 and (d) communicability distribution of the simulation.

6 Discussion and conclusions

We have studied seven different models to describe physics students knowledge structure. The models ranged from straightforward calculation of triangles and edges to more complex models that take in account the transitivity as well as cycles and triangles. The choice of parameters was based on studies that have shown humans to process information by similar motifs.

Each model was simulated in order for us to compare the resulting graphs with concept maps done by physics students. A model that produced realistic (compared to students' concept maps) clustering coefficient and communicability betweenness values was considered relevant. During the research it became obvious that the choice of model parameters had to be done very carefully. Including a seemingly ineffective parameter could easily degenerate the model. Degenerate models fail during the Markov Chain Monte Carlo steps and thus produce unrealistic and undesired results. Models 3 and 4 are degenerate. Model 6 was the most accurate model by many means. The clustering and communicability values it produced were very close to those calculated of the real world graphs. This observation answers to the second research question. We have found the most accurate model (in the scope of our research).

According to the research documented in this paper it seems possible that at least highly hierarchical information can be modeled by triangles and other simple patterns formed by the key concepts and meaningful links between them. That answers to the first research question. The simulated graphs that were produced by models that had basic motifs as their parameters do preserve some of the topological and hierarchical information of the real world concept maps and can thus be considered relevant.

This study has some meaningful implications for teachers as well as text book authors. In the future, the approach introduced in this paper could be developed into a tool to help monitor students' meaningful learning. Students could make the concept maps using an online tool. The data would then be processed automatically using powerful computers and results would be visible to the teacher. The tool could pinpoint areas of the subject matter that have been generally poorly learned and thus require more attention from the teacher. The kind of modeling that was done in this article provides insight for text book authors in arranging content in such manner that it supports creation of basic cognitive motifs in learners mind. As we have demonstrated in this study, this kind of practice will lead to formation of larger cognitive hierarchies and thus lead to better meaningful understanding of the subject matter.

There is much fruitful ground to be covered by the approach introduced in this article. There are two distinct ways to expand and deepen the study. First, new models could be created based on the findings in this article. Alternatively, the models proposed here could be refined even further. Secondly, there is also a possibility to look into more robust observables such as statistical attributes of the ensembles. This could be achieved by comparing statistical observables for real-world ensembles with ensembles created by simulating a model. These two directions would lead into deeper understanding of the link between how humans acquire and process knowledge and simple patterns found in well-developed concept maps.

References

[1] J. L. Snyder. An investigation of the knowledge structures of experts, intermediates and novices in physics. *International Journal of Science Education*, 22(9):979–992, 2000.

[2] I.T. Koponen and M. Pehkonen. Coherent knowledge structures of physics represented as concept networks in teacher education. *Science & Education*, 19(3):259–282, 2010.

[3] L. da Fontoura Costa. Learning about knowledge: A complex network approach. *Physical Review E*, 74(2):026103, 2006.

[4] J. B. Batista and L. F. Costa. Knowledge acquisition by networks of interacting agents in the presence of observation errors. *Physical Review E*, 82(1):016103, 2010.

[5] Ş. K. İngeç. Analysing concept maps as an assessment tool in teaching physics and comparison with the achievement tests. *International Journal of Science Education*, 31(14):1897–1915, 2009.

[6] P. Thagard. *Conceptual revolutions*. Princeton Univ Pr, 1992.

[7] L. BonJour. *The structure of empirical knowledge*. Harvard Univ Pr, 1985.

[8] J. D. Novak and D. B. Gowin. *Learning how to learn*. Cambridge Univ Pr, 1984.

[9] A. A. DiSessa. A bird's-eye view of the "pieces" vs. "coherence" controversy. *International handbook of research on conceptual change*, pages 35–60, 2008.

[10] A. M. O'Donnell, D. F. Dansereau, and R. H. Hall. Knowledge maps as scaffolds for cognitive processing. *Educational Psychology Review*, 14(1):71–86, 2002.

[11] C. Kemp, A. Perfors, and J. B. Tenenbaum. Learning overhypotheses with hierarchical bayesian models. *Developmental science*, 10(3):307–321, 2007.

[12] C. Kemp and J. B. Tenenbaum. The discovery of structural form. *Proceedings of the National Academy of Sciences*, 105(31):10687, 2008.

[13] T. H. Duong, G. S. Jo, J. J. Jung, and N. T. Nguyen. Complexity analysis of ontology integration methodologies: A comparative study. *Journal of Universal Computer Science*, 15(4):877–897, 2009.

[14] M. Nousiainen and I. T. Koponen. Concept maps representing knowledge of physics: Connecting structure and content in the context of electricity and magnetism. *Nordic Studies in Science Education*, 6(2):155–172, 2010.

[15] R. Albert and A.L. Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[16] P. W. Holland and S. Leinhardt. An exponential family of probability distributions for directed graphs. *Journal of the american Statistical association*, 76(373):33–50, 1981.

[17] J. Park and M. E. J. Newman. Statistical mechanics of networks. *Physical Review E*, 70(6):066117, 2004.

[18] J. Park and M. E. J. Newman. Solution of the two-star model of a network. *Physical Review E*, 70(6):066146, 2004.

[19] M. S. Handcock, D. R. Hunter, C. T. Butts, S. M. Goodreau, and M. Morris. statnet: Software tools for the representation, visualization, analysis and simulation of network data. *Journal of Statistical Software*, 24(1):1548, 2008.

[20] I.T. Koponen and M. Pehkonen. Entropy and energy in characterizing the organization of concept maps in learning science. *Entropy*, 12(7):1653–1672, 2010.

[21] I. J. Good. Some statistical applications of poisson's work. *Statistical Science*, 1(2):157–170, 1986.

[22] J. D. Bransford, A. L. Brown, R. R. Cocking, et al. *How people learn*. National Academy Press Washington, DC, 2000.

[23] M. Nousiainen. Coherence of pre-service physics teachers' views of the relatedness of physics concepts. *Science & Education*, pages 1–21, 2012.

[24] Qi-Ren Z. Intensity dependence of the photoelectric effect induced by a circularly polarized laser beam. *Physics Letters A*, 216, 1996.

[25] E. N. Gilbert. Random graphs. *The Annals of Mathematical Statistics*, pages 1141–1144, 1959.

[26] P. Erdős and A. Rényi. On the evolution of random graphs. *Magyar Tud. Akad. Mat. Kutató Int. Közl*, 5:17–61, 1960.

[27] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64:026118, 2001.

[28] M. Kretzschmar and M. Morris. Measures of concurrency in networks and the spread of infectious disease. *Mathematical biosciences*, 133(2):165–195, 1996.

[29] D. J. Watts. Networks, dynamics, and the small-world phenomenon. *American Journal of Sociology*, 105(2):493–527, 1999.

[30] M. E. J. Newman. The structure and function of complex networks. *SIAM review*, pages 167–256, 2003.

[31] A.-L. Barabási and Réka A. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[32] D. Strauss. On a general class of models for interaction. *SIAM review*, 28(4):513–527, 1986.

[33] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21:1087, 1953.

[34] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[35] W. R. Gilks. Markov chain monte carlo. *Encyclopedia of Biostatistics*, 2005.

[36] B. A. Berg. Introduction to markov chain monte carlo simulations and their statistical analysis. *Phys. Stat. Mech*, 2004.

[37] L.F. Costa, F.A. Rodrigues, G. Travieso, and P.R.V. Boas. Characterization of complex networks: A survey of measurements. *Advances in Physics*, 56(1):167–242, 2007.

[38] E. Estrada, D. J. Higham, and N. Hatano. Communicability betweenness in complex networks. *Physica A: Statistical Mechanics and its Applications*, 388(5):764–774, 2009.

[39] N. Biggs. *Algebraic graph theory*. Cambridge university press, 1994.

[40] C. T. Butts. network: a package for managing relational data in r. *Journal of Statistical*, 2008.

[41] M. Morris, M. S. Handcock, and D. R. Hunter. Specification of exponential-family random graph models: terms and computational aspects. *Journal of Statistical Software*, 24(4):1548, 2008.

[42] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011. ISBN 3-900051-07-0.

[43] C. T. Butts Steven M. Goodreau Pavel N. Krivitsky M. S. Handcock, D. R. Hunter and M. Morris. *statnet: Software Tools for the Statistical Modeling of Network Data*. Seattle, WA, 2003. Version 2.1-1 . Project home page at http://statnet.org.

[44] C. P. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Verlag, 2004.

[45] D. R. Hunter, M. S. Handcock, C. T. Butts, S. M. Goodreau, and M. Morris. ergm: A package to fit, simulate and diagnose exponential-family models for networks. *Journal of Statistical Software*, 24(3):nihpa54860, 2008.

Appendix

A. Statnet and R code listings

Here is listed the core Statnet code listings developed and applied in this research. Many more functions were developed and used by the author, but their main function was to automatize simulation and plotting functions. The core functions such as calculating the important observables are included. Note that some of the functions require the "statnet" and "Matrix" packages.

The first three functions listed here are essential support functions. As the data itself has directed networks and some of the functions are written for undirected network we need a way to transform directed networks into undirected matrices. This is done in the **`MakeSymmetrical`** function. The **`ReturnAsMatrix`** function converts Statnet network objects into matrices. The **`DeleteLines`** function is needed to make all elements $a_{rk} = 0$ and $a_{kr} = 0$ for some r and $k = 1, \ldots, 34$.

```
MakeSymmetrical <- function(net){
          m <- as.matrix(net)
          for(j in 1:34){
                    for(k in 1:34){
                              if(m[j,k]==1) m[k,j]<-1
                    }
          }
          return(m)
}

ReturnAsMatrix <- function(net){
          return(Matrix(symmetrisoi(as.matrix(net))))
}

DeleteLines <- function(A,r){
          for(k in 1:34){
                    A[k,r]=0
                    A[r,k]=0
          }
          return(A)
}
```

The `CalcCBC` function provides the core functionality for calculating the communicability betweenness for a single node *r*. The function only applies to the data of this research as the constant 1057 is a result of calculation involving the number of nodes. The other two CBC functions utilize aforementioned function to provide the CBC for a network and a model. The `CBCforModel` function illustrates how simulation commands are used in Statnet.

```
CalcCBC <- function(A,r){
        sum = 0
        B = DeleteLines(A,r)
        aar = Matrix(seq(1, 34^2, 1), nc=34)
        erk = Matrix(seq(1, 34^2, 1), nc=34)
        for (p in 1:34){
                if(p == r) next;
                for (q in 1:34){
                        if(q == r) next;
                        if(p == q) next;
                        erk = expm(B)
                        ap1 = erk[p,q]
                        aar = expm(A)
                        ap2 = aar[p,q]
                        sum <- sum + 1 - ap1/ap2
                }
        }
return(sum/1057)
}

CBCforNetwork <- function(net){
        A = ReturnAsMatrix(net)
        for(r in 1:34){
                omega[r]=CalcCBC(A,r)
        }
        return(mean(omega))
}

CBCforModel <- function(model, limit){
        list <- seq(1, limit, 1)
```

```
        for(num in 1:limit){
                sim <- simulate(model, burnin = 1e+5)
                print(num)
                list[as.integer(num)] <- CBC(sim)
        }
        print(lista)
        print(mean(lista))
        print(sd(lista))
}
```

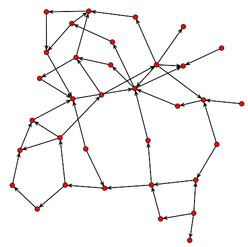
The remaining functions listed below provide for calculation of clustering coefficients.

```
Clustering <- function(a){
        x=0
        y=0
        for(j in 1:33){
                cc=j+1
                for(k in cc:34){
                        for(i in 1:34){
                                x=x+a[i,j]*a[i,k]*a[j,k]
                                y=y+a[i,j]*a[i,k]
                        }
                }
        }
        return(x/y)
}

ClusteringFromFile <- function(tds){
        m <- matrix(scan(file=tds,what=integer(0), sep=','),
                ncol=2, byrow=TRUE)
        net <- network(m, matrix.type="edgelist", directed=TRUE)
        return(Clustering(MakeSymmetrical(net)))
}

Characterize <- function(model, limit){
        list<-seq(1, limit, 1)
        for(num in 1:limit){
                sim <- simulate(model, burnin = 1e+5)
```

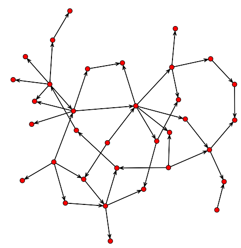
```
                list[as.integer(num)] =
                        Clustering(MakeSymmetrical(sim))
        }
        print(list)
        print(mean(list))
        print(sd(list))
}
```

B. Real-world concept maps

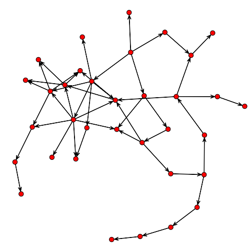
Here we provide an ordered listing of real-world concept maps studied in this research. Only the concept map used as a base for modeling is omitted. Figure of that network can be seen in Figure 5. Maps have been ordered based on their clustering coefficients.



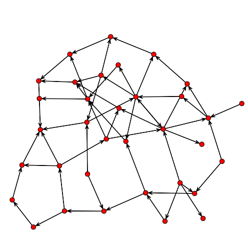
(a) 0.043



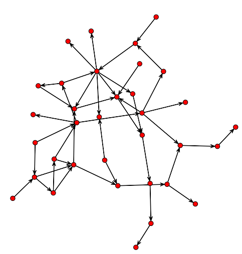
(b) 0.044



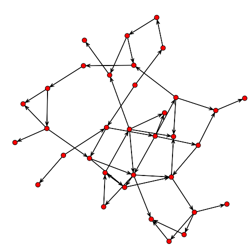
(c) 0.055



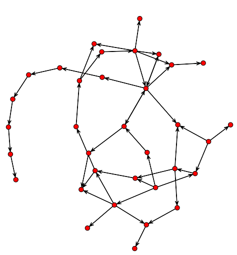
(d) 0.069



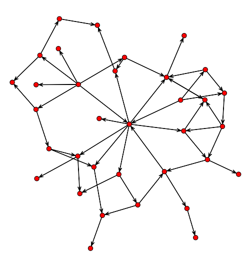
(e) 0.070



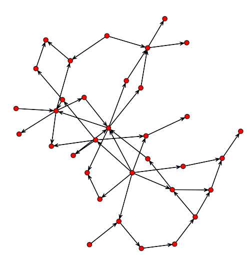
(f) 0.076



(g) 0.079

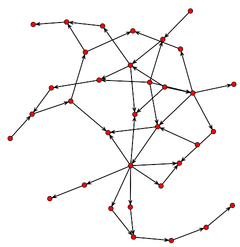


(h) 0.083

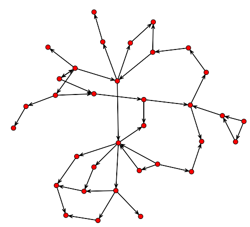


(i) 0.089

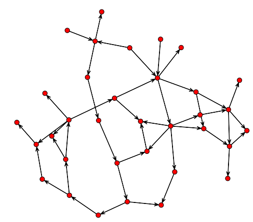
Figure 11: Real-world concept maps represented as networks, with their respective clustering coefficients.



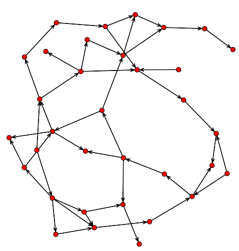
(a) 0.109



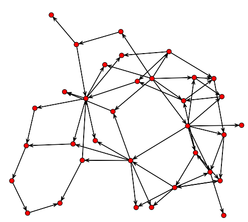
(b) 0.124



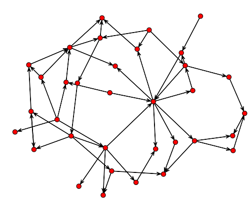
(c) 0.132



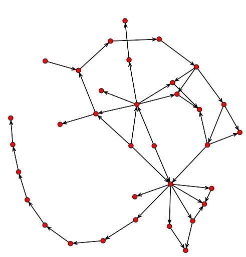
(d) 0.138



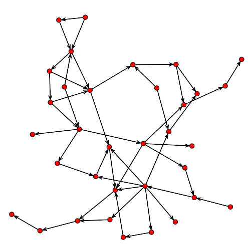
(e) 0.139



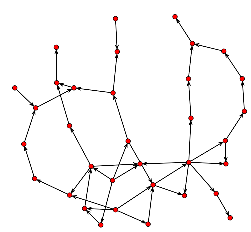
(f) 0.139



(g) 0.147

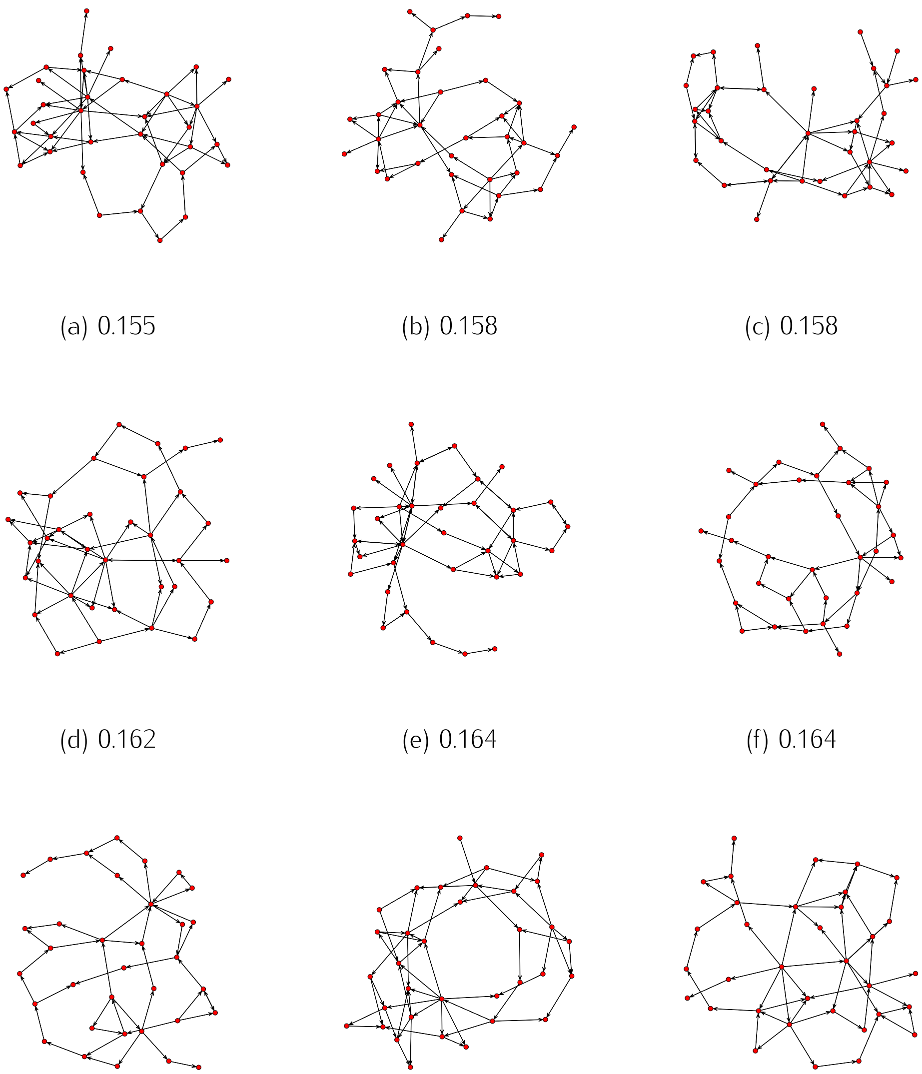


(h) 0.151



(i) 0.152

Figure 12: Real-world concept maps represented as networks, with their respective clustering coefficients.

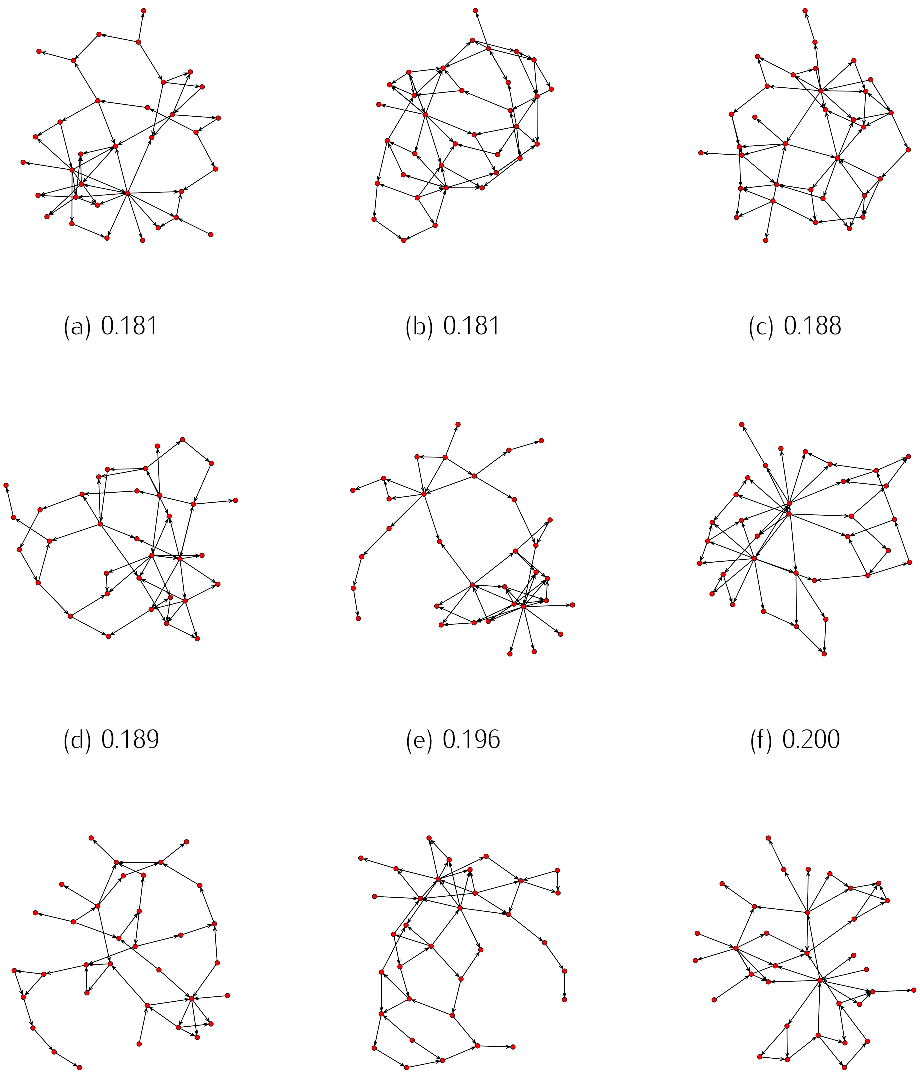


(a) 0.155 (b) 0.158 (c) 0.158

(d) 0.162 (e) 0.164 (f) 0.164

(g) 0.172 (h) 0.176 (i) 0.178

Figure 13: Real-world concept maps represented as networks, with their respective clustering coefficients.

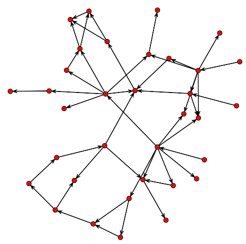


(a) 0.181 (b) 0.181 (c) 0.188

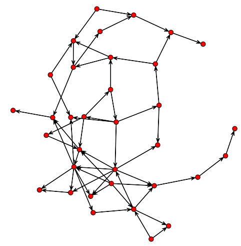
(d) 0.189 (e) 0.196 (f) 0.200

(g) 0.215 (h) 0.243 (i) 0.258

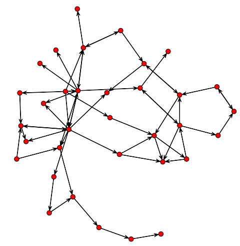
Figure 14: Real-world concept maps represented as networks, with their respective clustering coefficients.



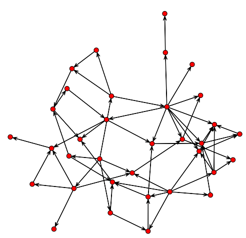
(a) 0.259



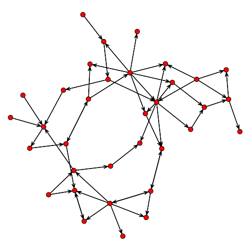
(b) 0.265



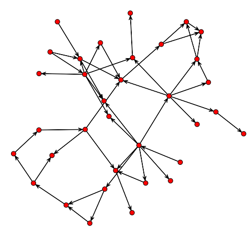
(c) 0.265



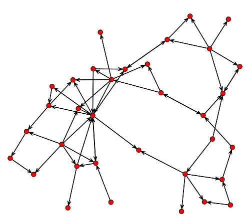
(d) 0.266



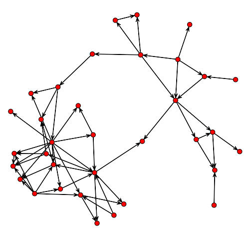
(e) 0.277



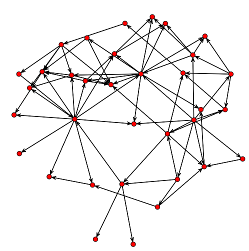
(f) 0.290



(g) 0.322



(h) 0.346



(i) 0.400

Figure 15: Real-world concept maps represented as networks, with their respective clustering coefficients.