Proceedings of the 2011 Winter Simulation Conference
S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, eds.

EXTRACTING HIERARCHIES WITH OVERLAPPING STRUCTURE FROM NETWORK DATA

Brian Cloteaux

Mathematical and Computational Sciences Division
National Institute of Standards and Technology
100 Bureau Drive, Stop 8910
Gaithersburg, MD 20899-8910, U.S.A

ABSTRACT

Relationships between entities in many complex systems, such as the Internet and social networks, have a natural hierarchical organization. Understanding these inherent hierarchies is essential for creating models of these systems. Thus, there is a recent body of research concerning the extraction of hierarchies from networks. We propose a new method for modeling hierarchies through extracting the affiliations of the network. From these affiliations, we construct a lattice of the relationships between nodes. A principal advantage of our approach is that any overlapping community structures of the nodes within the network have a natural representation within the lattice. We then show an example of our method using a real data set.

1 INTRODUCTION

Many complex real-world systems can be modeled by the relationships between the entities in the system. When the aspects of the network to be modeled are the binary relationships between the entities in the network, then the resulting model is a simple graph. One important discovery is that the graphs that are derived from these complex systems are not random. Instead, there is often a large amount of structure that underlies these networks that models need to capture. One structural aspect of these models is that many of them show a natural hierarchical organization of the entities within them. This type of organization has been shown for complex systems such as biological and social networks and the Internet. Understanding these inherent hierarchies is essential for creating models of these systems.

A large component of analyzing and modeling networks from certain domains, such as gene-expression networks (The Gene Ontology Consortium 2008), involves collecting expert's opinions to derive a hierarchy. This is a slow and tedious approach to the basic modeling problem. Because of the importance of understanding the hierarchical structure of the network, several approaches have been proposed to automatically extract hierarchical models from networks. These include the structural decomposition methods of Räcke (2002) (Harrelson, Hildrum, and Rao 2003, Bienkowski, Korzeniowski, and Räcke 2003) and Sales-Pardo, Guimer, Moreira, and Amaral (2007) and the statistical method of Clauset, Moore, and Newman (2008).

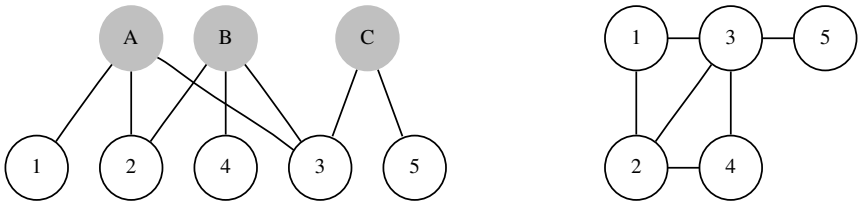
While these methods are successful in finding hierarchies, the hierarchies they find have a simple tree-like structure. In other words, if an entity in the network shares multiple types in the hierarchy (in other words, the hierarchy is not a simple tree), then these methods cannot capture this shared structure in their models. An example appears in gene-expression networks. Genes may have multiple functions, and so they may show up in multiple places in a hierarchy. Towards automating this type of modeling problem, we propose an approach of deriving the hierarchies from the affiliations in the network. This method leads to a partial-ordering of the entities, where the partial order can capture shared structure. We then examine a real-world data set using this method.

2 AFFILIATIONS NETWORKS

A graph, at an abstract level, records a binary relationship between entities. At this level of abstraction, we often represent the network as a simple undirected graph where the entities in the network are the nodes in the graph and the relationships between entities are shown as edges in the graph. This type of graph is typically called the *one-mode representation*. Formally, for a one-mode network G, we designate it as $G = (N, E)$ where N is the set of nodes or entities, and E is the set of edges. To denote the set of edges for a specific network G, we use the notation $E(G)$.

Alternatively, many networks have a natural representation between its entities and the groups (or *affiliations*) to which the entities belong. This representation of the data in the network is called the *affiliation network* or alternatively, its *two-mode representation*. To formalize our discussion of affiliation networks, we use notation borrowed from Guillaume and Latapy (2004). An affiliation network A is a bipartite graph $A = (\top, \bot, E)$ where $\top$ and $\bot$ are called the top and bottom sets respectively and E is the set of edges. The bottom set represents the entities in the graph, while the top set is the set of affiliations among the entities. All the edges in E link a node in the top set $\top$ to a node in bottom set $\bot$, i.e. $E \subseteq \top \times \bot$.

The connection between one-mode and two-mode networks is straightforward. For a given two-mode representation of a network, we can uniquely obtain a corresponding one-mode representation by taking a *projection* or *folding*. To compute the one-mode projection of an affiliation network, we start by defining its set of nodes as the bottom set of the two-mode graph. To define the edge set, two nodes in the projection have an edge between them if those nodes share an affiliation in the two-mode representation. In other words, an edge (n_i, n_j) is in the one-mode representation only if there exists an affiliation $a \in \top$ such that (n_i, a) and (n_j, a) are in the edge set of the two-mode representation. An example of a two-mode network and its one-mode projection is shown in Figure 1. Although multiple edges between two nodes can be defined from nodes sharing more than one affiliation, since the one-mode projection is a simple graph, then only one edge is recorded. This antisymmetry means that while the one-mode projection is unique for a given affiliation network, for any one-mode network, there can be many two-mode networks that project to it.



(a) Two-mode graph (b) One-mode projection

Figure 1: A two-mode network and its one-mode projection. Here in Figure 1a, the set of entities or the bottom set is $\bot = \{1,2,3,4,5\}$ while the set of affiliations or the top set is $\top = \{A,B,C\}$. In Figure 1b, we see the one-mode projection of the affiliation network. Here there is an edge between two nodes in the one-mode network if those nodes both are connected to the same affiliation.

Interest in affiliation networks stems partly from the observation that a large number of real-world networks have a natural bipartite structure inherent in them. In fact, the idea of examining affiliations can be traced back to research in sociology on the relationship of individuals to the groups to which they belong. But even for networks without an obvious bipartite structure, affiliation analysis is still a useful exercise. Recently, it has been shown that any one-mode network can be analyzed in terms of an associated

affiliation network (Guillaume and Latapy 2004), and that analysis of two-mode representations allows for additional approaches to understanding a network's structure (Latapy, Magnien, and Vecchio 2008).

In addition, many empirically observed properties of real-world networks such as power-law distribution, clustering, densification, and a shrinking diameter (Barabási and Albert 1999; Watts and Strogatz 1998; Leskovec, Kleinberg, and Faloutsos 2005) can be modeled in terms of changes to an associated affiliation network (Zheleva, Sharara, and Getoor 2009; Lattanzi and Sivakumar 2009; Guillaume and Latapy 2006). It appears that understanding the affiliations associated with many networks is necessary to be able to accurately model the evolution of those networks.

We are interested in an additional feature of affiliation networks. From the affiliations, we can easily create a lattice representation of the entities in the network. A lattice allows us to extract hierarchical information about the network while allowing overlap in the structure. To see this, let us first see how we extract affiliation information from a network.

3 CREATING AFFILIATION NETWORKS

In the last section, it was pointed out that for a given one-mode graph, there may be many two-mode graphs that project to it. Finding a useful representation among the different possibilities is not a trivial question. The simplest answer, originally proposed by Guillaume and Latapy (2004), is to look for a minimal two-mode representation whose projection is the network we are modeling. We measure a two-mode representation by the size of its top set, so a minimal two-mode representation has a minimal top set. Although finding this representation is a computationally difficult problem (*NP*-hard), in practice, the heuristic method of Guillaume and Latapy (2004) works well. Their basic method is to take each edge in the network and then find a maximal clique containing that edge. By collecting together the set of maximal cliques and assigning each one as an affiliation, a good approximation of a minimal two-mode representation is created. This heuristic works efficiently for many real-world networks because power-law distributed networks tend to contain relatively small cliques.

One difficulty in analyzing the affiliations in a network is that this straightforward clique decomposition scheme does not work well if information is missing about the network. If edges are missing in the network representation, the number of affiliations found may be much larger than the true value. In Cloteaux (2010), a scheme was presented to overcome some of these difficulties. Instead of looking for simple cliques to cover the network, an alternative structure, the *k-plex*, is used. A k-plex is essentially a clique that allows for a certain number edges to be missing from it. A graph $G = (N, E)$ is defined as a k-plex if the minimal node degree in G, denoted as $\mu(G)$, is no more than k less than $|N|$. In other words,

$$\mu(G) \geq |N| - k. \tag{1}$$

From this definition, since 1-plex cannot be missing any edges, it is therefore a clique. Thus, a k-plex is a strict generalization of cliques. For the analysis in the remainder of this paper, we use 2-plexes to match with affiliations. This was done under the assumption that few links in the networks we are analyzing are missing.

A difference between the method in Cloteaux (2010) and the approach in this paper is in how we create minimal affiliation sets from the collected 2-plexes. As k becomes larger, the number of maximal k-plexes for a given edge also tends to increase. In order to accommodate this increase, in the original approach all possible maximal 2-plexes were collected and then analyzed in order to compute a minimal affiliation set. In our approach, we do not collect all possible 2-plexes. Instead, we only save the maximal 2-plex for each edge which has the smallest conductance.

The *conductance* of a subgraph is a clustering metric introduced by Kannan, Vempala, and Vetta (2004) It compares the number of edges on the border of a subgraph versus the number edges within the subgraph. For a subset S in the network G, the conductance is defined as

$$\phi(S, G) = \frac{C_S}{2 \cdot M_S + C_S} \tag{2}$$

where M_S is the number of edges within S and C_S is the number of edges on the boundary of S. Intuitively, conductance captures how "self-contained" a subgraph is compared to the rest of the graph. An example for computing the conductance of a subgraph is shown in Figure 2. Of the maximal k-plexes we find, we use this measure to decide which ones form the most natural clustering and then assign those as affiliations.

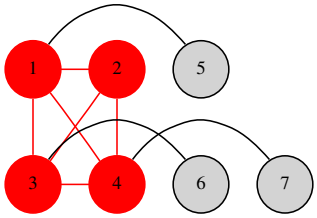


Figure 2: In this example, a subgraph S is marked by red nodes. The conductance of the subgraph is then computed from $M_S = 6$ and $C_S = 3$ for a value of $\phi(S,G) = \frac{3}{2\cdot 6+3} = 0.2$.

4 CONSTRUCTING CONCEPT LATTICES

When constructing a hierarchy of the entities in the network, a principal reason for us to examine the affiliations of the network is to use the well-established theory for converting these into lattices. This area, called Formal Concept Analysis (FCA), is used to generate concept (or Galois) lattices. We give a brief overview of FCA in order to establish the connection between affiliation mining and hierarchy building. In our discussion, we take our FCA definitions from Davey and Priestley (2002).

In FCA, the basic input is the triple $(\mathscr{G},\mathscr{M},\mathscr{I})$ called a *context*. The set $\mathscr{G}$ are called the *objects*, while the set $\mathscr{M}$ are the *attributes*. Relating the context to the affiliations of a network, $\mathscr{G} = \bot$, and $\mathscr{M} = \top$. The value $\mathscr{I}$ is a binary relation between objects and attributes; in our case, $\mathscr{I} = E$.

For the affiliation network $(\top,\bot,E)$ and a subset of entities $A \subseteq \bot$, we define the set

$$attr(A) = \{m \in \bot | \forall g \in A \text{ where } (m,g) \in E\} \tag{3}$$

This is the set of attributes (or, in this case, affiliations) that are common to every member in A. For a subset of affiliations B, we define the set

$$obj(B) = \{g \in \top | \forall m \in B \text{ where } (m,g) \in E\} \tag{4}$$

This is the set of common objects (or entities) to every affiliation in B. A *concept* is then a pair (A,B) where $A \subseteq \bot$ and $B \subseteq \top$ such that $attr(A) = B$ and $A = obj(B)$. In other words, a concept is a set of entities and affiliations which are closed to each other. The set of all concepts for an affiliation network is denoted as $\mathscr{B}(\top,\bot,E)$.

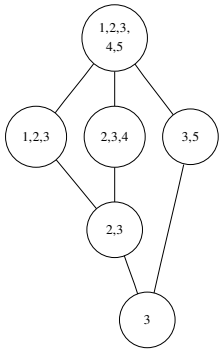
For any two concepts (A_1,B_1) and (A_2,B_2) in a set $\mathscr{B}(\top,\bot,E)$, we can order the concepts using subset inclusion of the objects. Formally,

$$(A_1,B_1) \leq (A_2,B_2) \Leftrightarrow A_1 \subseteq A_2 \tag{5}$$

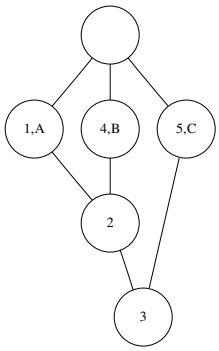
Under this ordering, the set of concepts forms a complete lattice (called a *concept lattice*). Concept lattices are a valuable tool for analyzing data across many domains. In our context of ordering the entities in a network hierarchically, a major advantage of using a lattice is that an entity can appear under multiple affiliations. In other words, we can capture shared structure for entities using the concept lattice.

1,2,3,4,5	∅
1,2,3	A
2,3,4	B
3,5	C
2,3	A,B
3	A,B,C

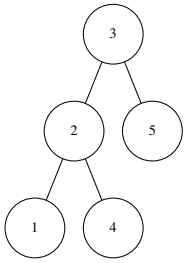
(a) Table of concepts



(b) Hasse diagram of concepts



(c) Hasse diagram with labels



(d) Hierarchy

Figure 3: This figure shows the formal concept analysis for the affiliation network in Figure 1. In Figure 3a, the concepts for the affiliation network are shown. The Figure 3b shows a Hasse diagram of the partial ordering of the concepts. In Figure 3c, the labels on the node represent the lowest concept in the ordering with a particular node and the highest concept with a particular affiliation. The lower an entity is in the diagram, the more general the class of affiliations it has. In this example, node 3 is in every affiliation, while node 2 is in every affiliation that 1 and 4 are in. By inverting the Hasse diagram, we get the more traditional hierarchy of the nodes as shown in Figure 3d .

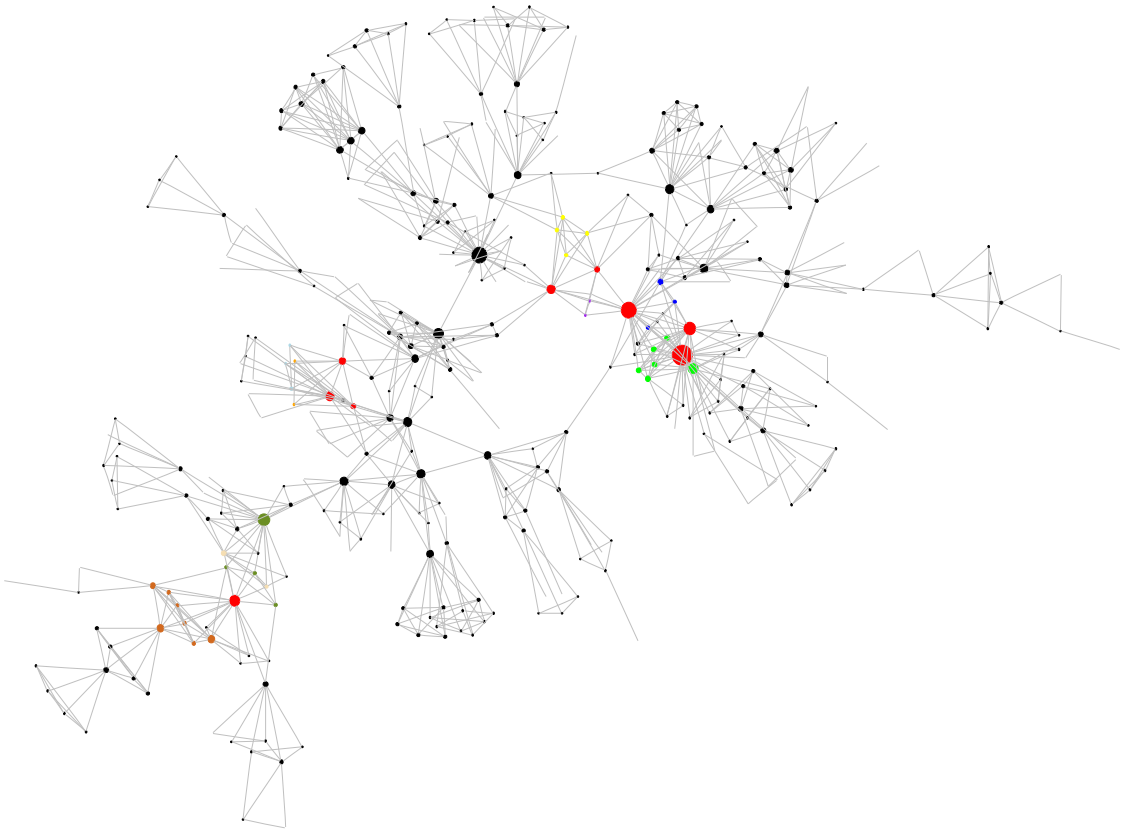
An example of converting the information from an affiliation network to a concept lattice is shown in Figure 3. In this case, the affiliation network of Figure 1 is converted into a partial ordering where we can easily see that node 3 has a central role in the network. In fact, all affiliations in the network have node 3 as a member, which is shown in Figure 3b. We can understand how the entities and affiliations are related in the lattice by examining the highest node that an entity occurs and the lowest node that an affiliation occurs. This type is labeling is shown in Figure 3c. Using this Hasse diagram, we have placed the nodes into a type of hierarchy. By inverting this diagram, we get a traditional hierarchy diagram for the member of the network as shown in Figure 3d.

5 A SIMPLE CASE STUDY

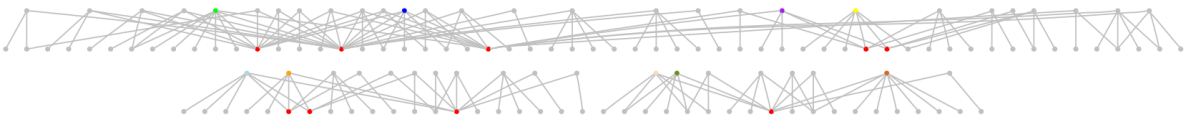
To see some advantages of this approach, we show an example of examining a network's hierarchical structure through extracted affiliations. The data set we examine is a collaboration network of co-authorships for network theory and experimental papers ending in 2006. This data was collected by Newman (2006), and we only use the largest connected component of 379 authors within it. The overall structure of the network is shown in Figure 4a.

In Figure 4b, we see a subset of the affiliation network extracted from the co-authorship network. In this bipartite representation, the top set of nodes are the affiliations in the set $\top$. The nodes marked in color correspond to large affiliations (in this case, affiliations with at least 6 authors) in the network. The bottom row is the set of entities in the network. The entities marked in red are those that share multiple large entities. We picked out these entities as those nodes provide connecting links between various research groups (or bridges between components within the network).

In Figure 4a, we see a diagram of the original network where the radius of each node is proportional to its degree. In other words, the more an author collaborates, the larger the node representing that author. The nodes marked in red correspond to the bridge nodes from the affiliation network. In this context, an affiliation roughly corresponds to a research group and the bridge nodes represent authors who have published with multiple large research groups. An examination of the curriculum vitae for these authors



(a) Collaboration graph with bridge nodes and nodes in large affiliation marked



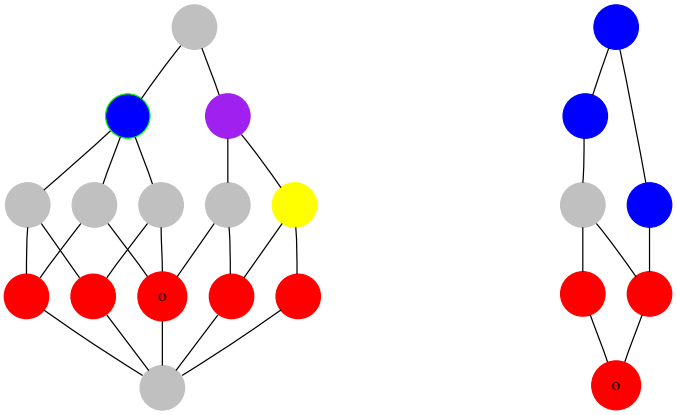
(b) Affiliation graph

Figure 4: Figure 4a shows the largest component of a collaboration graph. In Figure 4b, a partial representation of the extracted affiliations within the collaboration graph is given. The bottom nodes represent authors, while the top nodes are affiliation groups. The bottom nodes marked in red are authors with multiple large affiliations (in this case, we take affiliation groups of at least six authors as a large group). The large affiliations are marked in unique colors.

yields that the creation of these bridge nodes tends to be from job changes between groups. Often, this is the result of the author being a postdoctoral student.

In many contexts, bridge nodes in a network are essential to find and study. These nodes tend to have a large influence across the entire network. In the co-authorship network, their effect amounts to the sharing of ideas and techniques across multiple research groups. In a gene-expression network, these genes affect multiple functions which, if defective, can lead to disease with symptoms across functions.

To understand how the affiliations relate to one another, we also mark the affiliations in Figure 4a. In this figure, the colored nodes are all members of the large affiliations in the affiliation graph of Figure 4b. The colors correspond with membership in the affiliations with the same color.



(a) Hasse diagram of the relationship of five bridge nodes

(b) Hasse diagram of the members from one affiliation

Figure 5: This figure shows the Hasse diagrams extracted from subsets of the collaboration network. In Figure 5a, the concepts involving only the five bridges shown in the top line of Figure 4b are shown. The colored nodes show the associated affiliations in the diagram. In this case, the blue node outlined in green denotes that both affiliations are equivalent under this subset. In Figure 5b, the concepts for the entities in the blue affiliation are shown. The red nodes are the bridge entities, while the blue represent the other members.

From the extracted affiliations, we derived the concepts and performed formal concept analysis on them to understand the relationship certain entities Because of the size of the network, we looked at specific subsets in the network. Two of these subsets are shown in Figure 5. In the first figure, we took five nodes that had been identified as bridge nodes, and then performed FCA on the set of affiliations common to those nodes. The resulting lattice shows that the blue and green affiliations look identical under this analysis. Also, the yellow affiliation is a subset of the purple affiliations. Thus, under this view, only the middle bridge node (marked with a center black circle) connects truly different large affiliation groups.

In the second view (Figure 5b), we examine the entities in the blue affiliation group. The red nodes are the bridge nodes in the affiliation group, while the blue are the non-bridge nodes. As expected, we see the bridge nodes as higher in the hierarchy. As Figure 5b shows, one of these nodes (marked with a center black circle) dominates the affiliation. For this particular case, the affiliation corresponds to a research group and the marked node is the leader of the group.

6 CONCLUSIONS

Being able to extract hierarchies inherent in a network is necessary for the accurate modeling of these networks. While there has been recent research into the automatic extraction of these hierarchies, none of these proposed methods allow for entities to share structure across the derived hierarchies. We propose a method for allowing this type of analysis. Our method is automated, allowing for the data analysis of relatively large networks.

Our method is based on the extraction of affiliations within the network. Affiliation analysis is a promising branch of research with its connections to the structural aspects of how a network evolves. Finding and understanding the underlying affiliations seems to be an important direction in creating network models. We have extended this idea by examining the affiliations for the hierarchical structure in the network. As our example shows, using affiliations we are able to perform analysis on finding hierarchies of subsets of the network, understanding how affiliations relate to one another for these subsets, understanding how entities relate to one another within an affiliation, and finding the entities that have multiple memberships to large affiliations.

Future directions for our method include the analysis of gene and protein networks. By automating the process of hierarchy extraction in these networks, we hope to be able to compare hierarchies from multiple networks and create an automated ontology creation process for the members of these networks.

The principal questions left with our approach involve the accurate identification of the affiliations inherent within the network. A promising approach to identifying these affiliations is examining the network as it dynamically evolves in time. Another approach to examine is the use of other hierarchy methods. We are seeing how well provably good methods, such as Räcke decomposition, can be merged into our approach.

REFERENCES

Barabási, A.-L., and R. Albert. 1999. "Emergence of scaling in random networks". *Science* 286:509–512.

Bienkowski, M., M. Korzeniowski, and H. Räcke. 2003. "A practical algorithm for constructing oblivious routing schemes". In *Proceedings of the fifteenth annual ACM symposium on Parallel algorithms and architectures*, edited by A. Rosenburg and F. Meyer auf der Heide, SPAA '03, 24–33. New York, NY, USA: ACM.

Clauset, A., C. Moore, and M. E. J. Newman. 2008, May. "Hierarchical structure and the prediction of missing links in networks". *Nature* 453 (7191): 98–101.

Cloteaux, B. 2010, December. "Modeling affiliations in networks". In *Proceedings of the 2010 Winter Simulation Conference*, edited by B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, 2958–2967. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Davey, B., and H. Priestley. 2002. *Introduction to lattices and order*. Cambridge mathematical textbooks. Cambridge University Press.

Guillaume, J., and M. Latapy. 2004, June. "Bipartite structure of all complex networks". *Information Processing Letters* 90 (5): 215–221.

Guillaume, J., and M. Latapy. 2006, November. "Bipartite graphs as models of complex networks". *Physica A: Statistical and Theoretical Physics* 371 (2): 795–813.

Harrelson, C., K. Hildrum, and S. Rao. 2003. "A polynomial-time tree decomposition to minimize congestion". In *Proceedings of the fifteenth annual ACM symposium on Parallel algorithms and architectures*, edited by A. Rosenburg and F. Meyer auf der Heide, SPAA '03, 34–43. New York, NY, USA: ACM.

Kannan, R., S. Vempala, and A. Vetta. 2004, May. "On clusterings: Good, bad and spectral". *J. ACM* 51:497–515.

Latapy, M., C. Magnien, and N. D. Vecchio. 2008, January. "Basic notions for the analysis of large two-mode networks". *Social Networks* 30 (1): 31–48.

Lattanzi, S., and D. Sivakumar. 2009. "Affiliation networks". In *Proceedings of the 41st annual ACM symposium on Theory of computing*, edited by M. Mitzenmacher, 427–434. Bethesda, MD, USA: ACM.

Leskovec, J., J. Kleinberg, and C. Faloutsos. 2005. "Graphs over time: densification laws, shrinking diameters and possible explanations". In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, edited by R. Grossman, R. Bayardo, and K. Bennett, 177–187. Chicago, Illinois, USA: ACM.

Newman, M. E. J. 2006, September. "Finding community structure in networks using the eigenvectors of matrices". *Phys. Rev. E* 74 (3): 036104.

Räcke, H. 2002. "Minimizing congestion in general networks". In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, edited by P. Raghavan and B. Chazelle, 43–52.

Sales-Pardo, M., R. Guimer, A. A. Moreira, and L. A. N. Amaral. 2007. "Extracting the hierarchical organization of complex systems". *Proceedings of the National Academy of Sciences* 104 (39): 15224–15229.

The Gene Ontology Consortium 2008. "The Gene Ontology project in 2008". *Nucleic Acids Research* 36 (suppl 1): D440–D444.

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

Zheleva, E., H. Sharara, and L. Getoor. 2009. "Co-evolution of social and affiliation networks". In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, edited by J. Elder, F. Soulié-Fogelman, P. Flach, and M. J. Zaki, 1007–1016. Paris, France: ACM.

AUTHOR BIOGRAPHIES

BRIAN CLOTEAUX is a computer scientist in the Mathematical and Computational Sciences division at the National Institute of Standards and Technology. He holds a PhD degree in Computer Science from New Mexico State University. His email address is brian.cloteaux@nist.gov.