

Visualizing Hierarchical Social Networks

Kurtuluş Gemici¹ and Anthony Vashevko¹

Socius: Sociological Research for
a Dynamic World
Volume 4: 1–12
© The Author(s) 2018
Reprints and permissions:
sagepub.com/journalsPermissions.nav
DOI: 10.1177/2378023118772982
srd.sagepub.com



Abstract

The authors propose a novel technique for the visualization of networks that contain a hierarchical structure: networks in which certain nodes and groups of nodes can be classified through a relation of precedence. Networks with a hierarchical structure frequently arise in sociology and various other disciplines, but the existing methods for visualizing such networks leave much to be desired. The method developed in this work builds on the tradition of visualization in social network analysis; it aims to simultaneously represent the positions of different nodes and the relationships between groups containing the nodes in the network. As such, the proposed visualization method facilitates theoretical and empirical analysis of social structures by algorithmically combining information from the underlying network with the information from the hierarchical structure of the network. The authors illustrate the proposed method with social networks examined through cohesive blocking and *k*-core decomposition.

Keywords

visualization, social networks, cohesive blocking, financial networks, hierarchical networks

Networks that contain a hierarchical structure—networks in which certain nodes and groups of nodes can be classified through a relation of precedence—are ubiquitous in structures observed in various fields, including domains as different as firm ownership relations, scientific collaboration, and protein interaction chains (Clauset, Moore, and Newman 2008; Girvan and Newman 2002; Grabowski and Kosiński 2004; López, Mendes, and Sanjuán 2002; Mani and Moody 2014; Newman 2006). In this article, we contribute to social network analysis and data visualization in sociology (Correa and Ma 2011; Freeman 2000, 2005; Healy and Moody 2014; Krempel 2009; Moody, McFarland, and BenderdeMoll 2005) through a novel algorithm for visualizing hierarchical networks. In developing this method of visualization, we aim to simultaneously represent the positions of different nodes and the relationships between groups containing the nodes in the network. We illustrate, through substantive examples, that the algorithm we develop is an effective tool in the exploration and presentation of hierarchical networks. Although we focus on the visualization of social networks, the algorithm we develop is general and can be used to study networks outside the domain of sociology.

Our motivation in developing a new method to visualize hierarchical social networks is threefold. First, at the most general level, we believe that visualization is an essential aspect of scientific practice (Arnheim 1969; Taylor 1966) and that it complements measurement as well as algebraic methods (Diestel 2005; Gibbons 1985; Harary 1969; Newman,

Barabási, and Watts 2006) in the exploration and explanation of patterns (Brandes, Kenis, and Raab 2006; Crosby 1997; Tufte 2001). Visualization has a long tradition in social network analysis (Freeman 2000), going back to Moreno's (1932, 1934) work during the 1930s and his use of sociograms to represent connections among various actors. Over the years, visualization has developed in tandem with social network analysis. Some of the most illuminating applications of visualization, such as Kadushin's (1974) work on intellectual elites and Freeman and White's (1993) use of Galois lattices, were concerned with simultaneous representation of social positions and social groups. Second, as we illustrate below, the simultaneous visualization of a social network and the hierarchical structure of the network is an effective tool in capturing the complex interplay between social positions, groups, and the relationships among groups. Third, at the most practical level, we aim to understand the structure of financial networks around the world. The method we employ to examine groups and positions of different financial actors is cohesive blocking (Moody and White 2003; White and Harary 2001), which provides a rich set of information on the

¹Department of Sociology, National University of Singapore, Singapore

Corresponding Author:

Kurtuluş Gemici, National University of Singapore, Faculty of Arts and Social Sciences, Department of Sociology, 11 Arts Link, 03-25, 117570, Singapore.

Email: gemici.kurtulus@gmail.com



Creative Commons Non Commercial CC BY-NC: This article is distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 License (<http://www.creativecommons.org/licenses/by-nc/4.0/>) which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

structural cohesion and embeddedness patterns in financial networks. In our experience, visualization is an indispensable tool in understanding and explaining the multidimensional structure of networks revealed by the cohesive blocking analysis.

The technique we develop in this article builds on a voluminous literature in computer science and social network analysis (Ahajjam, El Haddad, and Badir 2018; Holten 2006; Nikolaev, Razib, and Kucheriya 2015). Harel (1988) is one of the first works to formalize the simultaneous representation of a relation between nodes and structural relations between sets of nodes in a graph. Since then, various authors have offered solutions such as elastic hierarchies (Storey and Muller 1995) and hierarchical multiperspective views (Zhao, McGuffin, and Chignell 2005) to this visualization problem. A recent influential approach is hierarchical edge bundling (Holten 2006), which bends and groups adjacency relations by inclusion relations among sets of nodes in a graph. Edge bundling in radial layouts is particularly effective in representing large graphs, and it has already found sophisticated application in social network analysis (Crnovrsanin et al. 2014). Our contribution differs from these approaches in its emphasis on representing inclusion and exclusion relations that are essential to understanding the structure of a social network without losing sight of the overall distribution of relations among nodes. As we show, the method we offer is particularly effective in visualizing hierarchical graphs produced by techniques such as k -core and k -shell decomposition (Alvarez-Hamelin et al. 2005; Carmi et al. 2007; Miorandi and de Pellegrini 2010), community detection (Girvan and Newman 2002), and cohesive blocking (Moody and White 2003).

Throughout this article, we use several conventions to refer to the various objects under consideration. We define a graph N as the unordered pair (V, E) where V is the set of vertices and E is the set of edges in N . Each edge $e \in E$ belongs to two-element subsets of V . We denote the edge from vertex i to vertex j as e_{ij} . In social network analysis, a graph N is often discussed by using the terminology of network, nodes (e.g., actors), and ties (e.g., social relations between actors) (Diestel 2005; Wasserman and Faust 1994). We predominantly use the social network terminology. A trivial network is a network consisting of a single node. Take any two nodes v_i and v_j in a network N . An alternating sequence of nodes and edges in N connecting v_i and v_j constitutes a *walk*. When the nodes and edges are distinct, it is called a *path*. A cycle is a walk where the beginning and end nodes are the same. Disjoint paths in N are paths where no edges and no nodes except v_i and v_j are common. A *cutset* C of N is defined as the set of nodes that separates N into two disjoint networks X and $\bar{X}$, which are called the cuts of N induced by C . Following Harary (1969), we define k -connectivity of a network N as the minimum number of nodes whose removal will result in a disconnected or trivial network. A tree is a network that does not contain any cycles and where each pair of nodes are connected by a unique path.

In analyzing hierarchical networks, we restrict our focus to hierarchies in which the relationship is containment. Thus, the nodes of the hierarchy (i.e., supernodes) in a branch are distinct. For two nodes that are related to each other, one of them is a strict subset of the other. We refer to the primary network as G and to an individual node and the set of nodes as v and V , respectively. We refer to the associated hierarchy as T , with nodes (or supernodes) denoted by t . Each node in G belongs to some supernode in T : the membership of a node v is given by $t(v)$. The *members* of t are the nodes in G belonging to t , that is, all v 's for which $t(v) = t$; the *children* of t are the other nodes in T that are direct descendants of t .

This article is structured as follows. We first describe the cohesive blocking technique, a network analysis method for which this visualization is particularly helpful. The next section describes the layout algorithm in detail. In the following section we present several examples that illustrate the visualization technique at work. The final section concludes the article with a discussion of the uses of this technique and avenues for further research.

Cohesive Blocking

Social structure as the patterned “crystallization of relationships”—and, once crystallized, a sui generis entity shaping social action—is a fundamental notion in sociology (Durkheim [1893] 1964; Giddens 1979; Lizardo 2010; Martin 2009:1–2; Simmel [1908] 1950:94–95). Cohesive blocking (Moody and White 2003) is a technique to formally analyze social structures through social network data. This technique builds on two concepts with deep roots in sociological thought, social cohesion—the binding of social actors into a collectivity (Durkheim 1964; Fantasia 1988; Hechter 1987; Meyer and Kimeldorf 2015; Simmel [1922] 1964)—and embeddedness (Granovetter 1985; Zukin and DiMaggio 1990). Classical sociologists such as Durkheim (1964), Weber (1978), and Tönnies ([1887] 1957) emphasized the structural, ideational, and affective elements in their approaches to social cohesion. The problem is that these different dimensions of social cohesion are analytically separate and can at times exercise effects in opposite directions. Instead of such an approach, Moody and White (2003) focused on structural cohesion by defining it through the relational togetherness of a group. Apart from gains in analytical precision, such a focus also enables an effective correspondence between the substantive sociological idea (i.e., relational togetherness) and the mathematical formalism used to examine social cohesion. The key to this effective correspondence is Menger’s theorem.

Menger’s theorem states that there is a crucial relationship between the number of nodes separating any two nodes s and t in a network and the number of disjoint s - t paths in N (Bondy and Murty 1976). Whitney (1932a, 1932b) offered a criterion for connecting the notion of k -connectivity to the number of disjoint paths in a network, which can be generalized to the

whole network. Formally, N is k -connected if and only if it contains k independent paths between any two nodes (Diestel 2005). Now, denote any group of nodes connected by at least k paths in N as k -connected and a k -component of N . Menger's theorem and its extensions imply that a k -component is a component with no cutset fewer than k nodes (Diestel 2005; Harary 1969; Moody and White 2003).

The mathematical formalism of graph connectivity, encapsulated in Menger's theorem and its extensions by various authors, offers a rich opportunity to express the concept of social cohesion with reference to the cohesion of a social structure. Namely, as Moody and White (2003:109) suggested, the concept of structural cohesion can be defined with reference to the minimum number of nodes that constitute a cutset and the minimum number of "independent relational paths" connecting any pair of nodes in a social structure. Then, each set of nodes with k -connectivity in a network, a k -component, constitutes a group with a defined degree of structural cohesion. Furthermore, each k -component in a network may contain other groups that possess their own degrees of cohesion. Moody and White called such groups, each of which possesses a measurable degree of cohesion and embedded in other groups with varying degrees of cohesion, the *cohesive blocks* of a network.

It can be shown that the nestedness of such groups, the *embeddedness* of groups in a network (Granovetter 1985), generates a tree, in which each node is a cohesive block consisting of nodes held together through k independent relational paths and in which the children of the block are cohesive groups nested in that block.¹ Nestedness follows from the fact that the components generated by a k -cutset might have stronger cohesion and thus higher k -connectivity. The tree structure generated by the nestedness relationship between cohesive blocks, T , represents the hierarchy that we aim to visualize. The embeddedness of a cohesive block is defined as the depth level of the block in T .

The tree structure obtained by the cohesive blocking technique can be illustrated with the karate club social network of Zachary (1977). The karate club social network data come from Zachary's ethnographic research on the interactions between members of a voluntary karate club. In this network, each tie represents two members who interact on a repeated basis outside the club itself (see Figure 1). In the original study, Zachary studied the conflict within the club between the karate instructor (node 1) and the club president (node 34) by assigning a numerical value ("capacity") to each tie on the basis of the number of distinct contexts in which two

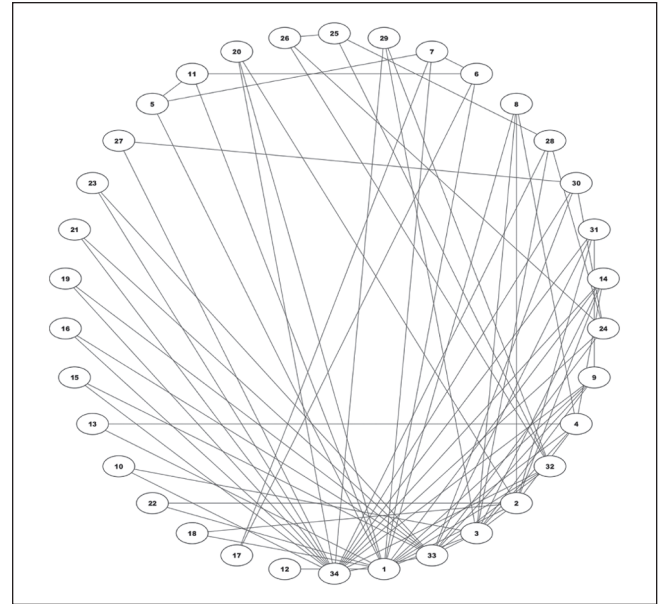


Figure 1. The Zachary karate club network.

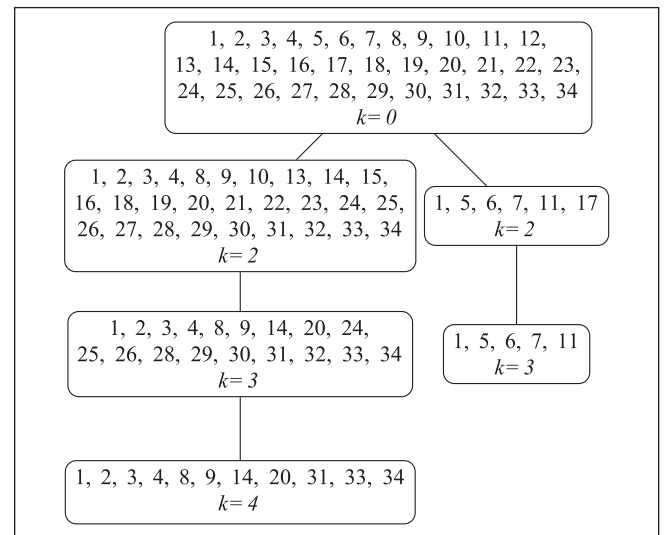


Figure 2. Cohesive blocks tree for the Zachary karate club.

individuals interact. Zachary used the maxflow-mincut algorithm of Ford and Fulkerson (1962) to predict the constituents of each faction after the eventual division of the club. The cohesive blocking technique does not use the flow data, which were crucial in the original study for predicting the different factions of the karate club. Despite this limitation, the cohesive blocking hierarchy generated by this technique can be used profitably to analyze the structural divisions in the karate club. However, such a task requires examining the underlying network data and the density of ties between different members (see Figure 1) in conjunction with the hierarchy tree (see Figure 2). Our goal is to facilitate such analysis by algorithmically combining information from the

¹For details, refer to Moody and White (2003:123–24), who offered an algorithm to generate all the cohesive blocks in a network. Their algorithm extends the works of Kanevsky (1993), who combined ideas from Even and Tarjan (1975) and Provan and Shier (1996) for finding all minimum $(s-t)$ -cutsets in a network.

underlying network data with the information from the hierarchy tree.

Layout Algorithm

Our goal is to simultaneously visualize both the hierarchical structure of T as well as the structure of the underlying network G . Although many algorithms exist for laying out networks (Fruchterman and Reingold 1991; Jacomy et al. 2014; Kamada and Kawai 1989) and for laying out hierarchies (Abello 2004; Abello, Kobourov, and Yusuf 2004; Eades and Feng 1996; Frishman and Tal 2004; Galili 2015; Johnson and Shneiderman 1991; Urbanek 2008), our review of the literature suggests no standard algorithms for simultaneous visualization of both aspects of a network (Chen, Härdle, and Unwin 2008; Di Giacomo and Lubiw 2015; Tamassia 2013). The hierarchical edge bundling approach of Holten (2006), despite its utility, does not meet one of our main requirements: representing the inclusion-exclusion patterns at different levels of the hierarchy. In addition, we discovered no standard ways to extend existing algorithms to fulfill this goal. In particular, because a key goal of our visualization is to ensure that the layout of the network respects the hierarchical boundaries, we needed an algorithm that separated distinct components of the hierarchy and ensured that nodes stayed inside hierarchical boundaries.² We found no standard algorithm that could impose such geometric constraints. In fact, many layout algorithms impose no particular constraint on the drawing area at all. The approach we ultimately adopt does, however, extend prior ideas in this field, particularly the boundary-repulsive force of Davidson and Harel (1996).

Under certain circumstances, these two layout goals, representing the hierarchical structure as well as the underlying network, can conflict. A hierarchy tree in which nodes are related to one another through subset or superset relationship can be directly represented on the Cartesian plane. However, nodes of the underlying network G might belong to multiple supernodes in T . For instance, in cohesive blocking, the nodes in a cutset end up on both sides of the cut induced by the cutset (Harary 1969). Sociologically, the overlaps between the different supernodes of the hierarchy tree are what gives cohesive blocking its relational focus (Emirbayer 1997) because it enables the identification of “groups in terms of sets of relationships” rather than simply “sets of individuals” (Moody and White 2003:111). These nodes, which act as articulation points between the hierarchy branches, have important

substantive implications in other disciplines as well. For example, they are crucial in the study of transportation networks and reliability analysis (Gibbons 1985:58; Jensen and Bellmore 1969:171–72; Nagamochi and Ibaraki 2008).

In these cases, we opt to duplicate nodes that fall into multiple blocks, placing a copy into each block along with that node’s connections to the rest of the network. Admittedly, we are motivated primarily by the cohesive blocking technique, where the cutset nodes are key determinants of structural cohesion (Moody and White 2003). However, we also believe that this approach is sufficiently general because it emphasizes that such nodes play an important bridging role in the network, helping bind otherwise distant branches of the hierarchy. Nonetheless, it should be noted that this approach would break down for networks where such nodes are the rule rather than the exception.

We combine three techniques to simultaneously represent the hierarchical structure as well as the underlying network. First, we use a standard hierarchy visualization algorithm—the “squarified treemap” algorithm (Bruls, Huizing, and Van Wijk 2000)—to partition the visual area for the hierarchy. Second, we use a nesting offset (Johnson and Shneiderman 1991), geometric subsets, and shading to represent hierarchy. Third, we construct and optimize an energy function to lay out network nodes within these partitions. The combination of these techniques produces graphs that reveal how network structure interacts with the hierarchical structure. Below, we discuss the algorithmic details of our approach.

Hierarchy Layout: Treemap

Visualizing complex and multidimensional information such as a hierarchical network requires close attention to several issues. First, space should be used efficiently, which implies that empty areas of the visual representation should be minimized without compromising informational accuracy. Second, the chosen method should offer means to represent the order relationship among the different components of the network in an unequivocal manner. Third, it is often desirable to represent the importance of different components (e.g., size of a network component) in a way that does not hinder the representation of the hierarchical structure.

Meeting all the three requirements is surprisingly hard. The most common way of representing trees as a directed graph growing in a single direction starting from a root node is space inefficient, as can be seen in Figure 2. Furthermore, it is difficult to parse when the tree is large.

Treemaps (Johnson and Shneiderman 1991; Shneiderman 1992; Shneiderman and Wattenberg 2001; Wood and Dykes 2008) offer a solution that meets all of these requirements. Since their introduction (Johnson and Shneiderman 1991), treemaps have become a standard tool for illustrating hierarchically structured information (e.g., file sizes within a hierarchy such as folders on a hard drive), and the popularity of the original algorithm has generated a family of algorithms

²Moody and White (2003) visualized cohesive blocks by first laying out nodes and then drawing a border around nodes in the same cohesive block. Although effective in visualizing small networks manually, their approach has two shortcomings when it comes to visualizing complex networks. First, standard layout algorithms cannot guarantee that vertices in the same block stick together. Second, the overlaps between multiple blocks create considerable visual noise in most networks with more than 20 nodes.

inspired by the treemap approach. This family of algorithms take as input a tree and recursively partition a visual area into regions whose areas are proportional to the size of their corresponding subtrees. The key idea is to map “the display space into a collection of rectangular bounding boxes representing the tree structure” (Johnson and Shneiderman 1991:284). The most important advantage of the treemap family of algorithms is that they are space filling: these algorithms use all the available two-dimensional space (Shneiderman 1992).

The most basic version of the treemap family of algorithms partitions areas into rectangles. However, such an approach results in rectangles with large aspect ratios: a common problem is the representation of various nodes in the tree as rectangles that are too thin. The “squarified treemap” variant (Bruls et al. 2000) avoids this problem. Squarified treemap algorithm attempts to make the rectangles as square as possible.³ The advantage of the squarified treemap algorithm is that the bounded rectangles representing tree nodes are much easier to parse and compare.

To represent the hierarchy and thus nestedness of a child supernode in parent supernodes, we go back to the method of nested treemaps in the original study of Johnson and Shneiderman (1991). Nested treemaps represent hierarchy by placing nodes into regions corresponding to the deepest level of the hierarchy in which the nodes occur. A nesting offset and geometric subset relationship become the instruments to mark hierarchy and separation between nodes: deeper levels of the hierarchy are laid out strictly inside the boundaries of their parents. Furthermore, shading of deeper blocks can emphasize hierarchical depth. Although a nesting offset and shading are not strictly necessary for small networks, they become crucial tools as the network grows in size and complexity. Thus, we combine the squarified treemap algorithm with nesting and shading in our visualization algorithm; we place duplicated nodes once at the deepest level of each branch in which they occur.

One disadvantage of the nested offset approach is that nodes deep in the hierarchy have less space assigned to them than nodes near the root. In extreme cases, the offset can reduce the space available to position these nodes to zero. As such we augment the nested treemap algorithm by increasing the “weight” of deeper nodes. By making deep nodes take up more space in the tree, we can ensure that nodes throughout the hierarchy have an equal amount of room available to them in the final layout.

Algorithm 1 gives the general description of our procedure.⁴ In the situation presented here, we use the squarified

treemap algorithm to lay out T , with the area of a subtree proportional to the number of members in the subtree. Given that the squarified treemap algorithm is a standard tool and the algorithm is well understood, we assume its availability and refer the reader to Bruls et al. (2000) for the details. Starting from the root node of T and the available display area, the LAYOUT function recursively assigns each node t a partition $P(t)$ of the area.⁵ It relies on the auxiliary function SIZE to calculate subtree sizes. In addition, the PAD function ensures the offset required for nesting partitions, producing a subset $\tilde{P}$ of the assigned partition area P to lay out each subtree.

Algorithm 1. Layout of T Based on Subtree Sizes.

```

function LAYOUT( $t, P$ )
   $\tilde{P} \leftarrow \text{PAD}(P)$ 
   $\text{sizes} \leftarrow [w(t) \cdot |\text{members}(t)|] ++ [\text{SIZE}(c) \text{ for } c \in \text{children}(t)]$ 
   $\text{partitions} \leftarrow \text{SQUARIFIED TREEMAP}(\tilde{P}, \text{sizes})$ 
   $P(t) \leftarrow \text{partitions}[0]$ 
  for  $c \in \text{children}(t)$  do
    LAYOUT( $c, \text{partitions}[c]$ )
  end for
end function

function SIZE( $t$ )
   $\text{size} \leftarrow w(t) \cdot |\text{members}(t)|$ 
  for  $c \in \text{children}(t)$  do
     $\text{size} \leftarrow \text{size} + \text{SIZE}(c)$ 
  end for
  return  $\text{size}$ 
end function

```

Finally, we ensure that all nodes have an equal amount of space after nesting by repeating LAYOUT to calculate appropriate weights $w(t)$ for each part of the tree. Starting from an equal weight for each subtree, we iterate the layout procedure, updating weights after each iteration to ensure that the available area per node remains constant.

Network Layout: Energy Minimization

Armed with partitions of the display area, we proceed to lay out individual nodes using a modification of a force-directed graph layout algorithm (Davidson and Harel 1996; Fruchterman and Reingold 1991; Kamada and Kawai 1989). We construct an energy function representing two components: the first, repulsion by the boundaries of the partition; the second, forces reflecting the structure of the network.

Formally, energy is a function of node positions in the display area. We denote the position of node v by x_v . We lay

³Other variants exist to lay out the tree with different geometric shapes, as discussed later.

⁴An example implementation of the algorithm presented here as an open-source Python package can be found at <https://bitbucket.org/avashevko/cblayout>.

⁵We use square brackets to either define a list ($[\dots]$) or select an item from the list ($\text{list}[i]$), and we use $++$ to denote list concatenation.

out the graph by minimizing this energy function through stochastic gradient descent over the set of node positions.

Partition Energy. Each node v in G has some corresponding partition, given by $P[t(v)]$ or more concisely $P(v)$. Under the treemap algorithm, each partition is rectangular, so we define distances δ_{vt} , δ_{vb} , δ_{vl} , and δ_{vr} , representing the distance of x_v from the top, bottom, left, and right walls of the partition, respectively. Then, the partition energy for a node equals

$$E_P(x_v) = \begin{cases} -\log(\delta_{vt}\delta_{vb}\delta_{vl}\delta_{vr}) & \text{if } v \text{ inside } P(v) \\ \infty & \text{if } v \text{ outside } P(v) \end{cases}.$$

This function pushes nodes to the center of the rectangle. Partition energy for the overall graph is the sum of node energies:

$$E_P = \sum_{v \in V} E_P(x_v).$$

Although we define the energy function only for rectangles, the partition energy function trivially extends to other shapes such as circles or toroids. Nonconvex shapes such as toroids, however, perform poorly under stochastic gradient descent.

Network Energy. The second energy component represents the energy embedded in the attractive and repulsive forces of the network. We choose to adopt the Kamada-Kawai energy formulation (Kamada and Kawai 1989). In this model, a spring connects each pair of nodes in the network, with the ideal length of the spring proportional to the shortest-path distance between the nodes and the force of each spring given by Hooke's law. Writing the position of node i as x_i , this energy component is given by

$$E_N(x_i) = \sum_{j \in V, j \neq i} \frac{1}{2} k_{ij} (|x_i - x_j| - l_{ij})^2.$$

As before, network energy for the overall graph is the sum of node energies:

$$E_N = \sum_{v \in V} E_N(x_v).$$

The two constants l_{ij} , the ideal length of the spring, and k_{ij} , the spring strength constant, both depend on the shortest-path distance between nodes, denoted d_{ij} . For disconnected nodes, we set d_{ij} equal to the maximum observed shortest-path distance in the network. Following Kamada and Kawai, we set $k_{ij} = K/d_{ij}^2$, with K an adjustable parameter; we diverge by setting $l_{ij} = L_{ij} \cdot d_{ij}$. Kamada and Kawai kept $L_{ij} = L$ constant and proportional to the available display area. We instead adjust L_{ij} to reflect the area of the partition containing the lowest common ancestor (LCA) of $t(i)$ and $t(j)$:

$$L_{ij} = \text{area}(LCA(i, j)) / \max_{a, b \in LCA(i, j)} d_{ab}.$$

Relative to the Kamada-Kawai algorithm, this adjustment lengthens springs in fully connected and completely connected partitions in the network. This allows dense regions of the network to take up more space.

The overall energy function can then be written as the sum of the energy contributions of each vertex:

$$E(x) = \sum_{v \in V} E_P(x_v) + \sum_{v \in V} E_N(x_v).$$

To reiterate, energy is a function of the positions of each vertex in G . Because energy can be decomposed into the energy contributions of individual vertices, we minimize the energy of the graph using stochastic gradient descent with backtracking line search (Armijo 1966). Algorithm 2 describes the overall procedure. An initial randomization of positions distributes vertices within their partitions.

Algorithm 2. Stochastic Gradient Descent with Backtracking Line Search.

```

 $x_v \leftarrow \text{RANDOMIZE}(v)$ 
repeat
  for  $v \in \text{SHUFFLE}(V)$  do
     $\nabla_v \leftarrow \nabla E(v)$ 
     $\alpha \leftarrow \text{LINESEARCH}(E, \nabla_v)$ 
     $x_v \leftarrow x_v - \alpha \cdot \nabla_v$ 
  end for
until converged

```

In our experience, backtracking line search is necessary to ensure that nodes do not wander outside of their partitions. Backtracking line search looks for a minimum in the direction of the gradient, looking for the furthest point at which the gradient still approximates the energy function. Starting from an optimistically large step, the algorithm checks whether the decrease in energy is close to that expected by extrapolating the gradient; if it is not, the algorithm checks a closer point. Because the energy function becomes infinite if nodes leave their partitions, backtracking line search avoids putting nodes near boundaries.

Examples

We present four examples of the algorithm at work. In each of these examples, we represent the embeddedness level and cohesion (in that order) on the lower right corner of each cohesive block. First, Figure 3 presents a visualization of cohesive blocks in the karate club network (Zachary 1977). This figure reveals the nested structure of blocks. The highest level block, at depth 0, contains a single unique member (node 12), and all other nodes fall into subblocks. The main



Figure 3. Cohesive blocks in Zachary's karate club network.

branch (see the left half of Figure 2) falls into three blocks along the upper left of the figure, while the smaller branch falls into two blocks along the right of the figure. Notably, node 1, highlighted in green, is part of both branches.

The figure reveals two aspects of the conflict between the instructor and the club president (nodes 1 and 34). First, it reveals the split of the karate club into two factions around nodes 1 and 34. Apart from the most cohesive block at the core of the network, almost all the members are connected to the rest of the network through paths that are structured around either node 1 or node 34. Second, as a corollary, the figure suggests that the instructor and club president have distinct bases of support: while the club president has extensive ties to members just outside the deepest block, the instructor has extensive ties to blocks outside of the main branch of the cohesive block tree.

To be sure, some of the structural divisions in the karate club network can be captured through the simultaneous analysis of the network (see Figure 1) and the hierarchy tree (see Figure 2). For instance, it is clear from Figures 1 and 2 that members 5, 6, 7, 11, 12, and 17 are connected to the rest of the network mainly through the instructor. However, recognizing this pattern requires comparing the different branches of the hierarchy tree to identify the members who belong solely to a single branch of the tree. Furthermore, such a comparison is not always fruitful. For instance, members 13, 18, and 22 belong to the instructor faction and remained with the instructor after the split of the club. Yet this pattern can be identified only by looking into the network involvement of members 13, 18, and 22. Visualization in Figure 3 fills precisely this gap by simultaneously representing network involvement and the hierarchical structure of the network.

Our second and third examples are concerned with the social positions and groups in equity capital markets (ECMs). We use a commercial data set provided by Dealogic, one of the leading firms in financial data collection and dissemination. In ECMs, we focus on initial public offerings, follow-on issues, and transactions pertaining to debt instruments convertible to shares. Our analysis pertains to relations financial intermediaries build during underwriting activities in ECMs.⁶ One of our main objectives is to understand the social positions of different types of financial actors and how these positions change across financial centers and over time. Thus, we are interested in understanding social groups in ECMs and which groups are at the core of financial networks in ECMs. Cohesive blocking is a powerful tool in probing these issues.

Figure 4 presents the visualization of cohesive blocks in Singapore ECMs between 2008 and 2010. Although one of the largest global financial centers, Singapore is not a leading market in equity issues (Lee and Vertinsky 2011; Woo 2016). As a result, the network structure we observe is relatively simple, compared with other financial centers such as Hong Kong (see Figure 5). The social groups we observe here and the relations among them, however, is much richer than in the karate club network (Figure 3). Figure 4 reveals a number of interesting patterns. First, we find predominantly local and regional small institutions at the periphery of the network, comprising blocks at embeddedness levels 0 and 1. Second, as we get to the core of the financial network in Singapore (block at embeddedness level 2), we find a combination of Singaporean, Malaysian, Korean, Chinese, and Japanese financial institutions. Many of these financial intermediaries are midmarket or large firms taking advantage of dense trade and investment ties between Singapore and the rest of Asia. However, it can be seen that their network involvement depends on the core of the network. This is not surprising, as these firms often play a subsidiary role to transactions led by global or regional powerhouses that dominate the market. Finally, the deeply nested blocks at the core of the network are global bulge bracket banks (e.g., Citibank, Credit Suisse, and Goldman Sachs) and regional behemoths such as Singapore's DBS and Malaysia's Commerce International Merchant Bankers. At the core of the Singapore's ECM financial network, we find prominent institutions such as DBS, UBS, Citibank, and Credit Suisse (in green) taking the lion's share and binding much of the ECMs. Overall, Figure 4 shows that in financial markets such as ECMs cumulative advantage effects are quite strong (Poon 2003) and financial institutions from advanced industrialized countries dominate the market. Nonetheless, regional institutions play a substantial role in Singapore ECMs. This is a pattern that differs from a mature market

⁶We restrict our analysis to the most important actors in these transactions: bookrunners, leading managers, and global coordinators.

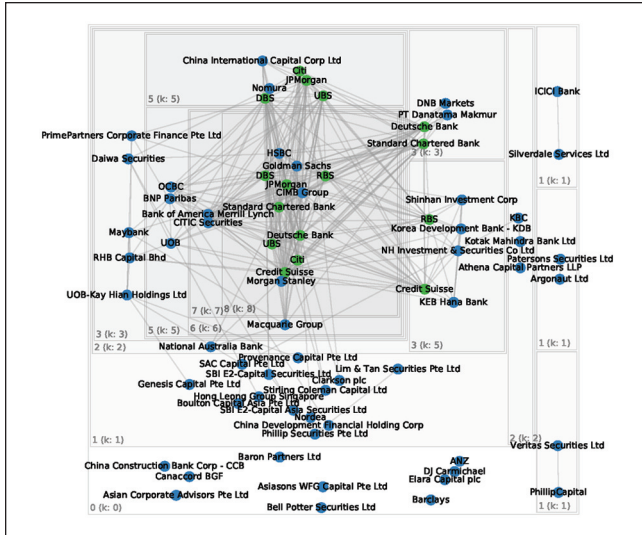


Figure 4. Cohesive blocks in the Singapore equity capital market financial network, 2008 to 2010.

such as Japan and many emerging markets, as we explore in other parts of our work.

Figure 5 presents the Hong Kong ECM financial network for the period from 2013 to 2015. Compared with Singapore, the financial network in Hong Kong ECMs are much denser, driven by Hong Kong's role as the main platform for Chinese firms to raise capital (Lee and Vertinsky 2011). Although the Hong Kong financial network has substantially less branching than that in Singapore, its cohesive block tree is much deeper. This network is much more hierarchical than either of the prior examples, with a core of institutions that operate in the deepest, most cohesive part of the network, and a series of institutions that are less and less affiliated with this central core. Because the hierarchy in this network features substantial depth but little branching, relatively few institutions serve as bridges within the network. Although we do not discuss it here, the Hong Kong ECM financial network reveals a highly competitive market in which the Chinese financial intermediaries play an increasingly larger role compared with global bulge bracket banks.

Last, Figure 6 presents an application of this visualization to a different network analysis technique and a larger network. The figure shows a hierarchy of friendship groups within a 2,587 member community in the National Longitudinal Study of Adolescent to Adult Health data set (Moody 2001), with the hierarchy derived through k -core decomposition (Alvarez-Hamelin et al. 2005). Instead of finding cohesive blocks, k -core decomposition identifies k -cores, subgraphs in which all vertices have degree at least k ; k -core decomposition can operate on larger graphs than cohesive blocking, and the visualization can keep up with these larger networks.⁷ The

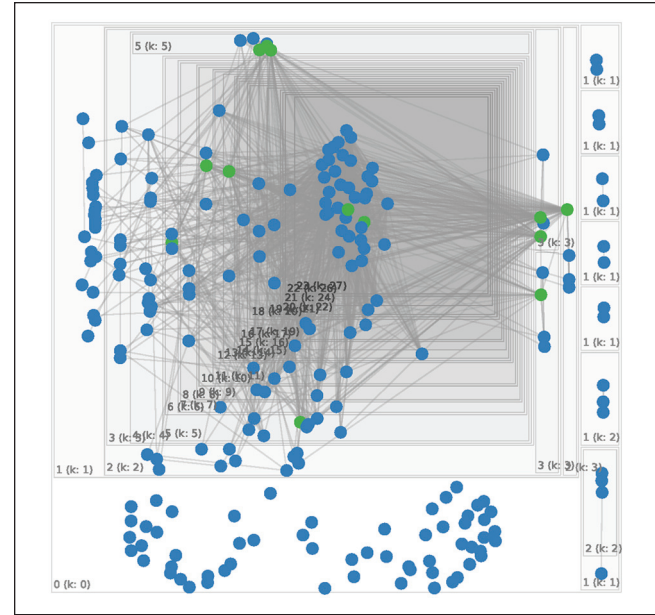


Figure 5. Cohesive blocks in the Hong Kong equity capital market financial network, 2013 to 2015.

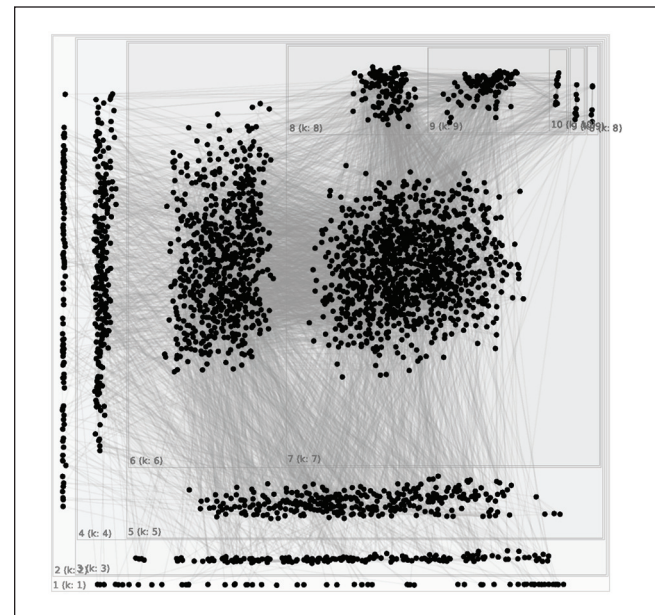


Figure 6. k -Core decomposition of Friendships in National Longitudinal Study of Adolescent to Adult Health data.

underlying Kamada-Kawai or Fruchterman-Reingold network layout algorithms struggle, however, to reveal fine structure at these higher scales. The figures does reveal a deeply nested hierarchy of high school friendships, showing a single, large, and close-knit friendship community at the 7-core, surrounded by a smaller number of increasingly disconnected students in the periphery. The hierarchy shows no branching until the deepest level, with two distinct but small groups at the 8-core

⁷We thank an anonymous reviewer for suggesting this comparison.

level, suggesting that most students participate in a common social hierarchy.

Discussion and Conclusion

The algorithm we describe in this article is inspired by a long tradition in social network analysis that focuses on visualization as a crucial tool in capturing social positions and social groups (Freeman 2005; Freeman and White 1993; Krempel 2009). It represents a novel approach to the visual representation of hierarchical networks—networks that contain a hierarchical structure. Our method emphasizes the dual aspect of such networks—the positions of different nodes and the relationships between groups containing the nodes in the network—and highlights how these two aspects interact. As such, the method we develop demonstrates how network structure manifests within hierarchical bounds and shows how a hierarchy constrains network interactions. Perhaps most important, our visualization approach makes clear how network ties cut across hierarchical boundaries, showing both how social relations bind social groups and how certain nodes act as bridges and articulation points, binding different parts of the hierarchy together.

An important application of our technique lies in the interpretation and exploration of results following from methods that identify hierarchies within networks. For instance, cohesive blocking (Moody and White 2003; White and Harary 2001) provides a powerful tool for analyzing social structures by focusing on social cohesion and embeddedness of groups. This method yields rich, multidimensional results that particularly benefit from visualization. However, the algorithm we develop is equally applicable to various methods that identify groups and subgroups in networks (Alba and Moore 1978; Alvarez-Hamelin et al. 2005; Girvan and Newman 2002; Richards and Rice 1981; Wasserman and Faust 1994:260–90). Although there are many techniques for the visualization of clusters and communities in networks, our approach offers a tool for researchers examining the nested structure of networks.

Our algorithm makes a number of principled decisions, some consequential, some less so. Our focus on embeddedness and bridge nodes underlies many of our design decisions about the layout algorithm. We use the squarified treemap algorithm to lay out elements of the hierarchy. Alternatives exist, including cushion (van Wijk and van de Wetering 1999) and Voronoi treemaps (Balzer and Deussen 2005). Although we use Kamada and Kawai's (1989) spring model to lay out the nodes within elements of the hierarchy, any network model that can be expressed in an energy formulation easily plugs into the existing algorithm. In our own experiments, energy formulations of Fruchterman and Reingold's (1991) and Davidson and Harel's (1996) force-directed layouts performed well. We note that various aspects of our algorithm are open to easy modification in future work. As such, we believe the visualization algorithm we

offer in this article provides a new tool to network researchers, a tool that is particularly tuned to the visual representation of hierarchical networks.

Acknowledgments

We wish to thank the editors of *Socius* and two anonymous reviewers for comments that helped transform the article.

Funding

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was supported by the National University of Singapore (NUS) Strategic Research Grant (WBS: R-109-000-183-646) awarded to the Global Production Networks Centre (GPN @ NUS) for the project Global Production Networks, Global Value Chains and East Asian Development.

References

- Abello, James. 2004. "Hierarchical Graph Maps." *Computers & Graphics* 28(3):345–59.
- Abello, James, Stephen G. Kobourov, and Roman Yusufov. 2004. "Visualizing Large Graphs with Compound-fisheye Views and Treemaps." Pp. 431–41 in *Graph Drawing: 12th International Symposium, GD 2004*, edited by János Pach. New York: Springer.
- Ahajjam, Sara, Mohamed El Haddad, and Hassan Badir. 2018. "A New Scalable Leader-community Detection Approach for Community Detection in Social Networks." *Social Networks* 54:41–49.
- Alba, Richard D., and Gwen Moore. 1978. "Elite Social Circles." *Sociological Methods & Research* 7(2):167–88.
- Alvarez-Hamelin, J. Ignacio, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani. 2005. "*k*-core Decomposition: A Tool for the Visualization of Large Scale Networks." Retrieved April 16, 2018 (<https://arxiv.org/abs/cs/0504107>).
- Armijo, Larry. 1966. "Minimization of Functions Having Lipschitz Continuous First Partial Derivatives." *Pacific Journal of Mathematics* 16(1):1–3.
- Arnheim, Rudolf. 1969. *Visual Thinking*. Berkeley: University of California Press.
- Balzer, M., and O. Deussen. 2005. "Voronoi Treemaps." Pp. 49–56 in *INFOVIS 2005*, edited by M. O. Ward and T. Munzner. New York: Institute of Electrical and Electronics Engineers.
- Bondy, J. A., and U. S. R. Murty. 1976. *Graph Theory with Applications*. New York: North-Holland.
- Brandes, Ulrik, Patrick Kenis, and Jörg Raab. 2006. "Explanation through Network Visualization." *Methodology* 2(1):16–23.
- Bruls, Mark, Kees Huizing, and Jarke J. Van Wijk. 2000. "Squarified Treemaps." Pp. 33–42 in *VisSym*, edited by Willem Cornelis de Leeuw and Robert van Liere. New York: Springer.
- Carmi, Shai, Shlomo Havlin, Scott Kirkpatrick, Yuval Shavitt, and Eran Shir. 2007. "A Model of Internet Topology Using *k*-shell Decomposition." *Proceedings of the National Academy of Sciences* 104(27): 11150–54.
- Chen, Chun-houh, Wolfgang Härdle, and Antony Unwin, eds. 2008. *Handbook of Data Visualization*. Berlin: Springer.

- Clauset, Aaron, Cristopher Moore, and Mark E. J. Newman. 2008. "Hierarchical Structure and the Prediction of Missing Links in Networks." *Nature* 453(7191):98–101.
- Correa, Carlos D., and Kwan-Liu Ma. 2011. "Visualizing Social Networks." Pp. 307–26 in *Social Network Data Analytics*, edited by Charu C. Aggarwal. Boston: Springer.
- Crnovrsanin, Tarik, Chris W. Muelder, Robert Faris, Diane Felmlee, and Kwan-Liu Ma. 2014. "Visualization Techniques for Categorical Analysis of Social Networks with Multiple Edge Sets." *Social Networks* 37:56–64.
- Crosby, Alfred W. 1997. *The Measure of Reality: Quantification in Western Europe, 1250–1600*. New York: Cambridge University Press.
- Davidson, Ron, and David Harel. 1996. "Drawing Graphs Nicely Using Simulated Annealing." *ACM Transactions on Graphics* 15(4):301–31.
- Diestel, Reinhard. 2005. *Graph Theory*. New York: Springer-Verlag.
- Di Giacomo, Emilio, and Anna Lubiw, eds. 2015. *Graph Drawing and Network Visualization, 23rd International Symposium, GD2015*. New York: Springer.
- Durkheim, Emile. [1893] 1964. *The Division of Labor in Society*. New York: Free Press.
- Eades, Peter, and Qing-Wen Feng. 1996. "Multilevel Visualization of Clustered Graphs." Pp. 101–12 in *Graph Drawing: Symposium on Graph Drawing, GD '96*, edited by Stephen C. North. New York: Springer.
- Emirbayer, Mustafa. 1997. "Manifesto for a Relational Sociology." *American Journal of Sociology* 103(2):281–317.
- Even, Shimon, and R. Endre Tarjan. 1975. "Network Flow and Testing Graph Connectivity." *SIAM Journal on Computing* 4(4):507–18.
- Fantasia, Rick. 1988. *Cultures of Solidarity: Consciousness, Action, and Contemporary American Workers*. Berkeley: University of California Press.
- Ford, L. R., and D. R. Fulkerson. 1962. *Flows in Networks*. Princeton, NJ: Princeton University Press.
- Freeman, Linton C. 2000. "Visualizing Social Networks." *Journal of Social Structure* 1(1):1–12.
- Freeman, Linton C. 2005. "Graphic Techniques for Exploring Social Network Data." Pp. 248–69 in *Models and Methods in Social Network Analysis*, edited by Peter J. Carrington, John Scott, and Stanley Wasserman. New York: Cambridge University Press.
- Freeman, Linton C., and Douglas R. White. 1993. "Using Galois Lattices to Represent Network Data." *Sociological Methodology* 23:127–46.
- Frishman, Y., and Ayellet Tal. 2004. "Dynamic Drawing of Clustered Graphs." Pp. 191–98 in *IEEE Symposium on Information Visualization*. New York: Institute of Electrical and Electronics Engineers.
- Fruchterman, Thomas M. J., and Edward M. Reingold. 1991. "Graph Drawing by Force-directed Placement." *Software: Practice and Experience* 21:1129–64.
- Galili, Tal. 2015. "Dendextend: An R Package for Visualizing, Adjusting and Comparing Trees of Hierarchical Clustering." *Bioinformatics* 31(22):3718–20.
- Gibbons, Alan. 1985. *Algorithmic Graph Theory*. New York: Cambridge University Press.
- Giddens, Anthony. 1979. *Central Problems in Social Theory: Action, Structure, and Contradiction in Social Analysis*. London: Macmillan.
- Girvan, Michelle, and Mark E. J. Newman. 2002. "Community Structure in Social and Biological Networks." *Proceedings of the National Academy of Sciences* 99(12):7821–26.
- Grabowski, A., and R. A. Kosiński. 2004. "Epidemic Spreading in a Hierarchical Social Network." *Physical Review E* 70(3):031908.
- Granovetter, Mark. 1985. "Economic Action and Social Structure: The Problem of Embeddedness." *American Journal of Sociology* 91(3):481–510.
- Harary, Frank. 1969. *Graph Theory*. Reading, MA: Addison-Wesley.
- Harel, David. 1988. "On Visual Formalisms." *Communications of the ACM* 31(5):514–30.
- Healy, Kieran, and James Moody. 2014. "Data Visualization in Sociology." *Annual Review of Sociology* 40(1):105–28.
- Hechter, Michael. 1987. *Principles of Group Solidarity*. Berkeley: University of California Press.
- Holten, Danny. 2006. "Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data." *IEEE Transactions on Visualization and Computer Graphics* 12(5):741–48.
- Jacomy, Mathieu, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. 2014. "ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software." *PLoS ONE* 9(6):1–12.
- Jensen, Paul A., and Mandell Bellmore. 1969. "An Algorithm to Determine the Reliability of a Complex System." *IEEE Transactions on Reliability* 18(4):169–74.
- Johnson, Brian, and Ben Shneiderman. 1991. "Tree-maps: A Space-filling Approach to the Visualization of Hierarchical Information Structures." Pp. 284–91 in *Proceedings of the 2nd Conference on Visualization 91*, edited by Gregory M. Nielson and Lary Rosenblum. New York: Institute of Electrical and Electronics Engineers.
- Kadushin, Charles. 1974. *The American Intellectual Elite*. Boston: Little, Brown.
- Kamada, Tomihisa, and Satoru Kawai. 1989. "An Algorithm for Drawing General Undirected Graphs." *Information Processing Letters* 31(1):7–15.
- Kanevsky, Arkady. 1993. "Finding All Minimum-size Separating Vertex Sets in a Graph." *Networks* 23(6):533–41.
- Krempel, Lothar. 2009. "Network Visualization." In *Handbook of Social Network Analysis*, edited by Peter J. Carrington and John Scott. Thousand Oaks, CA: Sage.
- Lee, Kam-Hon, and Ilan Vertinsky. 2011. "Strategic Adjustment of International Financial Centres (IFCs) in Small Economies: A Comparative Study of Hong Kong and Singapore." Pp. 239–58 in *The Pacific Rim: Investment, Development and Trade*, edited by Peter N. Nemetz. Vancouver, Canada: University of British Columbia Press.
- Lizardo, Omar. 2010. "Beyond the Antinomies of Structure: Levi-Strauss, Giddens, Bourdieu, and Sewell." *Theory and Society* 39(6):651–88.
- López, Luis, Jose F. F. Mendes, and Miguel A. F. Sanjuán. 2002. "Hierarchical Social Networks and Information Flow." *Physica A: Statistical Mechanics and Its Applications* 316(1):695–708.
- Mani, Dalhia, and James Moody. 2014. "Moving beyond Stylized Economic Network Models: The Hybrid World of the Indian

- Firm Ownership Network." *American Journal of Sociology* 119(6):1629–69.
- Martin, John Levi. 2009. *Social Structures*. Princeton, NJ: Princeton University Press.
- Meyer, Rachel, and Howard Kimeldorf. 2015. "Eventful Subjectivity: The Experiential Sources of Solidarity." *Journal of Historical Sociology* 28(4):429–57.
- Miorandi, D., and F. de Pellegrini. 2010. "k-shell Decomposition for Dynamic Complex Networks." Pp. 488–96 in *8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*. Piscataway, NJ: Institute of Electrical and Electronics Engineers.
- Moody, James. 2001. "Peer Influence Groups: Identifying Dense Clusters in Large Networks." *Social Networks* 23(4):261–83.
- Moody, James, Daniel McFarland, and Skye BenderdeMoll. 2005. "Dynamic Network Visualization." *American Journal of Sociology* 110(4):1206–41.
- Moody, James, and Douglas R. White. 2003. "Structural Cohesion and Embeddedness: A Hierarchical Concept of Social Groups." *American Sociological Review* 68(1):103–27.
- Moreno, Jacob Levy. 1932. *Application of the Group Method to Classification*. New York: National Committee on Prisons and Prison Labor.
- Moreno, Jacob Levy. 1934. *Who Shall Survive?* Washington, DC: Nervous and Mental Disease Publishing.
- Nagamochi, Hiroshi, and Toshihide Ibaraki. 2008. *Algorithmic Aspects of Graph Connectivity*. New York: Cambridge University Press.
- Newman, M.E.J. 2006. "Modularity and Community Structure in Networks." *Proceedings of the National Academy of Sciences* 103(23):8577–82.
- Newman, M.E.J., Albert-László Barabási, and Duncan J. Watts, eds. 2006. *The Structure and Dynamics of Networks*. Princeton, NJ: Princeton University Press.
- Nikolaev, Alexander G., Raihan Razib, and Ashwin Kucheriya. 2015. "On Efficient Use of Entropy Centrality for Social Network Analysis and Community Detection." *Social Networks* 40:154–62.
- Poon, Jessie P. H. 2003. "Hierarchical Tendencies of Capital Markets among International Financial Centers." *Growth and Change* 34(2):135–56.
- Provan, J. S., and D. R. Shier. 1996. "A Paradigm for Listing (s, t)-Cuts in Graphs." *Algorithmica* 15(4):351–72.
- Richards, William D., and Ronald E. Rice. 1981. "The NEGOPY Network Analysis Program." *Social Networks* 3(3):215–23.
- Shneiderman, Ben. 1992. "Tree Visualization with Tree-maps: 2-D Space-filling Approach." *ACM Transactions on Graphics* 11(1):92–99.
- Shneiderman, Ben, and Martin Wattenberg. 2001. "Ordered Treemap Layouts." Pp. 73–78 in *INFOVIS 2001*, edited by Keith Andrews, Steven Roth, and Chung Wong Pak. New York: Institute of Electrical and Electronics Engineers.
- Simmel, Georg. [1908] 1950. *The Sociology of Georg Simmel*. Glencoe, IL: Free Press.
- Simmel, Georg. [1922] 1964. "Web of Group Affiliations." Pp. 125–95 in *Conflict and the Web of Group-affiliations*, edited by Kurt H. Wolff and Reinhard Bendix. New York: Free Press.
- Storey, M.-A. D., and H. A. Muller. 1995. "Manipulating and Documenting Software Structures Using Shrimp Views." Pp. 275–84 in *ICSM '95*. Piscataway, NJ: Institute of Electrical and Electronics Engineers.
- Tamassia, Roberto, ed. 2013. *Handbook of Graph Drawing and Visualization*. Boca Raton, FL: CRC.
- Taylor, Alfred Maurice. 1966. *Imagination and the Growth of Science*. London: John Murray.
- Tönnies, Ferdinand. [1887] 1957. *Community and Society*. East Lansing: Michigan State University Press.
- Tufte, Edward R. 2001. *The Visual Display of Quantitative Information*, 2nd ed. Cheshire, CT: Graphics Press.
- Urbanek, Simon. 2008. "Visualizing Trees and Forests." Pp. 243–64 in *Handbook of Data Visualization*, edited by Chun-houh Chen, Wolfgang Härdle, and Antony Unwin. New York: Springer.
- van Wijk, Jarke J., and Huub van de Wetering. 1999. "Cushion Treemaps: Visualization of Hierarchical Information." Pp. 73–78 in *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis '99)*, edited by Steve Bryson and Theresa-Marie Rhyne. New York: Institute of Electrical and Electronics Engineers.
- Wasserman, Stanley, and Katherine Faust. 1994. *Social Network Analysis: Methods and Applications*. New York: Cambridge University Press.
- Weber, Max. 1978. *Economy and Society: An Outline of Interpretive Sociology*, Vol. I. Berkeley: University of California Press.
- White, Douglas R., and Frank Harary. 2001. "The Cohesiveness of Blocks in Social Networks: Node Connectivity and Conditional Density." *Sociological Methodology* 31(1):305–59.
- Whitney, Hassler. 1932a. "Congruent Graphs and the Connectivity of Graphs." *American Journal of Mathematics* 54(1):150–68.
- Whitney, Hassler. 1932b. "Non-separable and Planar Graphs." *Transactions of the American Mathematical Society* 17(2):339–62.
- Woo, J. J. 2016. *Singapore as an International Financial Centre: History, Policy and Politics*. London: Palgrave Macmillan.
- Wood, J., and J. Dykes. 2008. "Spatially Ordered Treemaps." *IEEE Transactions on Visualization and Computer Graphics* 14(6):1348–55.
- Zachary, Wayne W. 1977. "An Information Flow Model for Conflict and Fission in Small Groups." *Journal of Anthropological Research* 33(4):452–73.
- Zhao, Shengdong, M. J. McGuffin, and M. H. Chignell. 2005. "Elastic Hierarchies: Combining Treemaps and Node-link Diagrams." Pp. 57–64 in *INFOVIS*. Piscataway, NJ: Institute of Electrical and Electronics Engineers.
- Zukin, Sharon, and Paul DiMaggio. 1990. "Introduction." Pp. 1–36 in *Structures of Capital: The Social Organization of the Economy*, edited by Sharon Zukin and Paul DiMaggio. New York: Cambridge University Press.

Author Biographies

Kurtuluş Gemici is an assistant professor in the Department of Sociology and a member of the Global Production Networks Centre at the National University of Singapore. His research focuses on how emerging and developing countries respond to financial opening and market liberalization, the politics of international capital flows and mobility, and contentious politics against market reforms. Dr. Gemici holds a PhD in sociology from the

University of California, Los Angeles. His work has appeared in various venues, including journals such as *Socio-Economic Review*, *Theory & Society*, *Mobilization*, and *Politics & Society*. Recent publications include “Beyond the Minsky and Polanyi Moments: Social Origins of the Foreclosure Crisis” (*Politics & Society*, 2016) and “The Neoclassical Origins of Polanyi’s Self-regulating Market” (*Sociological Theory*, 2015).

Anthony Vashevko is a postdoctoral fellow with the Global Production Network Centre at the National University of Singapore. In 2017, he received his PhD in organizational behavior from the Stanford University Graduate School of Business, with a dissertation proposing formal models of innovative risk taking by organizations. His research focuses on organizational strategy, innovation, and market boundaries.