Social Networks 33 (2011) 12–19

Contents lists available at ScienceDirect

Social Networks

journal homepage: www.elsevier.com/locate/socnet

# An exact algorithm for a core/periphery bipartitioning problem

Michael Brusco*

*Department of Marketing, College of Business, Florida State University, Tallahassee, FL 32306-1110, USA*

ARTICLE INFO

*Keywords:*
Core/periphery
Bipartition
Exact algorithm
Branch-and-bound

ABSTRACT

The discrete optimization problem associated with partitioning a set of actors into core and periphery subsets has typically been approached using approximate procedures such as exchange heuristics, genetic algorithms, and simulated annealing. Although these procedures are effective and scalable for large networks, they do not guarantee an optimal bipartition. In this paper, an exact algorithm is presented for a core/periphery bipartitioning problem. Unlike the approximate procedures in the extant literature, this new algorithm, which is based on the principles of branch-and-bound programming, affords a guarantee of an optimal bipartition. Computational results for empirical and simulated data sets reveal that the proposed algorithm is extremely efficient for networks with 40 or fewer actors, and is often scalable for networks with up to 60 actors.

## 1. Introduction

The concept of a core/periphery structure has maintained a venerable status in social network analysis for several decades. Borgatti and Everett (1999) provided an excellent conceptual overview of core/periphery structure in networks and refinements are offered in a number of sources (Boyd et al., 2006, 2010; Everett and Borgatti, 1999, 2005; Garcia Muniz and Ramos Carvajal, 2006). In addition, Borgatti and Everett (1999) described permutation methods for testing for the presence of the core/periphery structure, as well as a modeling framework that attempts to uncover core/periphery structures in network matrices. Within this latter topic area, two types of models have been proposed: (a) a discrete model that treats the core and periphery as two mutually exclusive and exhaustive subsets of actors[1] (Borgatti and Everett, 1999; Boyd et al., 2006), and (b) a continuous model that posits a core-to-periphery continuum where each actor is measured with respect to a degree of "coreness" (Boyd et al., 2010). Throughout the remainder of this paper, we focus exclusively on the discrete model.

Borgatti and Everett (1999) offered a discrete formulation of core/periphery structure as a bipartitioning problem. Specifically, this model conceives that the actors of the network should be partitioned into two subsets: (a) a core of cohesive actors that share ties with fellow actors in the core, as well as actors in the periphery, and (b) a periphery subset of actors that are tied to some members of the core, but are not tied to other actors in the periphery. As an objective criterion for this bipartitioning model, Borgatti and Everett (1999) recommend maximizing the density within the core subset, while minimizing the density within the periphery subset. A genetic algorithm (Goldberg, 1989) was used to obtain bipartitions based on this criterion and this procedure has been incorporated in the popular UCINET software package (Borgatti et al., 2002). Boyd et al. (2006) offered their support for the objective of maximizing within-core density while minimizing within-periphery density and evaluated several heuristic procedures, including an exchange algorithm (Kernighan and Lin, 1970), simulated annealing algorithm (Aarts and Korst, 1989), and an alternative implementation of a genetic algorithm.

The approximate procedures for core/periphery bipartitioning problems developed by Borgatti and Everett (1999) and Boyd et al. (2006) are likely to perform effectively for many network matrices and are scalable for large networks. Nevertheless, none of these procedures afford a guarantee of a globally-optimal bipartition. It is the premise of our paper that guaranteed globally-optimal solutions can, in fact, frequently be obtained for networks with up to 60 (and possibly more) actors. More specifically, we propose a branch-and-bound programming method for a core/periphery bipartitioning problem. Branch-and-bound methods are exact solution procedures that are assured to provide globally-optimal solutions upon convergence, and they have proven effective for numerous partitioning problems (Brusco and Cradit, 2004; Brusco and Stahl, 2005b; Hansen and Delattre, 1978; Klein and Aronson, 1991; Koontz et al., 1975; Palubeckis, 1997). More recently, branch-and-bound methods have been recognized as a viable approach for blockmodeling of modestly-sized networks (Brusco et al., 2011; Brusco and Steinley, 2010). Although we in no way purport the branch-and-bound algorithm as a *replacement* for extant heuristic

* Tel.: +1 850 644 6512; fax: +1 850 644 4098.
*E-mail addresses:* mbrusco@fsu.edu, mbrusco@cob.fsu.edu.

[1] We use the term "actors" generically throughout the manuscript to refer to any set of objects with relational ties, which might be persons, firms, journals, etc. depending on the specific application.

0378-8733/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.socnet.2010.08.002

procedures for the core/periphery bipartitioning problem, we do believe this new algorithm offers a viable alternative for networks with up to about 60 actors.

In Section 2, we present a formal statement of the core/periphery bipartitioning problem studied in this paper. Section 3 provides a description of the branch-and-bound algorithm. The new algorithm is applied to a couple of empirical networks in Section 4. A simulation study is offered in Section 5 to evaluate the computational efficiency of the algorithm under various data conditions. A brief summary is provided in Section 6, along with limitations and extensions for future research.

## 2. Formulation of the core/periphery bipartitioning problem

Our algorithm for the core/periphery bipartitioning problem has the ability to accommodate either symmetric or asymmetric network matrices. We present the formulation assuming the more flexible case of asymmetry.

$n$ = the number of actors in the network;
$S$ = a set of $n$ indices that serve as identifiers for the $n$ actors, $S = \{1, 2, .., n\}$;
$\mathbf{A}$ = an $n \times n$ binary matrix, $\mathbf{A} = [a_{ij}]$, where $a_{ij} = 1$ if there is a tie from actor $i$ to actor $j$ and $a_{ij} = 0$ otherwise, for $1 \le i \neq j \le n$. As is common in most network applications (Wasserman and Faust, 1994), the main diagonal of $\mathbf{A}$ is ignored in the analyses;
$\Pi$ = the set of all $2^n - 2$ bipartitions of the $n$ actors into core and periphery subsets (the subtraction of 2 from the set of all bipartitions eliminates the possibility for empty core or periphery subsets);
$P$ = a bipartition of actors, $P = \{S_1, S_2\}$, where $S_1$ is the set of actors assigned to the core and $S_2$ is the set of actors assigned to the periphery.

The optimization problem for the core/periphery bipartitioning problem is:

$$\text{Minimize}: Z(P = \{S_1, S_2\}) = \left( \sum_{(i<j) \in S_1} (2 - a_{ij} - a_{ji}) \right) + \left( \sum_{(i<j) \in S_2} (a_{ij} + a_{ji}) \right), \qquad (1)$$

$$\text{Subject to}: \; P \in \Pi. \qquad (2)$$

The objective function (1) of the optimization problem represents the number of inconsistencies with perfect core/periphery structure. A perfect structure is realized when all members of the core exhibit ties (ties in both directions in the case of asymmetric network matrices) and there are no ties between members of the periphery. The first term in Eq. (1) collects inconsistencies that occur because of lack of ties between actors in the core. For example, if actors $i$ and $j$ are both members of the core (i.e., $S_1$), and $a_{ij} = a_{ji} = 1$, then the inconsistency penalty is $2 - 1 - 1 = 0$. However, if either $a_{ij} = 0$ or $a_{ji} = 0$ (or both equal zero), then an inconsistency penalty is incurred because members of the core should be tied. In a similar manner, the second term in Eq. (1) collects an inconsistency penalty if two actors in the periphery subset ($S_2$) exhibit ties in either (or both) directions. Constraint (2) ensures that $P$ is a permissible bipartition of the actors.

An exact solution to the core/periphery bipartitioning problem can be obtained via exhaustive enumeration over all bipartitions in $\Pi$. This approach is computationally infeasible as $n$ increases beyond 20. As noted by Boyd et al. (2006), at $n = 20$, there are more than 1 million bipartitions. For each increase in $n$ of 10 actors, the number of bipartitions increases by a factor of $2^{10} = 1024$. So, for $n = 30$, 40, 50, and 60 actors, there are roughly 1 billion, 1 trillion, 1 quadrillion, and 1 quintillion bipartitions in $\Pi$, respectively. To tackle problems of this size using an exact solution procedure, we employ an implicit enumeration approach based on the principles of branch-and-bound programming.

## 3. Branch-and-bound algorithm for the core/periphery bipartitioning problem

The branch-and-bound procedure for the core/periphery bipartitioning problem is similar in structure to those designed for within-cluster sum-of-squares partitioning (Koontz et al., 1975), within-cluster sum-of-cliques partitioning (Klein and Aronson, 1991), minimum-diameter partitioning (Brusco and Cradit, 2004), generalized structural balance partitioning (Brusco and Steinley, 2010), and relaxed structural balance partitioning (Brusco et al., 2011). However, there are a couple of salient differences. First, these previous applications permitted more than two clusters, whereas the current application is limited to bipartitioning. Second, the labeling of clusters in the previous applications was irrelevant, whereas the current application requires a single core cluster and a single periphery cluster.

The branch-and-bound algorithm considers each actor in the ordered list and makes assignments to either the core or periphery subset. A position pointer, $s$, marks the actor currently being considered for assignment. All actors with an index less than or equal to $s$ have received an assignment, and all actors with an index greater than $s$ have not been assigned to a subset. If $s < n$, then the current solution is a *partial solution* to the bipartitioning problem, whereas $s = n$ implies a *complete solution* because all actors have been assigned. For the case of a partial solution, the $s$ actors that have been assigned make a direct contribution to the objective function (1), which we denote as $Z_1$. The remaining $n - s$ actors have not been assigned and, therefore, their actual contribution to the objective function is unknown. However, it is possible to establish a lower bound on the contribution for each of the unassigned actors, and we denote the sum of these contributions as $Z_2$. Accordingly, a lower bound on the minimum possible objective function value for a complete bipartition resulting from the assignment of the remaining actors is $Z_1 + Z_2$. If this sum is equal to or greater than the objective function value for the current best-found bipartition, then there is no reason to attempt to pursue the current partial solution any further. In this case, we say that the partial solution is *pruned*, which means that any complete solution that could stem from that partial solution is not *explicitly* evaluated. The smaller $s$ is when a pruning operation occurs, the greater the computational efficiency of the algorithm because the fewer the compete solutions that will have to be generated. A precise description of the steps of our branch-and-bound algorithm are as follows:

Step 0. Set $s = 0$, $P^* = \varnothing$, $Z^* = \infty$, $Z_1 = 0$, $Z_2 = 0$, and $S_k = \varnothing$, for $1 \le k \le 2$.
Step 1. Set $s = s + 1$, $k = 1$, and $S_k = S_k \cup \{s\}$.
Step 2. Compute the following:

$$Z_1 = Z_1 + \sum_{j=1}^{s-1} \tau_j, \quad \text{where}$$

$$\tau_j = \begin{cases} (2 - a_{js} - a_{sj}) & \text{if } k = 1 \text{ and } j \in S_k \\ (a_{js} + a_{sj}) & \text{if } k = 2 \text{ and } j \in S_k \\ 0 & \text{otherwise.} \end{cases}$$

Step 3. Compute the following:

$$Z_2 = Z_2 + \sum_{j=s+1}^{n} \min(\eta_j, \omega_j), \quad \text{where}$$

$$\eta_j = \sum_{i=1}^{s} \alpha_{ji} \quad \text{for } s+1 \le j \le n,$$

$$\omega_j = \sum_{i=1}^{s} \beta_{ji} \quad \text{for } s+1 \le j \le n,$$

$$\alpha_{ji} = \begin{cases} (2 - a_{ji} - a_{ij}) & \text{if } i \in S_1 \\ 0 & \text{otherwise.} \end{cases}$$

$$\beta_{ji} = \begin{cases} a_{ji} + a_{ij} & \text{if } i \in S_2 \\ 0 & \text{otherwise.} \end{cases}$$

Step 4. If $(Z_1 + Z_2) \ge Z^*$, then go to Step 7.
Step 5. If $s < n$, then go to Step 1.
Step 6. Set $Z^* = Z_1$ and $P^* = P = \{S_1, S_2\}$.
Step 7. If $k = 2$, then go to Step 9.
Step 8. Perform the following sub-steps:

Step 8a. Set $Z_1 = Z_1 - \sum_{j=1}^{s-1} \tau_j$, where $\tau_j$ is as defined in Step 2.
Step 8b. Set $S_k = S_k - \{s\}$, $k = k + 1$, $S_k = S_k \cup \{s\}$.
Step 8c. Go to Step 2.

Step 9. Perform the following sub-steps:

Step 9a. $Z_1 = Z_1 - \sum_{j=1}^{s-1} \tau_j$, where $\tau_j$ is as defined in Step 2.
Step 9b. Set $S_k = S_k - \{s\}$.
Step 9c. Set $s = s - 1$.
Step 9d. If $s = 0$, then STOP; otherwise, set $k = l$: $s \in S_l$ and go to Step 7.

The parameters of the algorithm are initialized in Step 0 and the initial upper bound on the objective function is set to $\infty$. Step 1 advances the pointer, $s$, and assigns the actor to the core subset ($k = 1$). Step 2 computes the direct objective function contribution that arises from assigning actor $s$ to either the core or periphery subset and adds this value to $Z_1$. Step 3 computes $Z_2$, which is a lower bound on the objective function contribution that can be realized from the $n - s$ actors that have yet to be assigned to a subset. For each of these unassigned actors ($s + 1 \le j \le n$), we compute the contribution that would be realized if they were assigned to the core ($\eta_j$), as well as the contribution incurred if they were assigned to the periphery ($\omega_j$). Because the smaller of these two indices represents the best-case assignment for the actor, we add min $\{\eta_j, \omega_j\}$ to the bound component $Z_2$. The sum of $Z_1$ and $Z_2$ represents a lower bound on the best possible objective function value that can be realized after assigning all of the remaining actors. If this sum equals or exceeds the current upper bound in Step 4 (i.e., $Z_1 + Z_2 \ge Z^*$), then the partial solution is pruned and control is passed to Step 7.

If $s < n$ at Step 5, then the solution is not yet complete and control is passed to Step 1 for assignment of the next actor; otherwise, the complete solution is installed as the new best-found bipartition, $P^*$, in Step 6. Step 7 determines whether branching (Step 8) or depth retraction (Step 9) should occur. If actor $s$ is currently assigned to the core ($k = 1$), then the contribution to the core subset is removed in Step 8a, and the actor is moved to the periphery in Step 8b. Step 8c passes control to Step 2 to update the objective function contribution. Depth retraction occurs in Step 9 if actor $s$ is currently assigned to the periphery subset. The contribution to the periphery is removed in Step 9a and the assignment is removed in Step 9b. Step 9c reduces the position pointer by one (retraction) and, when this backtracking results in $s = 0$ at Step 9d, the algorithm terminates with $P^*$ as the optimal bipartition.

## 4. Application to empirical matrices

### 4.1. Co-citation network of social work journals

For our first application, we selected Baker's (1992) co-citation network data for 20 social work journals, which were obtained for the period from 1985 to 1986. These data have been analyzed by several authors in the social network literature (Borgatti and Everett, 1999; Boyd et al., 2006; Doreian et al., 2005). We focus on the symmetric dichotomization of the data used by Borgatti and Everett (1999). The branch-and-bound algorithm was applied to this binary network matrix and obtained the globally-optimal bipartition for the discrete core/periphery problem in less than 1/100th of 1 s. The optimal bipartition was identical to the solution reported by Borgatti and Everett (Table 6, p. 384), with a total of 10 inconsistencies (two 0s in the core and eight 1s in the periphery). The core consisted of 7 of the 20 (35%) of the journals {CW, CYSR, JSWE, SCW, SSR, SW, SWRA}.

### 4.2. Co-citation network of quantitative social sciences and statistical journals

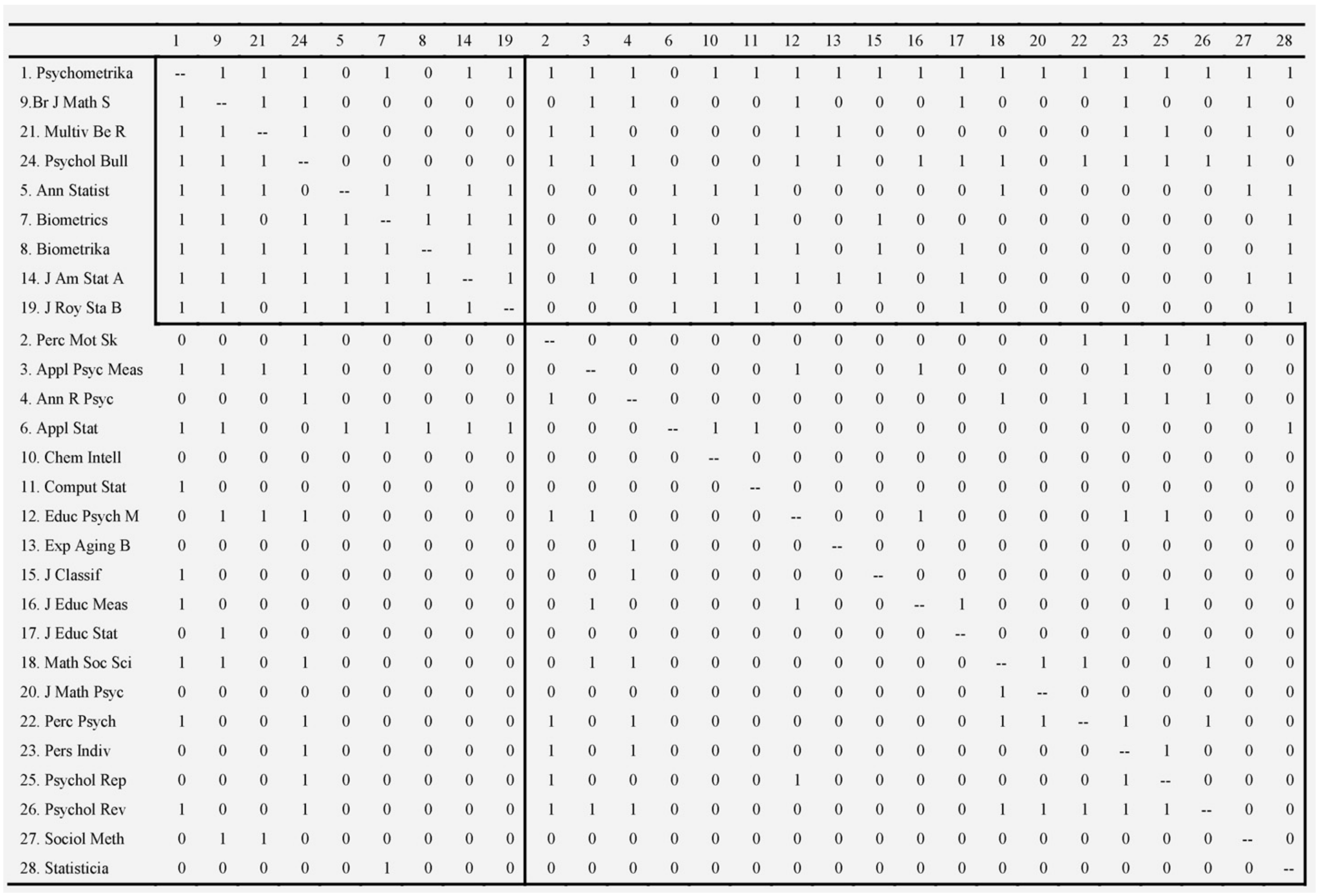
For our second application, we considered a co-citation matrix for 28 journals from the quantitative social sciences, which were collected for the time period of 1991–1992. Although less familiar to the social network literature, this co-citation data set has been examined using seriation and scaling procedures in the psychometric literature (Brusco and Stahl, 2001, 2005a; Groenen and Heiser, 1996). This particular co-citation matrix spans a somewhat broader range of journals than Baker's (1992) social work data set. For example, there are traditional quantitative psychology journals (e.g., *Psychometrika*, *Multivariate Behavioral Research*, *Journal of Mathematical Psychology*), leading journals in the general field of psychology (e.g., *Psychological Review*, *Psychological Bulletin*, *Annual Review of Psychology*), and leading journals in the field of statistics (*Annals of Statistics*, *Biometrics*, *Biometrika*, *Journal of the American Statistical Association*). We converted the co-citation data as published by Groenen and Heiser (1996, p. 547) to an asymmetric binary matrix by considering ties between journals only if the number of citations was 10 or greater. The resulting matrix is displayed in Table 1.

The branch-and-bound algorithm was applied to the network matrix in Table 1, yielding an optimal bipartition in 0.04 s. Table 2 presents a permuted matrix that illustrates the core and periphery structures. Not surprisingly, the core/periphery structure is not quite as strong for this network (there are 73 inconsistencies, as opposed to 10 for the social work data) in light of the greater heterogeneity in the set of journals. Nevertheless, it was encouraging

**Table 1**
A co-citation network among quantitative social sciences journals (rows are cited journals, columns are citing journals).

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. Psychometrika | – | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2. Perc Mot Sk | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 3. Appl Psyc Meas | 1 | 0 | – | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 4. Ann R Psych | 0 | 1 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 5. Ann Statist | 1 | 0 | 0 | 0 | – | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 6. Appl Stat | 1 | 0 | 0 | 0 | 1 | – | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 7. Biometrics | 1 | 0 | 0 | 0 | 1 | 1 | – | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 8. Biometrika | 1 | 0 | 0 | 0 | 1 | 1 | 1 | – | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 9.Br J Math S | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | – | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 10. Chem Intell | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11. Comput Stat | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12. Educ Psych M | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | – | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 13. Exp Aging B | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14. J Am Stat A | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | – | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 15. J Classif | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16. J Educ Meas | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | – | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 17. J Educ Stat | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18. Math Soc Sci | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 19. J Roy Sta B | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | – | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 20. J Math Psyc | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 21. Multiv Be R | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 22. Perc Psych | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | – | 1 | 1 | 0 | 1 | 0 | 0 |
| 23. Pers Indiv | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | 1 | 1 | 0 | 0 | 0 |
| 24. Psychol Bull | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | – | 1 | 1 | 1 | 0 |
| 25. Psychol Rep | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | – | 0 | 0 | 0 |
| 26. Psychol Rev | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | – | 0 | 0 |
| 27. Sociol Meth | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | – | 0 |
| 28. Statisticia | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – |

**Table 2**
Core/periphery permuted co-citation network among quantitative social sciences journals (rows are cited journals, columns are citing journals).



| | 1 | 9 | 21 | 24 | 5 | 7 | 8 | 14 | 19 | 2 | 3 | 4 | 6 | 10 | 11 | 12 | 13 | 15 | 16 | 17 | 18 | 20 | 22 | 23 | 25 | 26 | 27 | 28 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. Psychometrika | -- | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 9.Br J Math S | 1 | -- | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 21. Multiv Be R | 1 | 1 | -- | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 24. Psychol Bull | 1 | 1 | 1 | -- | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 5. Ann Statist | 1 | 1 | 1 | 0 | -- | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 7. Biometrics | 1 | 1 | 0 | 1 | 1 | -- | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8. Biometrika | 1 | 1 | 1 | 1 | 1 | 1 | -- | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 14. J Am Stat A | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -- | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 19. J Roy Sta B | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | -- | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2. Perc Mot Sk | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 3. Appl Psyc Meas | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 4. Ann R Psyc | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 6. Appl Stat | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | -- | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 10. Chem Intell | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11. Comput Stat | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12. Educ Psych M | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 13. Exp Aging B | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 15. J Classif | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16. J Educ Meas | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | -- | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 17. J Educ Stat | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18. Math Soc Sci | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 20. J Math Psyc | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | -- | 0 | 0 | 0 | 0 | 0 | 0 |
| 22. Perc Psych | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | -- | 1 | 0 | 1 | 0 | 0 |
| 23. Pers Indiv | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 1 | 0 | 0 | 0 |
| 25. Psychol Rep | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | -- | 0 | 0 | 0 |
| 26. Psychol Rev | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | -- | 0 | 0 |
| 27. Sociol Meth | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- | 0 |
| 28. Statisticia | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -- |

**Table 3**
Summary of simulation results – maximum and mean and computation times by feature level.

| Feature | Level | Maximum | Mean | Trimmed maximum[a] | Trimmed mean[a] |
|---|---|---|---|---|---|
| Number of actors | $n=20$ | 0.02 | 0.01 | 0.02 | 0.01 |
| | $n=30$ | 0.27 | 0.05 | 0.07 | 0.02 |
| | $n=40$ | 11.18 | 1.76 | 1.03 | 0.27 |
| | $n=50$ | 349.08 | 58.31 | 10.18 | 3.22 |
| | $n=60$ | 24,404.84 | 4108.31 | 267.38 | 63.23 |
| Percentage of actors in core | 20% | 24,404.84 | 1654.49 | 107.23 | 14.28 |
| | 40% | 267.38 | 12.89 | 267.38 | 12.89 |
| Density of core | 80% | 24,404.84 | 969.22 | 267.38 | 19.73 |
| | 90% | 17,445.31 | 698.16 | 74.03 | 6.97 |
| Density of periphery | 10% | 107.23 | 7.89 | 107.23 | 7.89 |
| | 20% | 24,404.84 | 1659.49 | 267.38 | 24.27 |
| Overall | | 24,404.84 | 833.69 | 267.38 | 13.35 |

[a] The trimmed mean is computed after removing the results for all test problems where the percentage of actors in the core is 20% and the density of the periphery is 20%.

to observe how rapidly the optimal solution was obtained for this larger, less well-structured network.

The optimal core subset consisted of 9 of the 28 (32%) journals {*Psychometrika, British Journal of Mathematical and Statistical Psychology, Multivariate Behavioral Research, Psychological Bulletin, Annals of Statistics, Biometrics, Biometrika, Journal of the American Statistical Association, Journal of the Royal Statistical Society B*}. This is an exceptionally solid core. The first four journals publish cutting-edge statistical research in the field of psychology, whereas the latter five are premiere statistical journals. Most of the inconsistencies in the core stem from the fact that the "statistical psychology" journals cite most of the statistics journals, but the reverse is not true.

## 5. A simulation study

Although the branch-and-bound algorithm was extremely effective and efficient for the modestly-sized empirical networks studied in the previous section, it is important to evaluate its performance across a broader set of data conditions. For this reason, we conducted a simulation study designed to measure the computation time requirements of the branch-and-bound algorithm under different scenarios. Four data features were manipulated in the experimental design. The first feature, the number of actors, was tested at five levels: $n=20$, 30, 40, 50, and 60. The second feature controlled the relative sizes of the core and periphery subsets, and was tested at two levels: (a) the core size $=.2n$ (i.e., 20% of the actors in the core) and (b) the core size $=.4n$. We selected these levels because the core is typically smaller than the periphery. For example, in the social work citation analyses (Borgatti and Everett, 1999; Boyd et al., 2006), the core size was roughly 30–35% (6 or 7 out of 20) of the total number of journals. The third and fourth features controlled the within-subset density of the core and periphery subsets, respectively. The levels of the third feature were 80% and 90%, whereas the levels of the fourth feature were 10% and 20%. The between-subset density was set at 50% for all test problems. Thus, density settings of 90% for the core and 10% for the periphery should allow for better fits because of greater distinction between the core and periphery. Contrastingly, density settings of 80% for the core and 20% for the periphery should result in poorer fits because the distinction is less pronounced.

A fully-crossed design was employed, resulting in $5 \times 2 \times 2 \times 2 = 40$ cells. Three problem replications were generated in each cell, which yielded a total of 120 test problems in the experiment. The branch-and-bound algorithm, which is freely available on request, was written in Fortran 90 and implemented on 2.4 GHz Core 2 Duo processor with 3GB of SDRAM. Each of the 120 test problems was submitted to the branch-and-bound algorithm and the objective function values and computation times were collected. The detailed results for each test problem are provided in the Appendix. A summary of the results is provided in Table 3.

Table 3 reports the maximum and mean computation time for each feature level of the experimental design. Across all 120 test problems, the maximum CPU time was 24,404.84 s and the mean was 833.69 s. It is important to recognize, however, that the mean computation time is grossly inflated because of only a few excessive CPU times. For example, the maximum computation time across all 72 test problems with 40 or fewer actors was only 11.18 s. The maximum computation time across the 24 problems with 50 actors was 349.08 s. Of the 24 test problems with 60 actors, 18 of them required 267.38 s or fewer; however, the remaining 6 problems each required between 8000 and 25,000 s of computation time, thus inflating the average.

A closer inspection of the data revealed that, in addition to the number of actors, the relative size of the core and the density of the periphery had a profound impact on computation time. The most challenging problems were those where the core size was 20% of the number of actors and the density of the periphery was 20%. In other words, network matrices with a large and fairly dense periphery subset presented a formidable challenge for the branch-and-bound algorithm as the number of actors increased beyond 40. To illustrate the effect of this condition, we also report "trimmed" maximum and mean values in Table 2, which reflect the omission of problems with a core size of 20% and a periphery density of 20%. We observe that trimming these test problems yields a substantial reduction in the maximum and mean computation times, particularly for the larger test problems. For example, the maximum time for 50-actor problems is reduced from 349.08 to 10.18 s. Similarly, the maximum time for the 60-actor networks is reduced from 24,404.84 to 267.38 s.

Overall, the simulation experiment was successful in identifying the strengths and shortcomings of the branch-and-bound algorithm. The algorithm was efficient for modestly-sized problems of up to 40 actors across all levels of the inherent core/periphery structure in the data. However, as the core/periphery structure weakened, computation time grew markedly and some 60-actor problems required massive computational effort. A weakening of structure with respect to larger periphery subsets of greater density appears to present the most formidable challenges for the algorithm.

## 6. Conclusions

We have presented an exact algorithm for a discrete core/periphery problem. The algorithm provided guaranteed optimal solutions for small empirical networks ($n=20$, $n=28$) in less than 1 s. In a simulation experiment, the new algorithm performed well for networks with up to 40 actors regardless of the data conditions. Although computation times for some of the 50- and

60-actor networks were excessive, these instances occurred only for those problems where the size of the core subset was only 20% of the number of actors and, simultaneously, the periphery subset was fairly dense (20% density). This data condition (a large, dense, periphery subset) corresponds to the weakest core/periphery condition in our experimental design. Accordingly, the stronger the differentiation between the core and periphery, the better the algorithm will perform.

In the sense that our branch-and-bound algorithm minimizes the number of 0 s in the core subset plus the number of 1 s in the periphery subset, it is consistent with Borgatti and Everett's (1999, p. 383) proposal "..to treat those off-diagonal regions of the matrix as missing data, so that the algorithm seeks only to maximize density in the core and minimize density in the periphery." This objective is also in accord with Boyd et al.'s (2006, p. 168) goal of finding a "..core/periphery bipartition that simultaneously maximizes connectivity in the core block and minimizes connectivity in the periphery block." Thus, we have implemented the algorithm to be concordant with core/periphery bipartitioning goals stated in the literature. Nevertheless, adaptation of the algorithm to allow for differential weighting of inconsistencies in core and periphery blocks is straightforward. It should also be possible to modify the algorithm to permit multiple core blocks (see, for example, Everett and Borgatti, 1999).

There are a number of possible enhancements to the branch-and-bound algorithm, which include but are not limited to the following: (a) using a heuristic to obtain a good initial upper bound, (b) reordering the actors in some manner to allow for faster pruning, and (c) employing a repetitive branch-and-bound approach. Moreover, there is a strong possibility that advanced mathematical programming approaches might enable much larger problem instances to be solved. Nevertheless, in its current form, the branch-and-bound algorithm is a straightforward standalone software program that can accommodate problems of moderate size.

As noted previously, the proposed branch-and-bound algorithm is not a replacement for extant heuristic methods for discrete core/periphery problems. The computational demands for some network matrices will be far too great for the algorithm. As our results indicate, a precise limit on problem size is impossible to specify. Optimal bipartitions for some 60-actor networks were obtained in less than 1 min, whereas other 60-actor networks required 5 or more hours. Thus, the efficiency of the algorithm is affected not only by size, but also by the characteristics of the data. Given that most discrete core/periphery analyses are apt to be performed on reasonably well-structured matrices, we feel that a 60-actor limit will be conservative in many instances. In some cases, optimal bipartitions for larger networks may be obtained in reasonable time.

## Appendix A. Detailed results of the simulation experiment

| | Feature level settings | | | | | |
|---|---|---|---|---|---|---|
| Cell replicate | Number of actors | % of actors in core | Core density | Periphery density | $Z^*$ | CPU time (in seconds) |
| 1 | 20 | 20% | 80% | 10% | 17 | 0.01 |
| 1 | 20 | 20% | 80% | 20% | 33 | 0.01 |
| 1 | 20 | 20% | 90% | 10% | 17 | 0.01 |
| 1 | 20 | 20% | 90% | 20% | 33 | 0.01 |
| 1 | 20 | 40% | 80% | 10% | 23 | 0.01 |
| 1 | 20 | 40% | 80% | 20% | 33 | 0.02 |
| 1 | 20 | 40% | 90% | 10% | 20 | 0.01 |
| 1 | 20 | 40% | 90% | 20% | 30 | 0.01 |
| 1 | 30 | 20% | 80% | 10% | 64 | 0.02 |
| 1 | 30 | 20% | 80% | 20% | 114 | 0.27 |
| 1 | 30 | 20% | 90% | 10% | 62 | 0.02 |
| 1 | 30 | 20% | 90% | 20% | 112 | 0.23 |
| 1 | 30 | 40% | 80% | 10% | 59 | 0.02 |
| 1 | 30 | 40% | 80% | 20% | 94 | 0.07 |
| 1 | 30 | 40% | 90% | 10% | 43 | 0.01 |
| 1 | 30 | 40% | 90% | 20% | 78 | 0.02 |
| 1 | 40 | 20% | 80% | 10% | 123 | 0.67 |
| 1 | 40 | 20% | 80% | 20% | 201 | 11.18 |
| 1 | 40 | 20% | 90% | 10% | 114 | 0.33 |
| 1 | 40 | 20% | 90% | 20% | 192 | 5.84 |
| 1 | 40 | 40% | 80% | 10% | 106 | 0.06 |
| 1 | 40 | 40% | 80% | 20% | 171 | 0.61 |
| 1 | 40 | 40% | 90% | 10% | 81 | 0.03 |
| 1 | 40 | 40% | 90% | 20% | 146 | 0.25 |
| 1 | 50 | 20% | 80% | 10% | 166 | 5.76 |
| 1 | 50 | 20% | 80% | 20% | 291 | 333.52 |
| 1 | 50 | 20% | 90% | 10% | 157 | 4.99 |
| 1 | 50 | 20% | 90% | 20% | 282 | 278.39 |
| 1 | 50 | 40% | 80% | 10% | 152 | 0.98 |
| 1 | 50 | 40% | 80% | 20% | 238 | 6.61 |
| 1 | 50 | 40% | 90% | 10% | 120 | 0.34 |
| 1 | 50 | 40% | 90% | 20% | 206 | 2.22 |
| 1 | 60 | 20% | 80% | 10% | 241 | 34.74 |
| 1 | 60 | 20% | 80% | 20% | 436 | 20,471.92 |
| 1 | 60 | 20% | 90% | 10% | 231 | 28.64 |
| 1 | 60 | 20% | 90% | 20% | 426 | 17,445.31 |
| 1 | 60 | 40% | 80% | 10% | 236 | 5.97 |
| 1 | 60 | 40% | 80% | 20% | 357 | 87.18 |
| 1 | 60 | 40% | 90% | 10% | 174 | 1.21 |
| 1 | 60 | 40% | 90% | 20% | 295 | 13.28 |
| 2 | 20 | 20% | 80% | 10% | 17 | 0.01 |
| 2 | 20 | 20% | 80% | 20% | 37 | 0.01 |
| 2 | 20 | 20% | 90% | 10% | 15 | 0.01 |
| 2 | 20 | 20% | 90% | 20% | 35 | 0.01 |

| | Feature level settings | | | | | |
|---|---|---|---|---|---|---|
| Cell replicate | Number of actors | % of actors in core | Core density | Periphery density | $Z^*$ | CPU time (in seconds) |
| 2 | 20 | 40% | 80% | 10% | 17 | 0.01 |
| 2 | 20 | 40% | 80% | 20% | 33 | 0.01 |
| 2 | 20 | 40% | 90% | 10% | 14 | 0.01 |
| 2 | 20 | 40% | 90% | 20% | 30 | 0.01 |
| 2 | 30 | 20% | 80% | 10% | 63 | 0.02 |
| 2 | 30 | 20% | 80% | 20% | 107 | 0.13 |
| 2 | 30 | 20% | 90% | 10% | 61 | 0.01 |
| 2 | 30 | 20% | 90% | 20% | 105 | 0.11 |
| 2 | 30 | 40% | 80% | 10% | 58 | 0.01 |
| 2 | 30 | 40% | 80% | 20% | 87 | 0.03 |
| 2 | 30 | 40% | 90% | 10% | 46 | 0.01 |
| 2 | 30 | 40% | 90% | 20% | 75 | 0.01 |
| 2 | 40 | 20% | 80% | 10% | 102 | 0.33 |
| 2 | 40 | 20% | 80% | 20% | 177 | 7.78 |
| 2 | 40 | 20% | 90% | 10% | 96 | 0.28 |
| 2 | 40 | 20% | 90% | 20% | 171 | 6.21 |
| 2 | 40 | 40% | 80% | 10% | 85 | 0.06 |
| 2 | 40 | 40% | 80% | 20% | 138 | 0.27 |
| 2 | 40 | 40% | 90% | 10% | 67 | 0.03 |
| 2 | 40 | 40% | 90% | 20% | 120 | 0.12 |
| 2 | 50 | 20% | 80% | 10% | 171 | 4.99 |
| 2 | 50 | 20% | 80% | 20% | 308 | 349.08 |
| 2 | 50 | 20% | 90% | 10% | 159 | 3.3 |
| 2 | 50 | 20% | 90% | 20% | 296 | 236.28 |
| 2 | 50 | 40% | 80% | 10% | 158 | 1.02 |
| 2 | 50 | 40% | 80% | 20% | 249 | 8.11 |
| 2 | 50 | 40% | 90% | 10% | 121 | 0.31 |
| 2 | 50 | 40% | 90% | 20% | 212 | 2.03 |
| 2 | 60 | 20% | 80% | 10% | 247 | 101.66 |
| 2 | 60 | 20% | 80% | 20% | 447 | 24,404.84 |
| 2 | 60 | 20% | 90% | 10% | 230 | 61.17 |
| 2 | 60 | 20% | 90% | 20% | 430 | 15,355.67 |
| 2 | 60 | 40% | 80% | 10% | 227 | 12.36 |
| 2 | 60 | 40% | 80% | 20% | 364 | 267.38 |
| 2 | 60 | 40% | 90% | 10% | 179 | 3.64 |
| 2 | 60 | 40% | 90% | 20% | 316 | 74.03 |
| 3 | 20 | 20% | 80% | 10% | 20 | 0.01 |
| 3 | 20 | 20% | 80% | 20% | 33 | 0.01 |
| 3 | 20 | 20% | 90% | 10% | 20 | 0.01 |
| 3 | 20 | 20% | 90% | 20% | 33 | 0.01 |
| 3 | 20 | 40% | 80% | 10% | 19 | 0.01 |
| 3 | 20 | 40% | 80% | 20% | 27 | 0.01 |
| 3 | 20 | 40% | 90% | 10% | 15 | 0.01 |
| 3 | 20 | 40% | 90% | 20% | 23 | 0.01 |
| 3 | 30 | 20% | 80% | 10% | 55 | 0.02 |
| 3 | 30 | 20% | 80% | 20% | 86 | 0.06 |
| 3 | 30 | 20% | 90% | 10% | 52 | 0.01 |
| 3 | 30 | 20% | 90% | 20% | 83 | 0.05 |
| 3 | 30 | 40% | 80% | 10% | 56 | 0.01 |
| 3 | 30 | 40% | 80% | 20% | 82 | 0.03 |
| 3 | 30 | 40% | 90% | 10% | 39 | 0.01 |
| 3 | 30 | 40% | 90% | 20% | 65 | 0.01 |
| 3 | 40 | 20% | 80% | 10% | 98 | 0.19 |
| 3 | 40 | 20% | 80% | 20% | 174 | 3.91 |
| 3 | 40 | 20% | 90% | 10% | 87 | 0.13 |
| 3 | 40 | 20% | 90% | 20% | 163 | 2.52 |
| 3 | 40 | 40% | 80% | 10% | 107 | 0.19 |
| 3 | 40 | 40% | 80% | 20% | 171 | 1.03 |
| 3 | 40 | 40% | 90% | 10% | 76 | 0.05 |
| 3 | 40 | 40% | 90% | 20% | 140 | 0.27 |
| 3 | 50 | 20% | 80% | 10% | 147 | 2.00 |
| 3 | 50 | 20% | 80% | 20% | 277 | 81.39 |
| 3 | 50 | 20% | 90% | 10% | 137 | 1.66 |
| 3 | 50 | 20% | 90% | 20% | 267 | 62.8 |
| 3 | 50 | 40% | 80% | 10% | 165 | 0.74 |
| 3 | 50 | 40% | 80% | 20% | 249 | 10.18 |
| 3 | 50 | 40% | 90% | 10% | 121 | 0.19 |
| 3 | 50 | 40% | 90% | 20% | 205 | 2.53 |
| 3 | 60 | 20% | 80% | 10% | 248 | 107.23 |
| 3 | 60 | 20% | 80% | 20% | 425 | 11,601.22 |
| 3 | 60 | 20% | 90% | 10% | 233 | 70.14 |
| 3 | 60 | 20% | 90% | 20% | 410 | 8182.45 |
| 3 | 60 | 40% | 80% | 10% | 251 | 14.94 |
| 3 | 60 | 40% | 80% | 20% | 374 | 212.25 |
| 3 | 60 | 40% | 90% | 10% | 188 | 2.74 |
| 3 | 60 | 40% | 90% | 20% | 311 | 39.52 |

## References

Aarts, E., Korst, J., 1989. Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing. Wiley, New York.

Baker, D.R., 1992. A structural analysis of the social work journal network: 1985–1986. Journal of Social Service Research 15, 153–168.

Borgatti, S.P., Everett, M.G., 1999. Models of core/periphery structures. Social Networks 21, 375–395.

Borgatti, S.P., Everett, M.G., Freeman, L.C., 2002. Ucinet for Windows, Version 6.153: Software for Social Network Analysis. Analytic Technologies, Harvard, MA.

Boyd, J.P., Fitzgerald, W.J., Beck, R.J., 2006. Computing core/periphery structures and permutation tests for social relations data. Social Networks 28 (2), 165–178.

Boyd, J.P., Fitzgerald, W.J., Mahutga, M.C., Smith, D.A., 2010. Computing continuous core/periphery structures for social relations data with MINRES/SVD. Social Networks 32, 125–137.

Brusco, M.J., Cradit, J.D., 2004. Graph coloring, minimum-diameter partitioning, and the analysis of confusion matrices. Journal of Mathematical Psychology 48 (5), 301–309.

Brusco, M., Doreian, P., Mrvar, A., Steinley, D., 2011. Linking theory, models, and data to understand social network phenomena: two algorithms for relaxed structural balance partitioning. Sociological Methods & Research 40 (1).

Brusco, M.J., Stahl, S., 2001. An interactive approach to multiobjective combinatorial data analysis. Psychometrika 66 (1), 5–24.

Brusco, M.J., Stahl, S., 2005a. Optimal least-squares unidimensional scaling: improved branch-and-bound procedures and comparison to dynamic programming. Psychometrika 70 (2), 253–270.

Brusco, M.J., Stahl, S., 2005b. Branch-and-Bound Applications in Combinatorial Data Analysis. Springer, New York.

Brusco, M.J., Steinley, D., 2010. K-balance partitioning: An exact method with application to generalized structural balance and other psychological contexts. Psychological Methods 15 (2), 145–157.

Doreian, P., Batagelj, V., Ferligoj, A., 2005. Generalized Blockmodeling. Cambridge University Press, Cambridge, UK.

Everett, M.G., Borgatti, S.P., 1999. Peripheries of cohesive subsets. Social Networks 21 (4), 397–407.

Everett, M.G., Borgatti, S.P., 2005. Extending centrality. In: Carrington, P.J., Scott, J., Wasserman, S. (Eds.), Models and Methods in Social Network Analysis. Cambridge University Press, New York, pp. 57–76.

Garcia Muniz, A.S., Ramos Carvajal, C., 2006. Core/periphery structure models: an alternative proposal. Social Networks 28 (4), 442–448.

Goldberg, D.E., 1989. Genetic Algorithms. Addison-Wesley, New York.

Groenen, P.J.F., Heiser, W.J., 1996. The tunneling method for global optimization in multidimensional scaling. Psychometrika 61, 529–550.

Hansen, P., Delattre, M., 1978. Complete-link cluster analysis by graph coloring. Journal of the American Statistical Association 73, 397–403.

Kernighan, B.W., Lin, S., 1970. An efficient heuristic procedure for partitioning graphs. Bell Systems Technical Journal 49, 221–226.

Klein, G., Aronson, J.E., 1991. Optimal clustering: a model and method. Naval Research Logistics 38, 447–461.

Koontz, W.L.G., Narendra, P.M., Fukunaga, K., 1975. A branch and bound clustering algorithm. IEEE Transactions on Computing C-24, 908–915.

Palubeckis, G., 1997. A branch-and-bound approach using polyhedral results for a clustering problem. INFORMS Journal on Computing 9 (1), 30–42.

Wasserman, S., Faust, K., 1994. Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge.