Finding globally optimal macrostructure in multiple relation, mixed-mode social networks

Matthew F Dabkowski1, Neng Fan2 and Ronald Breiger3

Abstract
From the outset, computational sociologists have stressed leveraging multiple relations when blockmodeling social networks. Despite this emphasis, the majority of published research over the past 40 years has focused on solving blockmodels for a single relation. When multiple relations exist, a reductionist approach is often employed, where the relations are stacked or aggregated into a single matrix, allowing the researcher to apply single relation, often heuristic, blockmodeling techniques. Accordingly, in this article, we develop an exact procedure for the exploratory blockmodeling of multiple relation, mixed-mode networks. In particular, given (a) \( N_1 \) actors, (b) \( N_2 \) events, (c) an \( (N_1 \times N_1) \) binary one-mode network depicting the ties between actors, and (d) an \( (N_1 \times N_2) \) binary two-mode network representing the ties between actors and events, we use integer programming to find globally optimal \( (P_1 \times P_1 \times P_1 \times P_2) \) image matrices and partitions, where \( P_1 \) and \( P_2 \) represent the number of actor and event positions, respectively. Given the problem’s computational complexity, we also develop an algorithm to generate a minimal set of non-isomorphic image matrices, as well as a complementary, easily accessible heuristic using the network analysis software Pajek. We illustrate these concepts using a simple, hypothetical example, and we apply our techniques to a terrorist network.

Keywords
Exploratory blockmodeling, integer programming, mixed-mode network, social position, isomorphism, structural equivalence

Introduction and background
Loosely defined as the partitioning of a social network’s \( N \) actors into \( P \) non-overlapping subsets (or positions, \( P < N \) ) which are linked by a pattern of ties, blockmodels use the individual, visible connections between actors to reveal a network’s collective, often hidden macrostructure. For example, consider the binary sociomatrix on the left-side of Figure 1.

Consisting of 20 actors labeled \( A \) through \( T \), imagine the presence of a 1 in row \( i \) and column \( j \) (e.g. \( s_{A,B} = 1 \) ) implies actor \( i \) confides in actor \( j \). In its current form, the sociomatrix provides little insight into the actors’ collective trust network. However, as seen on the right-side of Figure 1, a clever permutation of the actors reveals that the trust network has a three-tier hierarchical structure, where the actors inside of each tier (or position, delineated by blue lines) have identical patterns of ties. Abstractly, this macrostructure may be further distilled into the image matrix seen in Figure 2, which shows the ties between the positions (labeled B1 through B3).

Collectively, the partitioning of the actors (right-side of Figure 1) and its associated image matrix (Figure 2) constitute a blockmodel for the trust network, and, in this case, the fit is perfect. That said, reality is rarely this kind, as inconsistencies between the ties of the positions and the partitioned network...
Methodological Innovations

are the norm. Moreover, since its earliest formulation in the mid-1970s, blockmodeling has emphasized the necessity of incorporating multiple relations (Boorman and White, 1976; White et al., 1976). As White et al. (1976: 739) argue in their seminal paper, “many different types of tie are needed to portray the social structure of a population.” However, despite this foundational emphasis and several notable exceptions (e.g. Baker, 1986; Batagelj et al., 2007; Borgatti and Everett, 1992; Brusco et al., 2013b; Žiberna, 2014), the vast majority of published research over the past 40 years has focused on solving blockmodels for a single relation.

When multiple relations exist, a reductionist approach is often employed, where the relations are either stacked or aggregated into a single matrix, allowing the researcher to apply single relation blockmodeling techniques. Nonetheless, this simplification can mask structural nuances within the individual relations, and as Doreian (2006: 128) rightly puts it, “we need an approach that permits different relations to have different structures.” In short, blockmodeling multiple relations remains an open, relevant problem.

Accordingly, in this paper we extend Brusco and Steinley’s (2009) exact procedure to the exploratory blockmodeling of multiple relation, mixed-mode networks. In particular, given (a) \(N_1\) actors, (b) \(N_2\) events, (c) an \(N_1 \times N_1\) binary one-mode network depicting the ties between actors, and (d) an \(N_1 \times N_2\) binary two-mode network representing the ties between actors and events, we use integer programming and structural equivalence to simultaneously find the globally optimal \(P_1\) and \(P_2\) image matrices and partitions, where \(P_1\) and \(P_2\) represent the number of actor and event positions, respectively. Given the computational complexity of this problem, we also develop an algorithm to generate a minimal set of non-isomorphic image matrices of size \((|P_1| \times |P_2|)\), greatly reducing the total number of image matrices that must be fit when compared to the set of all possible \((|P_1| \times |P_1|)\) image matrices. In addition, we provide a here.

To facilitate our explanation and provide researchers with a flexible set of tools, we derive our multiple relation, mixed-mode approach (section “Exact exploratory blockmodeling for multiple relation, mixed-mode networks using integer programming and structural equivalence”) from its initial application to single one-mode (Dabkowski et al., 2016; section “Exact exploratory blockmodeling for one-mode networks using integer programming and structural equivalence”) and two-mode networks (section “Exact exploratory blockmodeling for two-mode networks using integer programming and structural equivalence”).
set to 0. Stacking \( S^{(1)} \) and \( S^{(0)} \) produces the single matrix seen in Panel (a) of Figure 4, and subsequently partitioning the actors into \( P = 2 \) positions using CONCOR yields the permuted adjacency matrices seen in Panel (b). Finally, to transform our results into image matrices, we apply the well-known \( \alpha \) density criterion, where a given block in relation \( k \) is set to 1 if the tie density in the block exceeds the overall tie density for relation \( k \) (0.40 for \( S^{(1)} \) and 0.70 for \( S^{(0)} \)); otherwise, the block is set to 0 (Wasserman and Faust, 1994: 400). The block densities and corresponding 2-position image matrices are given in Panels (c) and (d) of Figure 4, respectively.

As Panel (d) of Figure 4 shows, the image matrices derived from CONCOR reflect an insular, compartmentalized social structure where actors in position \( P \) are generally linked with each other but are not tied to actors in position \( l \) (\( p, l = \{1, 2\}; p \neq l \)). Based on Panel (b) of Figure 4, this seems reasonable. That said, as Wasserman and Faust (1994: 380–381) note “the exact mathematical properties of CONCOR remain unclear . . . [and] it is not clear what, if anything, it is optimizing.” Moreover, although our primary concern rests on the actors, this does not imply that we are uninterested in the events.

To address both of these shortcomings, Doreian et al. (2005) developed an alternative, direct approach, which is based on minimizing the number of inconsistencies between a permuted, partitioned adjacency (or affiliation) matrix and its corresponding image matrix. Specifically, if actors \( i \) and \( j \) are assigned to positions \( P \) and \( l \), respectively; an inconsistency exists when the tie from actor \( i \) to actor \( j \) does not match the tie from position \( P \) to position \( l \). For example, consider position B1 in Panels (b) and (d) of Figure 4. In the top-half of Panel (d) we note that position B1 is tied to itself, while in Panel (b) there are several off-diagonal 0’s in the corresponding block (\( s_{44,41} = 0 \) and \( s_{45,44} = 0 \)). These are inconsistencies, and they would negatively, collectively impact the quality-of-fit by two.

Using this construct, Doreian et al. (2005: 150) employ a simple actor (or event) relocation routine to minimize the number of inconsistencies, and this is implemented for a single relation in Pajek (Mrvar and Batagelj, 2013). Dabkowski et al. (2015) further extend Pajek’s implementation of Doreian et al.’s (2005) direct approach to multiple one-mode networks using an augmented adjacency matrix and appropriate penalties and constraints, and they note that their extension can be modified to accommodate two-mode networks.

As such, it seems reasonable to assume that we can adapt Dabkowski et al.’s (2015) approach for our hypothetical multiple relation, mixed-mode example, and, in fact, it is quite simple and similar to Žiberna’s (2014) true multilevel approach. To illustrate this, we first note that Pajek treats one-mode and two-mode networks differently with respect to inconsistencies. Specifically, in one-mode networks Pajek assumes that self-ties are irrelevant, and the main
diagonal of the adjacency matrix is ignored. For our actor-by-actor network in Panel (a) of Figure 3, this is the desired behavior. That said, for our actor-by-event network in Panel (b) of Figure 3, it is a different story, as there are no self-ties in an $N_1 \times N_2$ affiliation matrix and every element must be assessed. Accordingly, as seen in Panel (a) of Figure 5, we need to construct our augmented adjacency matrix such that the one-mode network lies along the main diagonal and the two-mode network does not. Furthermore, in order to be treated as a one-mode network in Pajek and accrue inconsistencies correctly, our augmented adjacency matrix must be square, and we pad it with an appropriate number of zero rows.

Our next task is to feed Pajek constraints that force it to maintain the assignment of individual actors and events within their respective actor and event positions. Assuming
we want to split both the actors and events into two positions, this implies there are four total positions, and we designate the first two positions as actor positions and the last two positions as event positions. As such, we need constraints that prohibit actors (A1 through A5 in Panel (a) of Figure 5) from being assigned to the last two positions, as well as constraints that prohibit events (E1 through E7 in Panel (a) of Figure 5) from being assigned to the first two positions.

With respect to penalties, unlike Dabkowski et al.’s (2015) onerous requirement to maintain the assignment of actors to identical positions across multiple one-mode networks, we have a single one-mode network of actors augmented by a second two-mode network of events. This greatly simplifies our model, as it eliminates the need for Dabkowski et al.’s (2015) variable penalties, and we can view each inconsistency equivalently across the blockmodel, allowing us to use the simple penalty matrix of 1’s seen in Panel (b) of Figure 5. Finally, we need to specify the permissible block types in the image matrix. Captured in Panel (c) of Figure 5, we allow Pajek to consider both 0 and 1 blocks in the substantive cells (the top-half of Panel (c)), while we set the lower blocks to 0 to match the zero-row padding of the augmented adjacency matrix.

Armed with these constraints, penalties, and permissible block types, we loaded the model into Pajek (see Appendix I in the online supplemental material for the .net and.mdl files), and Figure 6 displays the optimal solution returned after 200 repetitions. As indicated by the red cells and font in Panel (a) of Figure 6, there are 13 inconsistencies between the permuted, partitioned augmented adjacency matrix and its corresponding image matrix in Panel (b). Moreover, the actor partitioning and its associated image matrix match the results returned by CONCOR in the top-halves of Panels (b) and (d) of Figure 4, lending some support to CONCOR’s indirect approach. Nonetheless, while Pajek’s output looks convincing, its relocation routine generates locally optimal results, and we have no assurance that better solutions do not exist. Of course, our angst will only grow with larger, messier real-world examples.

**Exact exploratory blockmodeling for one-mode networks using integer programming and structural equivalence**

In order to alleviate concerns related to the potential suboptimality of blockmodeling heuristics, one can employ exact procedures that guarantee globally optimal solutions. An excellent example of this approach is Brusco and Steinley’s (2009) use of integer programming to find optimal partitions for one- and two-mode blockmodels when the image matrix is prespecified. Known as confirmatory blockmodeling, they apply their method to several well-known, previously studied social networks, confirming the global optimality of heuristically obtained results. When the image matrix is not known in advance (i.e. exploratory blockmodeling), Brusco and Steinley (2009) note that their approach can be applied to all possible image matrices of a given size, provided the number of positions is small.

In a recent paper, Dabkowski et al. (2016) extend the applicability of Brusco and Steinley’s (2009) work to the exploratory blockmodeling of one-mode networks by demonstrating how a much smaller collection of image matrices can cover all possible image matrices of a given size. Specifically, Dabkowski et al. (2016) develop an algorithm to generate a set of non-isomorphic, binary image matrices, and they further reduce the size of this set by identifying and removing image matrices with structurally equivalent positions. In short, the computational savings are impressive. For example, while there are \( \sum_{r=1}^{4} 2^{r} = 66,066 \) possible one-mode, binary image matrices with 4 or fewer positions, only 2876 of these must be fit (Dabkowski et al., 2016). Exploiting this efficiency, Dabkowski et al. (2016) embed Brusco and Steinley’s (2009) integer program inside a loop that cycles through each image matrix of this minimal, non-isomorphic set. At termination, the image matrix(matrices) with the smallest number of inconsistencies is(are) the global optimum(optima).

Although the details of their work are beyond the scope of this paper, Brusco and Steinley’s (2009) integer program, as modified by Dabkowski et al. (2016), plays a central role in
the development of our exact exploratory blockmodeling for multiple relation, mixed-mode networks. As such, we provide Dabkowski et al.’s (2016) formulation below, along with additional discussion and expanded notation to denote one-mode networks.

One-mode optimization model (Dabkowski et al., 2016)

Index and set use: Let (a) \( N = \{1, \ldots, N\} \) denote a set of actors where \( i, j \in N \); (b) \( P = \{1, \ldots, P\} \) denote a set of social positions where \( p, l \in P \); and (c) \( R = \{1, \ldots, R\} \) denote the index set of minimal, non-isomorphic \( (P \times P) \) one-mode binary image matrices \( B_r^{(r)} \) where \( r \in R \).

Data: The primary input is an \((N \times N)\) binary adjacency matrix (sociomatrix) \( S^{(1)} \), where the rows and columns of \( S^{(1)} \) represent the set of actors; a value of 1 in row \( i \) and column \( j \) \( (s^{(1)}_{ij} = 1) \) indicates actor \( i \) is “tied” to actor \( j \); and a value of 0 implies otherwise. In addition, we have \( R \) ideal image matrices \( B_r^{(1)} \), where the rows and columns of \( B_r^{(1)} \) represent the set of positions; a value of 1 in row \( p \) and column \( l \) \( (b_{pl} = 1) \) indicates position \( P \) is “tied” to position \( l \) in image matrix \( r \), and a value of 0 implies otherwise.

Decision variables: Our decision variables are \( x_{i,p}^{(r)} \in \{0, 1\} \), where \( x_{i,p}^{(r)} = 1 \) indicates that actor \( i \) is assigned to position \( p \) when using image matrix \( r \), and \( x_{i,p}^{(r)} = 0 \) otherwise.

Formulation: For \( r \in R \), solve

\[
\begin{align*}
\min & \sum_{p,l=1}^{P} \sum_{i,j=1}^{N} x_{i,p}^{(r)} x_{j,l}^{(r)} \left( b_{pl}^{(r)} + s_{ij}^{(1)} - 2 b_{pl}^{(r)} s_{ij}^{(1)} \right) \\
\text{s.t.} & \sum_{p=1}^{P} x_{i,p}^{(r)} = 1, \quad \forall i \in N \quad (1a) \\
& \sum_{i=1}^{N} x_{i,p}^{(r)} \geq 1, \quad \forall p \in P \quad (1b) \\
& x_{i,p}^{(r)} \in \{0, 1\} \quad \forall i \in N, p \in P \quad (1d)
\end{align*}
\]

Discussion: The objective function, (1a), counts the number of inconsistencies between the image matrix \( r \) and the assignment of actors \( i \) and \( j \) to positions \( p \) and \( l \). To see this, consider the coefficient \( \left( x_{i,p}^{(r)} + s_{ij}^{(1)} - 2 b_{pl}^{(r)} s_{ij}^{(1)} \right) \) and assume that the tie from actor \( i \) to position \( P \) matches the tie from position \( P \) to position \( l \). In this case, there is no inconsistency, and there are two ways this can occur: (a) \( b_{pl}^{(r)} = 1 \) and \( s_{ij}^{(1)} = 0 \) \( \Rightarrow (1+1-2 \cdot 1) = 0 \), and (b) \( b_{pl}^{(r)} = 0 \) and \( s_{ij}^{(1)} = 0 \) \( \Rightarrow (0+0-2 \cdot 0) = 0 \). On the other hand, if the tie from actor \( i \) to actor \( j \) does not match the tie from position \( P \) to position \( l \), there is an inconsistency, and it should be captured by the objective function. Again, this can occur in two ways: (a) \( b_{pl}^{(r)} = 1 \) and \( s_{ij}^{(1)} = 0 \) \( \Rightarrow (1+0-2 \cdot 1) = 1 \), and (b) \( b_{pl}^{(r)} = 0 \) and \( s_{ij}^{(1)} = 1 \) \( \Rightarrow (0+1-2 \cdot 0) = 1 \). As for constraints, constraint (1b) requires each actor to be assigned to a single position, while constraint (1c) mandates that at least one actor be assigned to each position. Finally, constraint (1d) restricts the set of decision variables to be binary.

Linearization: As Brusco and Steinley (2009) note, the above formulation is non-linear via the terms \( x_{i,p}^{(r)} x_{j,l}^{(r)} \). With this in mind, we can linearize the formulation by substituting the variable \( y_{i,j,p,l}^{(r)} \) for \( x_{i,p}^{(r)} x_{j,l}^{(r)} \) in the objective function and subsequently adding the constraints

\[
\begin{align*}
y_{i,j,p,l}^{(r)} & \leq x_{i,p}^{(r)}, \quad \forall i \in N, p, l \in P \\
y_{i,j,p,l}^{(r)} & \leq x_{j,l}^{(r)}, \quad \forall i \in N, p, l \in P \\
y_{i,j,p,l}^{(r)} & \geq x_{i,p}^{(r)} + x_{j,l}^{(r)} - 1, \quad \forall i \in N, p, l \in P \\
y_{i,j,p,l}^{(r)} & \geq 0, \quad \forall i \in N, p, l \in P.
\end{align*}
\]

Specifically, if both \( x_{i,p}^{(r)} \) and \( x_{j,l}^{(r)} \) equal 1, then we need to accrue any inconsistencies in the objective function. In this case, the constraints above reduce to \( y_{i,j,p,l}^{(r)} \geq 1 \) and \( y_{i,j,p,l}^{(r)} \leq 1 \), implying \( y_{i,j,p,l}^{(r)} = 1 \), and the objective function increases by \( (b_{pl}^{(r)} + s_{ij}^{(1)} - 2 b_{pl}^{(r)} s_{ij}^{(1)}) \). On the other hand, if \( x_{i,p}^{(r)} \) or \( x_{j,l}^{(r)} \) equal 0, inconsistencies should not be accrued. When this occurs, we have \( y_{i,j,p,l}^{(r)} \geq 0 \) and \( y_{i,j,p,l}^{(r)} \leq 0 \), implying \( y_{i,j,p,l}^{(r)} = 0 \), and the objective function value remains the same.

Application to hypothetical example

Returning to our hypothetical example, we can apply Dabkowski et al.’s (2016) exploratory procedure to its one-mode network (Panel (a) of Figure 3). In this case, we have \( N = 5 \) actors and \( P = 2 \) positions, implying there are 10 binary decision variables that need to be assigned for each image matrix in the minimal, non-isomorphic set \( \{B_1^{(1)}, \ldots, B_{10}^{(1)}\} \). Given the small number of positions, we can easily visualize these matrices using Figure 7.

Starting with the top panel, there are 16 possible 2-position, one-mode binary image matrices, and 6 pairs of these matrices are isomorphic, implying each member of a given pair can be obtained by permuting the other member’s first row and column with its second row and column. Highlighted by red boxes, there is no need to find the optimal partition for both matrices in a given pair, as the actors assigned to the first and second positions in the first matrix can simply be swapped in the second matrix. Therefore, without loss of generality, we can remove the second matrix from each pair, yielding the 10 non-isomorphic image matrices seen in the middle panel of Figure 7.

At this point, we note that the zero matrix’s first row is equal to its second row, and its first column is equal to its second column; the first and second positions have an identical
pattern of ties. Accordingly, by definition, these positions are structurally equivalent, and they can be merged, allowing us to fit a more parsimonious \((1 \times 1)\) zero matrix instead. Of course, the same holds true for the all-ones matrix, and removing these from the 10 non-isomorphic image matrices yields the bottom panel of Figure 7. These eight image matrices constitute the minimal, non-isomorphic set \(\{B_1^{(1)}, \ldots, B_1^{(4)}\}\). 

With our indices, data, and decision variables established, we coded the formulation in C++ and solved the series of integer programs using CPLEX 12.3.0 via IBM’s Concert Technology library. This returned the unique globally optimal image matrix seen in Panel (a) of Figure 8 with the corresponding partition given in the permuted adjacency matrix in Panel (b). With four inconsistencies (denoted by red cells and font), this solution matches the one-mode portion of Pajek’s optimal solution in Panel (a) of Figure 6.

### Exact exploratory blockmodeling for two-mode networks using integer programming and structural equivalence

As noted earlier, Brusco and Steinley (2009) also formulated integer programs to find globally optimal partitions for two-mode blockmodels when the image matrix is prespecified. Given our desire to perform exact exploratory blockmodeling of multiple relation, mixed-mode networks, extending their work in a manner similar to Dabkowski et al. (2016) seems natural, and the modified optimization model is given below.

#### Two-mode optimization model

Index and set use: Let \(N_1 = \{1, \ldots, N_1\}\) denote a set of row objects (e.g. actors) where \(i \in N_1\); (b) \(N_2 = \{1, \ldots, N_2\}\) denote a set of column objects (e.g. events) where \(j \in N_2\); (c) \(P_1 = \{1, \ldots, P_1\}\) denote a set of row positions where \(p \in P_1\); (d) \(P_2 = \{1, \ldots, P_2\}\) denote a set of column positions where \(l \in P_2\); and (e) \(R = \{1, \ldots, R\}\) denote the index set of minimal, non-isomorphic \((P_1 \times P_2)\) binary image matrices \(B_2^{(q)}\) where \(q \in R\).

Data: The primary input is an \((N_1 \times N_2)\) binary affiliation matrix \(S^{(2)}\), where the rows of \(S^{(2)}\) represent the set of row objects and the columns of \(S^{(2)}\) represent the set of column objects; a value of 1 in row \(i\) and column \(j\) \((s_{ij}^{(2)} = 1)\) indicates row object \(i\) is “tied” to column object \(j\) (e.g. actor \(i\) attended event \(j\)); and a value of 0 implies otherwise. In addition, we have \(R\) ideal image matrices \(\{B_2^{(q)}\}\), where the rows and columns of \(B_2^{(q)}\) represent the set of row and column positions; a value of 1 in row \(p\) and column \(l\) \((b_{pl}^{(q)} = 1)\) indicates row position \(p\) is “tied” to column position \(l\) in image matrix \(q\); and a value of 0 implies otherwise.

Decision variables: Our decision variables are (a) \(x_{i,p}^{(q)} \in \{0,1\}\), where \(x_{i,p}^{(q)} = 1\) indicates that row object \(i\) is assigned to row position \(p\) when using image matrix \(q\), and \(x_{i,p}^{(q)} = 0\) otherwise; and (b) \(y_{j,l}^{(q)} \in \{0,1\}\), where \(y_{j,l}^{(q)} = 1\) indicates that column object \(j\) is assigned to column position \(l\) when using image matrix \(q\), and \(y_{j,l}^{(q)} = 0\) otherwise.

Formulation: For \(q \in R\), solve

\[
\min \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \sum_{p=1}^{P_1} x_{i,p}^{(q)} \sum_{j=1}^{P_2} y_{j,l}^{(q)} \left( k_{p,j}^{(q)} + s_{i,j}^{(2)} \right) - 2h_{p,j}^{(q)} \right) \tag{2a}
\]

s.t.

\[
\sum_{p=1}^{P_1} x_{i,p}^{(q)} = 1, \quad \forall i \in N_1 \tag{2b1}
\]

\[
\sum_{j=1}^{P_2} y_{j,l}^{(q)} = 1, \quad \forall j \in N_2 \tag{2b2}
\]
The objective function and constraints in the two-mode formulation behave similarly to the one-mode formulation. Specifically, the objective function, (2a), counts the number of inconsistencies between the image matrix $q$ and the assignment of row object $i$ and column object $j$ to row position $p$ and column position $l$. Constraints (2b1)/(2b2) require each row/column object to be assigned to a single row/column position, while constraints (2c1)/(2c2) mandate that at least one row/column object be assigned to each row/column position. Finally, constraints (2d1) and (2d2) restrict the set of decision variables to be binary.

**Linearization:** As with the one-mode formulation, the objective function is non-linear via the terms $x_{i,p} y_{j,l}$, and we can linearize it by substituting the variable $z_{i,j,p,l}$ for $x_{i,p} y_{j,l}$ in the objective function and adding appropriate constraints.

### Minimal set of non-isomorphic two-mode image matrices

Consistent with the one-mode optimization model, we solve the minimization problem for each member of the minimal, non-isomorphic set of image matrices. In this case, however, the image matrices are two-mode, and finding isomorphisms and assessing structural equivalence are slightly different. Specifically, unlike the one-mode image matrices in Figure 7 (which have a single set of positions and, therefore, rows and columns must be permuted together), two-mode image matrices have two sets of positions, and rows and columns can be permuted independently. In addition, unlike the positions of the one-mode image matrices in Figure 7 (which are structurally equivalent if both their rows and their columns are identical), positions in two-mode image matrices are structurally equivalent if either their rows or their columns are identical. Moreover, once an identical pair of rows (or columns) is identified, only the identical rows (or columns) are merged. To visualize these differences, consider Figure 9, which shows the distillation of all possible 2-position, two-mode binary image matrices into a minimal, non-isomorphic set.

Once again, we start with 16 possible image matrices, and 5 groups of these matrices (enclosed within red boxes) are isomorphic. For example, permuting the rows of $egin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ yields $egin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, permuting its columns yields $egin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$, and permuting its rows and columns yields $egin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Therefore, without loss of generality, we can retain the first matrix from each group and discard the others, yielding the seven non-isomorphic image matrices seen in the middle panel of Figure 9.

Moving on to structural equivalence, the four non-isomorphic image matrices inside red boxes have structurally equivalent positions which can be merged. For instance, the columns of the image matrix are identical; therefore, by definition, its event positions are structurally equivalent, and merging them yields the more parsimonious (2×1) two-mode binary image matrix $egin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$. Accordingly, if we optimally partition the actors using $egin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, there is no need to optimally partition them using $egin{bmatrix} 0 & 0 \end{bmatrix}$. Put another way, the second event position is superfluous.

An apt real-world example of this is given in Table 3 of Brusco and Steinley’s (2009) paper, where they report the globally optimal number of inconsistencies for a (5×3) blockmodel of the votes cast on 26 issues (the row objects) by the nine Supreme Court Justices (the column objects) (see Doreian et al. (2004) for a detailed description of the data). Given in Panel (a) of Figure 10, the last three rows of the image matrix are identical, and, as we argue, they can be merged into a single position, yielding the smaller (3×3) image matrix in Panel (b).

To verify this computationally, we used Brusco and Steinley’s (2009) integer program to find the globally

---

**Figure 9.** Distillation of the minimal, non-isomorphic set of (2×2) two-mode binary image matrices.

$$
\sum_{i=1}^{N} x_{i,p} \geq 1, \quad \forall p \in P_1 \quad (2c1)
$$

$$
\sum_{j=1}^{N} y_{j,l} \geq 1, \quad \forall l \in P_2 \quad (2c2)
$$

$$
x_{i,p} \in \{0,1\}, \quad \forall i \in N_1, p \in P_1 \quad (2d1)
$$

$$
y_{j,l} \in \{0,1\}, \quad \forall j \in N_2, l \in P_2 \quad (2d2)
$$
optimal solution for the \((5 \times 3)\) and \((3 \times 3)\) image matrices, and, as expected, both the number of inconsistencies \((22)\) and the block membership of the merged and unmerged row positions matched. Accordingly, the fourth and fifth row positions of the \((5 \times 3)\) image matrix are unnecessary, and, in the spirit of parsimony, they should be discarded. In the same way, we can remove the four \((2 \times 2)\) image matrices with structurally equivalent positions from Figure 9, leaving the minimal, non-isomorphic set in the bottom panel.

Of course, as the number of row and/or column positions increases, deriving the minimal, non-isomorphic set can no longer be done by hand. Accordingly, we developed a simple algorithm in the open source, statistical software R, and we verified it with known results (see Appendix II in the online supplemental material). As Table 1 clearly highlights, the computational advantage of fitting these minimal, non-isomorphic sets is tremendous, and it increases as the number of actor and event positions grow. For example, if we were interested in finding the globally optimal \((4 \times 4)\) two-mode binary image matrix and partition for a given network and we fit all possible image matrices, we would need to solve 65,536 integer programs. In fact, we only need to find globally optimal partitions for the 317 non-isomorphic image matrices, or 0.48% of the 65,536, to cover the entire set of all possible \((4 \times 4)\) image matrices. Alternatively, if we fit the 153 minimal, non-isomorphic image matrices, we would cover not only the entire set of all possible \((4 \times 4)\) image matrices but also every possible image matrix with four or fewer actor and event positions. This reduces our computational workload by 99.79%! Simply put, by developing sets of minimal, non-isomorphic image matrices, we can greatly extend the applicability of Brusco and Steikey’s (2009) integer programming approach to exploratory blockmodeling of two-mode networks.

**Application to hypothetical example**

Applying this approach to the two-mode portion of our hypothetical example, we have \(N_1 = 5\) actors, \(N_2 = 7\) events, \(P_1 = 2\) actor positions, and \(P_2 = 2\) event positions, implying there are 24 binary decision variables that need to be assigned for each of the three minimal, non-isomorphic image matrices in the bottom panel of Figure 9. Solving these integer programs returned the unique globally optimal image matrix seen in Panel (a) of Figure 11 with the corresponding partition given in the permuted adjacency matrix in Panel (b). With seven inconsistencies (denoted by red cells and font), this solution has two fewer inconsistencies than the two-mode portion of Pajek’s optimal solution in Panel (a) of Figure 6. That said, Pajek was simultaneously optimizing both the one- and two-mode blockmodels, and a fair comparison requires an exact multiple relation, mixed-mode approach.

**Exact exploratory blockmodeling for multiple relation, mixed-mode networks using integer programming and structural equivalence**

At this point, we have optimization models for the exact exploratory blockmodeling of one- and two-mode networks. Combining them to accommodate multiple relation, mixed-mode networks is natural, and the optimization model is given below.

**Mixed-mode optimization model**

Index and set use: Let \((a)\) \(N_1 = \{1, \ldots, N_1\}\) denote a set of actors where \(i, j \in N_1\); \((b)\) \(N_2 = \{1, \ldots, N_2\}\) denote a set of events where \(m \in N_2\); \((c)\) \(P_1 = \{1, \ldots, P_1\}\) denote a set of actor positions where \(p_1 \in P_1\); \((d)\) \(P_2 = \{1, \ldots, P_2\}\) denote a set of event positions where \(k \in P_2\); and \((e)\) \(\mathcal{R} = \{1, \ldots, R\}\) denote the index set of minimal, non-isomorphic \((P_1 \times P_1)\) binary image matrices \((B_1^{(r)} \mid B_2^{(r)})\) where \(r, q \in \mathcal{R}\) and \(r = q\).

Data: The primary inputs are \((a)\) an \((N_1 \times N_2)\) binary adjacency matrix \(S^{(i)}\), where the rows and columns of \(S^{(i)}\) represent the set of actors; \((b)\) a value of 1 in row \(i\) and column \(j\) \((s_{ij}^{(i)} = 1)\) indicates actor \(i\) is “tied” to actor \(j\); and a value of 0 implies otherwise; and \((c)\) an \((N_1 \times N_2)\) binary affiliation matrix \(S^{(2)}\), where the rows of \(S^{(2)}\) represent the set of actors and the columns of \(S^{(2)}\) represent the set of events; \((d)\) a value of 1 in row \(i\) and column \(m\) \((s_{im}^{(2)} = 1)\) indicates actor \(i\) is “tied” to event \(m\) (e.g., actor \(i\) attended event \(m\)); and \((e)\) a value of 0 implies otherwise. In addition, we have \(R\) ideal image matrices \((B_1^{(r)} \mid B_2^{(r)})\), where the rows and columns of \(B_1^{(r)}\) and the rows of \(B_2^{(r)}\) represent the set of actor positions, and \((f)\) the columns of \(B_1^{(r)}\) represent the set of event positions; \((g)\) a value of 1 in row \(p\) and column \(l\) of \(B_1^{(r)}\) \((b_{pl}^{(r)} = 1)\) indicates actor position \(P_1\) is “tied” to actor position \(P_1\), and a value of 0 implies otherwise; and \((h)\) a value of 1 in row \(p\) and column \(k\) of \(B_2^{(r)}\) \((b_{rk}^{(r)} = 1)\) indicates actor position \(p\) is “tied” to event position \(k\), and a value of 0 implies otherwise.

Decision variables: Our decision variables are \((a)\) \(x_{ij}^{(r,p)} \in \{0, 1\}\), where \(x_{ij}^{(r,p)} = 1\) indicates that actor \(i\) is assigned to actor position \(p\) when using image matrix.
Table 1. Total number of image matrices to fit for two-mode blockmodels with four or fewer actor and event positions.

| Number of positions \((P, P_2)\) | All possible image matrices | Non-isomorphic image matrices | Non-isomorphic image matrices without structurally equivalent positions | Percentage of all possible image matrices to fit |
|----------------------------------|-----------------------------|-------------------------------|-------------------------------------------------|-----------------------------------------------|
| (2, 2)                           | 16                          | 7                             | 3                                               | 18.75%                                         |
| (2, 3)                           | 64                          | 13                            | 3                                               | 4.69%                                          |
| (3, 2)                           | 64                          | 13                            | 3                                               | 4.69%                                          |
| (3, 3)                           | 512                         | 36                            | 12                                              | 2.34%                                          |
| (3, 4)                           | 4096                        | 87                            | 19                                              | 0.46%                                          |
| (4, 3)                           | 4096                        | 87                            | 19                                              | 0.46%                                          |
| (4, 4)                           | 65,536                      | 317                           | 94                                              | 0.14%                                          |
| Total image matrices to fit      | 74,384                      | 560                           | 153                                             | 0.21%                                          |

**Formulation:** For \(r, q \in R\) and \(r = q\), solve

\[
\min \sum_{p, l=1}^{P_l} \sum_{i, j=1}^{N_{r, p}} x_{i, j}^{(r, q)} + \sum_{m=1}^{N_{r, m}} \sum_{p=1}^{P_l} y_{m, k}^{(r, q)} + \sum_{m=1}^{N_{r, m}} \sum_{k=1}^{P_l} \sum_{l=1}^{P_l} z_{i, j, k}^{(r, q)} \left(2b_{p, p} + \gamma_{l, m}\right)
\]

s.t.

\[
\sum_{p=1}^{P_l} x_{i, p}^{(r, q)} = 1, \quad \forall i \in N_1
\]

\[
\sum_{p=1}^{P_l} y_{m, k}^{(r, q)} = 1, \quad \forall m \in N_2
\]

\[
\sum_{i=1}^{N_{r, i}} x_{i, p}^{(r, q)} \geq 1, \quad \forall p \in P_1
\]

\[
\sum_{m=1}^{N_{r, m}} y_{m, k}^{(r, q)} \geq 1, \quad \forall k \in P_2
\]

\[
x_{i, p}^{(r, q)} \in \{0, 1\}, \quad \forall i \in N_1, p \in P_1
\]

\[
y_{m, k}^{(r, q)} \in \{0, 1\}, \quad \forall m \in N_2, k \in P_2
\]

**Discussion:** The above formulation is essentially the aggregation of the one- and two-mode formulations. In particular, the objective function, (3a), counts the total number of inconsistencies between \(B^{(r)}\) and \(B^{(q)}\) and the assignment of actor \(i\) and actor \(j\) to position \(p\) and actor position \(l\), respectively, as well as between \(B^{(q)}_1\) and \(B^{(q)}_2\) and the assignment of actor \(i\) and event \(m\) to actor position \(p\) and event position \(k\), respectively. Constraints (3b1)/(3b2) require each actor/event to be assigned to a single actor/event position, while constraints (3c1)/(3c2) mandate that at least one actor/event be assigned to each actor/event position. Finally, constraints (3d1) and (3d2) restrict the set of decision variables to be binary.

**Linearization:** Consistent with the one- and two-mode formulations, the objective function is non-linear via the terms \(x_{i, p}^{(r, q)}\) and \(y_{m, k}^{(r, q)}\). Accordingly, we linearize it by substituting the variables \(x_{i, p}^{(r, q)}\) for \(x_{i, j}\) and \(y_{m, k}^{(r, q)}\) for \(y_{m, k}\) in the objective function and subsequently adding appropriate constraints.

**Minimal set of non-isomorphic multiple relation, mixed-mode image matrices**

As with the two-mode network, to derive our minimal, non-isomorphic set of mixed-mode binary image matrices we must modify our notions of finding isomorphisms and assessing structural equivalence. Specifically, in a multiple relation, mixed-mode network, the rows of the \((P_1 \times P_1) \times (P_1 \times P_2)\) image matrix reflect the single set of actor positions, while the first \(P_1\) columns and the last \(P_2\) columns reflect actor and event positions, respectively. Accordingly, the rows and the first \(P_1\) columns of the image matrix must be permuted together, while the last \(P_2\) columns can be permuted independently. Similarly, in order for two actor positions to be structurally equivalent in a multiple relation, mixed-mode network, their rows must be identical across the \(P_1 + P_2\) columns of the image matrix, and
Dabkowski et al.

their columns must be identical in the \( P_1 \times P_2 \) portion. If structurally equivalent actor positions are found, their corresponding rows and columns are merged. Less stringently, two event positions are structurally equivalent if their columns are identical in the \((P_1 \times P_2)\) portion, and, if identified, these columns are merged.

Assuming there are two actor and two event positions, we can illustrate this using Figure 12. Starting with the top panel, there are 28 or 256 possible \( |P_1| \times |P_2| \) multiple relation, mixed-mode binary image matrices, which can be grouped into isomorphism classes (enclosed within red boxes). For example, permuting the actor positions of 

\[
\begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 \\
1 & 0 & 1 & 0 \\
1 & 0 & 1 & 0
\end{bmatrix}
\]

yields 

\[
\begin{bmatrix}
1 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}
\]

by permuting its event positions yields 

\[
\begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix}
\]

and permuting both its actor and event positions yields 

\[
\begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 \\
1 & 0 & 0 & 1 \\
1 & 0 & 0 & 1
\end{bmatrix}
\].

Accordingly, we can retain 

\[
\begin{bmatrix}
1 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 \\
1 & 0 & 0 & 1 \\
1 & 0 & 0 & 1
\end{bmatrix}
\]

and discard the other three image matrices. Applying this approach to the remaining 252 image matrices generates 87 additional non-isomorphic image matrices, which are notionally represented in the middle panel of Figure 12. Of these 88, 38 have at least one pair of structurally equivalent positions, leaving 50 non-isomorphic image matrices without structurally equivalent positions. For instance, the columns of the \( P_1 \times P_2 \) portion of 

\[
\begin{bmatrix}
1 & 0 & 1 & 1 \\
0 & 1 & 0 & 0
\end{bmatrix}
\]

are identical. Thus, the event positions are structurally equivalent and can be merged, yielding the more parsimonious \(2 \times 2|(1 \times 1)\) image matrix 

\[
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}
\].

Even for a small number of positions, manually deriving the minimal, non-isomorphic set of multiple relation, mixed-mode image matrices is tedious and error-prone. As such, we made minor modifications to our two-mode algorithm and implemented it in R (see Appendix III in the online supplemental material). The results of running it for three or fewer actor and event positions are given in Table 2, and, once again, there are significant computational savings. Specifically, if we want to find the globally optimal multiple relation, mixed-mode blockmodels with three or fewer actor and event positions, we only need to fit 2.71% or 8148 of the 300,364 possible image matrices. Admittedly, this is still a significant computational chore, but it is two orders of magnitude more manageable.

**Application to hypothetical example**

Armed with the model in section “Optimization model,” we can now find the globally optimal blockmodel (or blockmodels) for our hypothetical multiple relation, mixed-mode network (see in Panel (a) of Figure 13). As with the 2-mode network, our integer program has 24 binary decision variables that need to be assigned for each of the 50 image matrices in the minimal, non-isomorphic set, and solving them yields the 2 global optima seen in Figure 13.
The first solution, given in Panels (b1) and (c1), is identical to the solution in Figure 6. With 13 inconsistencies (denoted by red cells and font), it confirms the global optimality of Pajek's heuristically generated solution. On the other hand, the second solution, seen in Panels (b2) and (c2), reflects a different social structure. Specifically, actor A1 is clustered with A2 and A3 versus A4 and A5, and A4 and A5 no longer occupy an actor position with a self-tie. Moreover, event E1 is clustered with E5, E6, and E7 versus E2, E3, and E4. In fact, subsequent rerunning of the Pajek model seen in Appendix I in the online supplemental material returned the second solution in Figure 13, as well as an alternative assignment of actors and events for the images matrices in Panels (b1) and (b2). Of course, on a basis of inconsistencies, none of these solutions is best; they are equally compelling. Confronted with this situation, the researcher would need additional, substantive evidence to select a best fit. Alternatively, she could embrace the dissonance and view the solutions as reasonable abstractions of the same network.

Analysis of the Noordin top terrorist network

Although illustrating our approach on the simple hypothetical example is useful for explanation and validation, testing it on larger, real-world networks is necessary for implementation. With this in mind, we selected a contemporary multiple relation, mixed-mode network with significant background and practical appeal, namely “Noordin’s Network”—a dark network centered on the Tanzim Qaedat al-Jihad (TQJ; or Organization for the Basis of Jihad) terror group (Everton, 2012).

Description: In 2005, Noordin Mohammed Top, a 37-year-old Muslim extremist and Indonesia’s most wanted man, established a violent offshoot of the Jemaah Islamiyah terrorist organization (International Crisis Group, 2006). Dubbed TQJ, Noordin focused on attacking Westerners in Southeast Asia, and in September 2009, just 2 months after his organization successfully bombed the JW Marriott and Ritz-Carlton Hotels in Jakarta, he was killed (Acharya, 2015). In light of his death, a natural question is who, if any, of his associates might succeed Noordin as the emir of TQJ. While available data may be sparse or of questionable quality, known or presumed relationships between terrorists provide valuable information. This approach is highlighted by Everton (2012: 43) who notes the potential value of identifying structurally equivalent actors, as “such actors are essentially substitutable for one another [and] this could indicate potential emergent leaders.”

Taking this tack, we can exploit Roberts and Everton’s (2011) relational data set, which leverages an International Crisis Group (2006) report to generate several TQJ networks, including an aggregated one-mode network reflecting trust and two-mode networks indicating participation in (or facilitation of) documented meetings or operations. In the case of the former, the trust network combines classmate, friendship, kinship, and soulmate ties between terrorists (Everton, 2012: 112). When binarized and restricted to the 40 members of Noordin’s network who were alive as of May 2006 (labeled A1 through A40), this yields the $40 \times 40$ binary one-mode network seen in the left-half of Figure 14, where the yellow-shaded row and column (A27) denotes Noordin. As for the latter, the meetings and operations networks are “stacked” horizontally, producing the $40 \times 34$ binary two-mode network seen in the right-half of Figure 14.

Results: Using the multiple relation, mixed-mode network in Figure 14 as input, we solved the optimization model
in section “Optimization model,” for all image matrices with three or fewer actor and event positions. The results of this exploratory fitting are given in Figure 15, which shows the globally optimal image matrices and number of inconsistencies. As Figure 15 shows, when there is one or two actor positions \( P_1 \) and \( P_2 \), increasing the number of event positions increases the number of inconsistencies. This phenomenon appears to contradict known results for the direct blockmodeling of a single relation, where “the optimized values of the criterion function decrease monotonically with the number of clusters” (Doreian et al., 2005: 187). That said, the minimal, non-isomorphic set of image matrices do not contain elements that can be expressed more parsimoniously; when these are considered, the apparent contradiction disappears.

For example, we could augment the globally optimal \((1 \times 1 | 1 \times 1)\) image matrix \([0 | 0]\) by one additional event position, yielding \([0 | 00]\). Following this augmentation, any partition of the 34 meetings and operations between the 2 event positions will produce 270 inconsistencies, thereby preserving the property of a monotonically decreasing criterion function. Nonetheless, the use of the minimal, non-isomorphic set and the result in Figure 15 is instructive, because it emphasizes that one event position is sufficient when there are less than three actor positions. On the other hand, when there are three actor positions, increasing the number of event positions generates a smaller number of inconsistencies. Similarly, regardless of the number of event positions, the same holds true when increasing the number of actor positions. With this in mind, on a basis of inconsistencies, the globally optimal \((3 \times 3 | 3 \times 3)\) solution is attractive, and its permuted, partitioned network is given in Figure 16.

Discussion: As seen in Figure 16, the first two actor positions—\{A2, A21, A30, A33, A36, and A39\} and \{A1, A7, A20, A25, A27, and A34\}—are essentially cliques, with each terrorist trusting every other terrorist in his position, while the third actor position is a large, less tightly knit group. In addition, terrorists in the first actor position largely share affiliation with Meeting #6 and Operation #3, while the third actor position is a large, less tightly knit group. In addition, terrorists in the first actor position largely share affiliation with Meeting #6 and Operation #3, while those in the second actor position map to Meeting #1 and Operation #8. In Everton’s (2012: 393) codebook, he notes that “[m]eetings infer the necessity of a decision, but the data do not specifically identify the decision or meeting subject.” The coupling of these meetings and operations in the two-mode portion of the optimal image matrix may provide clues. For example, Meeting #1 occurred on 7 June 2003, during

![Figure 14. Multiple relation, mixed-mode network for the 40 members of the Noordin Top Terrorist Network who were alive as of May 2006 (Roberts and Everton, 2011). The one-mode network (located to the left of the vertical red line) reflects the trust between terrorists, where a “1” in cell \((i, j)\) indicates that terrorist \(i\) trusts terrorist \(j\). The two-mode network (located to the right of the vertical red line) reflects the terrorists’ participation in (or facilitation of) 34 documented meetings or operations, where M1 through M20 represent meetings, O1 through O14 denote operations, and a “1” in cell \((i, m)\) indicates terrorist \(i\) participated in event \(m\) (see supplementary document of Everton (2012) for additional details on these events). The yellow-shaded row and column (A27) denotes Noordin Mohammad Top, the self-proclaimed emir of the TQJ terrorist group, who was subsequently killed in September 2009 (Acharya, 2015).](image-url)
which Noordin “met the secretary of the [Jemaah Islamiyah’s] central command” (International Crisis Group, 2006: 5). Almost exactly 2-months later, four of the five terrorists who attended Meeting #1 facilitated or participated in Operation #8—the bombing of the Marriot Hotel in South Jakarta. Based on their shared affiliation in these seemingly synchronized events, it seems likely that the Marriot Hotel bombing was discussed.

Moreover, given our desire to identify potential replacements for Noordin (A27), the terrorists in the second actor position represent likely candidates. That said, at the time of Noordin’s death in September 2009, every member of this position was incarcerated and either sentenced to death or serving prison sentences through at least mid-2011 (International Crisis Group, 2006; Quiano, 2008). With this in mind, the first actor position provides the next logical place to look. After all, several members of the second position (including Noordin) trust members of the first position, and Noordin participated in multiple events with its members. Taking this approach, we find that four terrorists in the first position—{A2, A33, A36, and A39}—were free at the time of Noordin’s death (International Crisis Group, 2006). Of these, A39 (Urwah, who has been described as a “key lieutenant” [Stewart, 2009]) was killed alongside Noordin, and A36 (Ubeid) had left TQJ for another terrorist organization (International Crisis Group, 2009). With only two members of the first position left, A2 (Abu Fida)—the only free member of the first position trusted by Noordin—seems to be the likely successor.

Ultimately, however, a heretofore unknown named Syaifudin Jaelani, who operated “in a chameleon-like fashion . . . [and] patiently waited for three years to prepare suicide bombers” for the July 2009 attacks in Jakarta (Christanto, 2009), took the reins from Noordin. Syaifudin does not appear in Roberts and Everton’s (2011) relational data set, and no approach, no matter how exact, can make him materialize. This is not an indictment of our approach. It simply highlights a challenge of analyzing dark networks, namely, the actors typically prefer to remain anonymous. That said, Everton (see Everton and Cunningham, 2015) has expanded the size of the Noordin dataset to 237 individuals. Although this version is not yet publicly available, it is possible Syaifudin is among this expanded set.

**Limitations**

Among the limitations of our exact exploratory approach, the principal shortcoming is its scalability, especially as the number of actor and event positions grow. After all, in this paper, we restricted our real-world analysis to image

![Figure 15. Globally optimal image matrices for the multiple relation, mixed-mode network of the 40 alive members of the Noordin Top Terrorist Network, where the error score corresponding to each globally optimal \(P_1 \times P_2 \times P_3\) image matrix is given at the bottom left of the matrix.](image-url)
matrices with three or fewer actor and event positions, and, while we feel this was illustratively and substantively sufficient, adding a fourth position makes the problem practically infeasible. Specifically, when we move from a \( \binom{3}{3} \times \binom{3}{3} \) blockmodel, the number of possible binary image matrices increases from 262,144 to 4,294,967,296. Quite literally, the problem is huge.

To combat such issues, Brusco and Steinley (2009: 584) recommend “a two-stage procedure [could be employed] . . . where a heuristic method is first used to identify an image matrix and the integer program is subsequently formulated and solved to identify the optimal solution for that image matrix.” Adopting this approach, our multiple relation, mixed-mode method in Pajek provides an attractive heuristic, as our experience suggests it yields exact solutions in a very reasonable amount of time. For example, when applied to the Noordin Top Terrorist Network in Figure 14, Pajek completed its 200 repetitions in less than 4 minutes, and the number of inconsistencies, image matrix, and partition of its best solution matched the globally optimal result in Figure 16 with one minor exception—Meeting #1 was no longer clustered with Operation #8; it was in the third event position. This not only reinforces previous findings on the excellent performance of the relocation routine used in Pajek but also reemphasizes the possibility of multiple global optima.

In fact, when one considers that CPLEX requires specialized training and Pajek is quite accessible, one might question the value of the integer programming approach at all. However, it is crucial to remember that objectively and definitively assessing the quality of Pajek’s solutions requires an exact method, and our integer programming approach fills this void. In sum, we see our exact and heuristic procedures as symbiotic, and both are currently necessary to analyze real-world multiple relation, mixed-mode problems of a non-trivial size.

**Conclusion**

From the outset, computational sociologists have stressed leveraging multiple relations when blockmodeling social networks. In the current era of big data and fast computation, our ability to harvest such networks and analyze them in an exact exploratory way has never been greater. Accordingly, in this paper we extended Brusco and Steinley’s (2009) exact procedure for the confirmatory blockmodeling of a single relation to the exploratory blockmodeling of multiple relation, mixed-mode networks. Given the computational complexity of this problem, we developed an algorithm to generate a minimal, non-isomorphic set of \( \binom{P_1 \times P_1}{P_1 \times P_2} \) image matrices, greatly reducing the size of the problem. Unfortunately, the benefits of this approach quickly erode as the number of event and actor positions increases. In light of this limitation, we tested the performance of our multiple relation, mixed-mode Pajek heuristic on a real-world network—a sparse \( \binom{40 \times 40}{40 \times 34} \) network of terrorists, and the results were outstanding. Nonetheless, without a complementary exact approach, the quality of Pajek’s solutions would remain unknown; therefore, both methods are required.

Although we formulated our exact approach for multiple relation, mixed-mode networks with a single one- and two-mode network, extending it to multiple one- and two-mode
networks is relatively straightforward. In fact, given the current emphasis on multilevel networks (e.g., Lazega and Snijders, 2015), this appears to be the next logical step. That said, unless we migrate to high performance / high throughput computing systems, increasing the number of relations without improving the efficiency of the current formulation seems somewhat fruitless. With this in mind, we hope this paper will inspire creative efforts to extend the applicability of exact methods to multiple relation, mixed-mode networks.

Acknowledgements
The views expressed herein are those of the author(s) and do not reflect the official policy or position of the United States Military Academy, the Department of the Army, or the Department of Defense. In addition, author contributions are as follows: M.D., N.F., and R.B. designed research, performed research, and analyzed data; M.D. wrote the paper.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iDs
Matthew F Dabkowski https://orcid.org/0000-0002-0813-7672
Ronald Breiger https://orcid.org/0000-0003-0575-9211

Supplemental material
Supplemental material for this article is available online.

References
Acharya A (2015) Whither Southeast Asia Terrorism? London: Imperial College Press.
Baker WE (1986) Three-dimensional blockmodels. Journal of Mathematical Sociology 12(2): 191–223.
Batagelj V, Fertigoj A and Doreian P (2007) Indirect blockmodeling of 3-way networks. In: Brito P, Cucumel G, Bertrand P, et al. (eds) Selected Contributions in Data Analysis and Classification. New York: Springer-Verlag, pp. 151–159.
Boorman SA and White HC (1976) Social structure from multiple networks. II. Role structures. American Journal of Sociology 81(6): 1384–1446.
Borgatti SP and Everett MG (1992) Regular blockmodels of multway, multimode matrices. Social Networks 14(1): 91–120.
Brandes U and Lerner J (2010) Structural similarity: Spectral methods for relaxed blockmodeling. Journal of Classification 27(3): 279–306.
Breiger RL, Boorman S and Arabie P (1975) An algorithm for clustering relational data, with applications to social network analysis and comparison with multi-dimensional scaling. Journal of Mathematical Psychology 12: 328–383.
Brusco M, Doreian P, Mrvar A, et al. (2013a) An exact algorithm for blockmodeling of two-mode network data. The Journal of Mathematical Sociology 37(2): 61–84.
Brusco M, Doreian P, Steinley D, et al. (2013b) Multiobjective blockmodeling for social network analysis. Psychometrika 78(3): 498–525.
Brusco M and Steinley D (2009) Integer programs for one- and two-mode blockmodeling based on prespecified image matrices for structural and regular equivalence. Journal of Mathematical Psychology 53: 577–585.
Christanto D (2009) Special Report: Saifudin Jaelani; Noordin’s prodigy in action. The Jakarta Post, 15 August. Available at: https://www.thejakartapost.com/news/2009/08/15/special-report-saifudin-jaelani-noordin039s-prodigy-action.html
Dabkowski M, Breiger R and Sizdarovszky F (2015) Simultaneous-direct blockmodeling for multiple relations in Pajek. Social Networks 40: 1–16.
Dabkowski M, Fan N and Breiger R (2016) Exploratory blockmodeling for one-mode, unsigned, deterministic networks using integer programming and structural equivalence. Social Networks 47: 93–106.
Doreian P (2006) Some open problem sets for generalized blockmodeling. In: Batagelj V, Bock H, Fertigoj A, et al. (eds) Data Science and Classification. New York: Springer-Verlag, pp. 119–130.
Doreian P, Batagelj V and Fertigoj A (2004) Generalized blockmodeling of two-mode network data. Social Networks 26(1): 29–53.
Doreian P, Batagelj V and Fertigoj A (2005) Generalized Blockmodeling. Cambridge: Cambridge University Press.
Everton SF (2012) Disrupting Dark Networks. Cambridge: Cambridge University Press.
Everton SF and Cunningham D (2015) Dark network resilience in a hostile environment: Optimizing centralization and density. Journal of Criminology, Criminal Justice, Law & Society 16(1): 1–20.
International Crisis Group (2006) Terrorism in Indonesia: Noordin’s networks. Asia report no. 114, 5 May. Available at: https://www.crisisgroup.org/asia/south-east-asia/indonesia/terrorism-indonesia-noordin-s-networks
International Crisis Group (2009) Indonesia: Noordin top’s support base. Asia briefing no. 95, 27 August. Available at: https://www.crisisgroup.org/asia/south-east-asia/indonesia/indonesia-noordin-top-s-support-base
Lazega E and Snijders TA (2015) Multilevel Network Analysis for the Social Sciences—Theory, Methods and Applications. New York: Springer.
Mrvar A and Batagelj V (2013) Pajek64—Program for analysis and visualization of very large networks (version 3.12). Available at: mrvr.fdv.uni-lj.si/pajek (accessed 1 July 2013).
Quiano K (2008) Indonesian court sentences key militant. CNN, 21 April. Available at: www.cnn.com/2008/WORLD/asiapcf/04/21/indonesia.militant (accessed 2 September 2019).
Roberts N and Everton SF (2011) Roberts and Everton terrorist data: Noordin top terrorist network (subset). Available at: sites.google.com/site/ucinetsoftware/datasets/covert-networks/noordin-top (accessed 2 September 2019).
Stewart S (2009) The death of a top Indonesian militant. 23 September. Available at: https://worldview.stratfor.com/article/death-top-indonesian-militant (accessed 2 September 2019).

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

White HC, Boorman SA and Breiger RL (1976) Social structure from multiple networks. I. Blockmodels of roles and positions. *American Journal of Sociology* 81(4): 730–780.

Žiberna A (2014) Blockmodeling of multilevel networks. *Social Networks* 39: 46–61.

**Author biographies**

Matthew F Dabkowski is an associate professor in the United States Military Academy’s Department of Systems Engineering, where he serves as the Director of the Systems Engineering Program. He holds a PhD in Systems and Industrial Engineering from the University of Arizona, and his research interests include multiple objective decision making, applied statistics, social network analysis, and simulation modeling.

Neng Fan is an associate professor in the Department of Systems and Industrial Engineering at the University of Arizona, and he received his PhD in Industrial and Systems Engineering from the University of Florida. His research focuses on the development of various optimization methodologies and their applications in energy systems, renewable energy integration, healthcare, sustainable agriculture, and data analytics.

Ronald Breiger is a regents professor and a professor in the School of Sociology at the University of Arizona. His PhD in Sociology is from the Harvard University. His research is in the areas of social network analysis, networks and culture, estimation of case-specific contributions to regression coefficients, and measurement issues in cultural and institutional analysis.