Social Networks 56 (2019) 70–80

Contents lists available at ScienceDirect

Social Networks

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

Partitioning signed networks using relocation heuristics, tabu search, and variable neighborhood search

Michael J. Brusco[a], Patrick Doreian[b,c,*]

[a] Florida State University, United States
[b] University of Ljubljana, Slovenia
[c] University of Pittsburgh, United States

ARTICLE INFO

Keywords:
Signed networks
Structural balance
Partitioning
Heuristics
Tabu search
Variable neighborhood search

ABSTRACT

Recently, there have been significant advancements in the development of exact methods and metaheuristics for partitioning signed networks. The metaheuristic advancements have led commonly to adverse implications for multiple restart (multistart) relocation heuristics for these networks. Most notably, it has been reported that multistart relocation heuristics are not computationally feasible for large signed networks with thousands or tens of thousands of vertices. In this paper, we show that combining multistart relocation heuristics with tabu search or variable neighborhood search can rapidly produce partitions of the vertices of signed networks that are competitive with those obtained using existing metaheuristics.

1. Introduction

The problem of partitioning the vertices of a signed network has applications in a variety of different scientific contexts. This includes partitioning for signed social networks studied within the general rubric of structural balance theory with clear substantive concerns originally articulated by Heider (1946) and later extended by Cartwright and Harary (1956); Davis (1967), and Doreian and Mrvar (1996, 2009). This partitioning is important for testing substantive theories (Doreian and Mrvar, 2014), adapting methods (Doreian, 2008), and tracking imbalance over time (Doreian and Mrvar, 2015). Of additional interest is the study of signed networks in other contexts and fields (Facchetti et al., 2011, Huffner et al., 2010; Yang et al., 2007; Kim et al., 2014). Further, Aref et al. (2017) and Levorato et al. (2017) recently noted examples in the physical sciences including: (i) chemistry: the study of fullerene graphs pertaining to carbon allotropes (Došlić and Vukičević, 2007), (ii) biology: measuring the distance of a biological network from monotonicity (Iacono et al., 2010), and (iii) physics: the study of energy states (Kasteleyn, 1963). In the arena of political and social science, signed networks have been analyzed to investigate voting patterns of the United Nations General Assembly (Doreian et al., 2013; Doreian and Mrvar, 2015). Recent literature reviews pertaining to the analysis of signed networks are provided by Tang et al. (2016) and Traag et al. (2018).

For all these applications, getting the partitioning done correctly and efficiently is important, especially so when the available signed network data sets are larger. Broadly, there are a variety of alternative approaches for community detection in the context of signed networks. These include, but are not necessarily limited to, modularity-based approaches (Anchuri and Magdon-Ismail, 2012), spectral clustering (Kunegis et al., 2010), mixture-modeling (Chen et al., 2013), and dynamic model-based algorithms (Yang et al., 2007). Our focus herein is on direct partitioning approaches for signed networks (Doreian and Mrvar, 1996; Bansal et al., 2004; Traag and Bruggeman, 2009) with emphasis on methods that have great potential for partitioning larger signed networks.

Our focus in this paper is on undirected signed networks associated with a vertex set $V = \{1,\ldots, n\}$, edge set $E = \{u, v\}$, and edge weights w_{uv} (for all $\{u, v\} \in E$).[1] The edge set is partitioned into two subsets E^+ and E^-, which correspond to positive and negative edges, respectively. There are several common formulations of optimization problems for the partitioning of signed networks. Perhaps the most general of these is the correlation clustering problem, where the goal is to partition the vertex set into clusters to minimize the total level of *frustration* or inconsistency in the network. While recognizing that there are alternative objective functions, such as weighed functions and positive and negative inconsistencies (Doreian and Mrvar, 1996) and Hamiltonians (Traag and Bruggeman, 2009), we restrict our attention to the popular

* Corresponding author.
E-mail addresses: mbrusco@business.fsu.edu (M.J. Brusco), pitpat@pitt.edu (P. Doreian).

[1] The methods described in this paper can also be applied to directed networks. The assumption of undirected networks is made for consistency with the network problems analyzed later in the paper.

https://doi.org/10.1016/j.socnet.2018.08.007

criterion of minimizing the total level of frustration.

We note that the frustration index is formally identical to the line index of imbalance used in structural balance partitioning but only for two clusters. Moving beyond partitions having only two clusters is critical. With correlation clustering, the number of clusters is not specified in advance. Correlation clustering is based on principles of weak structural balance described by Davis (1967) building on the work of Cartwright and Harary (1956). Frustration occurs when two vertices sharing a positive edge are in different clusters, or when two vertices sharing a negative edge are in the same cluster. Mathematical programming formulations for the correlation clustering problem have been described by Figueiredo and Moura (2013) and Aref et al. (2017). Metaheuristics for large networks have also been proposed and include genetic algorithms (Zhang et al., 2010) and iterated local search (Levorato et al., 2017).

Some authors have focused on a variation of the correlation clustering problem whereby a desired number of clusters, K, is prespecified (Brusco and Steinley, 2010; Doreian and Mrvar, 1996; Giotis and Guruswami, 2006). A branch-and-bound algorithm for this *K-balance partitioning* problem was proposed by Brusco and Steinley (2010); however, it is only scalable for small networks ($n < 30$). Preferable exact approaches based on mathematical programming have been proposed by Figueiredo and Moura (2013) and Aref et al. (2017). For larger problems, Doreian and Mrvar (1996) proposed a relocation heuristic. The special case of $K = 2$ for the K-balance partitioning problem is of considerable theoretical interest because it corresponds to minimization of the frustration index via a partition of the vertex set so as to obtain the minimum number of edges that must be removed so as to bring the network into balance. Aref et al. (2017) present a formal treatment of this problem within a graph coloring framework and provide an efficient exact solution approach using integer linear programming.

Despite advancements in the development of exact procedures (Aref et al., 2017; Brusco and Steinley, 2010; Brusco et al., 2011; Figueiredo and Moura, 2013), heuristic procedures remain useful for the partitioning of large signed networks (e.g., $n > 1000$). The relocation heuristic developed by Doreian and Mrvar (1996) is one of the most general heuristic procedures for partitioning signed networks. It begins with an initial (often randomly generated) partition that is refined by two local-search operations: (i) relocation of each vertex from its current cluster to one of the other clusters, and (ii) pairwise interchanges (or exchanges) of the cluster memberships for each pair of vertices that are not currently in the same cluster. The implementation of the relocation heuristic in the Pajek software system (see de Nooy et al., 2011) has been shown to perform well on small real-world instances of K-balance partitioning problems (Brusco and Steinley, 2010; Figueiredo and Moura, 2013); however, its performance deteriorated somewhat for synthetic random networks with roughly 50 vertices (Figueiredo and Moura, 2013). Of greater concern is the reported computational impracticality of the relocation heuristic for problems with $n > 1000$ vertices (Levorato et al., 2017). This problem needs to be addressed.

In an effort to tackle large instances of correlation clustering and K-balance partitioning problems, several researchers have focused on the development of metaheuristics such as genetic algorithms (Goldberg, 1989), greedy randomized adaptive search procedure (GRASP: Feo and Resende, 1995), variable neighborhood search (Hansen and Mladenovic, 1997), and iterated local search (Lourenco et al., 2003, 2010). More specifically, genetic algorithm approaches have been designed by Zhang et al. (2008) and Ma et al. (2015). Drummond et al. (2013) obtained promising results with the GRASP method. Most recently, however, Levorato et al. (2017) showed that their implementation of iterated local search convincingly outperformed the greedy relocation heuristics of Doreian and Mrvar (1996) and Elsner and Schudy (2009), GRASP, and an implementation of variable neighborhood search.

Our specific operational goals in this paper are twofold. First, we demonstrate that an efficient multistart implementation of the relocation heuristic is often scalable for networks with far more than 1000 vertices. Second, and more importantly, we show that the relocation heuristic is also an effective engine for metaheuristics for K-balance partitioning and, by extension, correlation clustering. More specifically, we recommend a two-phase approach. The first phase uses the multistart relocation heuristic to establish a good initial solution and either tabu search (Glover, 1989, 1990; Glover and Laguna, 1993) or variable neighborhood search (Mladenovic and Hansen, 1997) is used in the second phase to refine the solution. We evaluate this procedure using test problems considered by Levorato et al. (2017), which are slices of the Slashdot zoo data (Leskovec et al., 2010). We also apply the method to the full Slashdot zoo network and the Wiki elections data (Leskovec et al., 2010). Our results compare very favorably to the methods evaluated by Levorato et al. (2017). Moreover, we show that high-quality solutions are achievable using *far fewer* clusters than reported by Levorato et al. (2017).

Section 2 presents a formal presentation of the K-balance partitioning problem. This section also includes a description of the relocation heuristic, tabu search, and variable neighborhood search procedures. Section 3 presents an evaluation of the two-phase procedure using the test problems from the Levorato et al. (2017) study. The paper concludes in section 4 with a summary of the findings and suggestions for future research.

2. Generalized structural balance partitioning (K-balance partitioning)

2.1. The optimization problem

We recall our previous definitions of V, E^+, E^-, and w_{uv} as the set of n vertices, set of positive undirected edges, set of undirected edges, and edge weights, respectively. The K-balance partitioning problem seeks a partition, $P = \{S_1, \ldots, S_K\}$, of the vertex set into $K \geq 2$ clusters, where S_k contains the vertices assigned to cluster k for all $1 \leq k \leq K$ clusters. The goal to find the partition P from the set of all possible partitions (Π) of n vertices into K clusters so as minimize the following objective criterion function:

$$Z(P) = \sum_{k=1}^{K} \sum_{\substack{(\{u,v\}\in E^-)\wedge \\ (\{u,v\}\in S_k)}} w_{uv} + \sum_{1\leq k<l\leq K} \sum_{\substack{(\{u,v\}\in E^+)\wedge \\ \left[\begin{smallmatrix}(u\in S_k\wedge v\in S_l)\vee \\ (u\in S_l\wedge v\in S_k)\end{smallmatrix}\right]}} w_{uv} \quad (1)$$

The first term in Eq. (1) is a summation of the weights of vertex pairs $\{u, v\}$ whereby u and v are both in the same cluster and are associated with a negative edge. The second term in the equation is the summation of the weights of vertex pairs $\{u, v\}$ whereby u and v are in different clusters and are associated with a positive edge. Therefore, the objective criterion function value, Z(P), is a measure of the total amount of inconsistency with perfect structural balance.

There are several possible exact solution approaches for finding the partition P that minimizes $Z(P)$. One approach, complete enumeration, is to compute $Z(P)$ for all partitions $P \in \Pi$ and select the partition that yields the minimum value. The number of partitions in Π is a Stirling number of the second kind and precludes complete enumeration for even modest values of n and K (e.g., $n = 20$ and $K = 4$). Alternatively, an implicit enumeration scheme based on branch-and-bound programming was developed by Brusco and Steinley (2010); however, it too is limited to relatively small problems. Perhaps the most robust exact procedure is based on mixed integer linear programming (Figueiredo and Moura, 2013), yet this approach is also limited to problems with $n < 50$. In light of the limitations of extant exact procedures, there is a necessary reliance on heuristic methods for large problems.