Noname manuscript No.
(will be inserted by the editor)

Correlation Clustering Problem under Mediation

Zacharie Ales · Céline Engelbeen · Rosa Figueiredo

Received: DD Month YEAR / Accepted: DD Month YEAR

Abstract In the context of community detection, Correlation Clustering (CC) provides a measure of balance for social networks as well as a tool to explore their structures. However, CC does not encompass features such as the mediation between the clusters which could be all the more relevant with the recent rise of ideological polarization. In this work, we study Correlation Clustering under mediation (CCM), a new variant of CC in which a set of mediators is determined. This new signed graph clustering problem is proved to be NP-hard and formulated as an integer programming formulation. An extensive investigation of the mediation set structure leads to the development of two efficient exact enumeration algorithms for CCM. The first one exhaustively enumerates the maximal sets of mediators in order to provide several relevant solutions. The second algorithm implements a pruning mechanism which drastically reduces the size of the exploration tree in order to return a single optimal solution. Computational experiments are presented on two sets of instances: signed networks representing voting activity in the European Parliament and random signed graphs.

Keywords Accessible system · Correlation clustering · Enumeration algorithm · Signed graph · Structural balance

Z. Ales
UMA, CEDRIC, ENSTA Paris, Institut Polytechnique de Paris, France
E-mail: zacharie.ales@ensta.fr

C. Engelbeen
Laboratoire Quaresmi, ICHEC, Brussels, Belgium
E-mail: Celine.Engelbeen@ichec.be

R. Figueiredo
Laboratoire Informatique d'Avignon, Avignon Université, France
E-mail: rosa.figueiredo@univ-avignon.fr

1 Introduction

Community detection is largely applied to understanding the structure of social networks. In the presence of a network with antithetical relationships (like/dislike, for/against, similar/different...) community detection can be modeled as correlation clustering (CC) (Doreian and Mrvar, 1996), a signed graph clustering problem later formalized by Bansal et al. (2004) for document classification.

In a signed graph, the edges are labeled as either positive (+) or negative (-). The CC problem consists in partitioning the vertices of such a graph while minimizing disagreements, i.e., the total number of positive edges between the clusters plus the total number of negative edges inside the clusters. A weighted version of the problem was lately defined in Demaine et al. (2006).

The CC problem is related to the concept of structural balance introduced in the field of social network analysis (Heider, 1946; Cartwright and Harary, 1956). According to structural balance theory, the equilibrium of a social system is associated with the propensity of individual elements to be organized in groups avoiding conflictual situations. This concept is perfectly described by graph theory (Davis, 1967). A signed graph is structurally balanced if it can be partitioned into clusters, such that all positive (resp. negative) edges are located inside (resp. in-between) these modules.

Applications of the CC problem overtakes the area of social networks analysis and also arise in system biology (DasGupta et al., 2007), portfolio analysis for risk management (Figueiredo and Frota, 2014; Harary, 2002), voting behavior (Arinik et al., 2017; Kropivnik and Mrvar, 1996), document classification (Bansal et al., 2004), surface detection in 3D images (Kolluri et al., 2004), and in the detection of embedded matrix structures (Figueiredo et al., 2011). Variants of the CC problem have been proposed and discussed in the literature. Some of them motivated by a redefinition of the concept of structural balance (Doreian and Mrvar, 2009) or by applications to community detection in unsigned graphs.

The recent rise of ideological polarization makes it harder to reach agreements across partisan lines (Abramowitz and Saunders, 2008). Since mediation could allow productive exchanges in polarized signed networks, we study a new variant of CC in which a set of key-players, called *mediators*, is additionally identified. We apply the concept of positive mediation as introduced by (Doreian and Mrvar, 2009): a set of mediators must have *good* relations among themselves and with other individuals in the network. We define a *good* relation by two parameters, α and β, which represent the maximal proportion of negative to positive relations allowed inside and outside the mediation set, respectively. The aim of the *correlation clustering problem under mediation (CCM)* is to obtain a partition in which one cluster is composed of mediators and which minimizes the imbalance (as defined in original CC) of the remaining clusters.

Unlike the CC problem, to the best of our knowledge, the CCM problem only has applications in social networks analysis. In this work, we are not only focused on identifying one optimal set of mediators (a unique optimal solution) but also on determining several of them as various as possible (multiple and diverse solutions). Indeed, in a decision aid process based on the CCM problem, such sets can be used as a basis to form committees (e.g. in political institutions): identifying alternative solutions can enable to assign different committees to different tasks (e.g., one committee per law or topic). Moreover, multiple optimal solutions can also be used to indicate the importance of each individual in the whole group. For example, if only one element is present in all the sets of mediators, it indicates that it plays a major role in the social network.

The contributions of this paper are fourfold.

1. We introduce the CCM problem, a new variant of CC in which the definition of a set of mediators is parametrized by two parameters.
2. We prove that CCM is NP-hard and formulate this problem as an integer linear programming model.
3. We provide two enumeration algorithms for CCM which take advantage of properties of sets of mediators to break symmetry in the search tree. One of these algorithms is designed for generating all the maximal sets of mediators.
4. We present extensive computational results to compare the performances of these algorithms to those of CPLEX applied to our formulation.

The paper is organized as follows. The next section is dedicated to a review of the works related to the CCM problem. We give the notations and the formal definition of this problem in Section 3 and prove its NP-hardness. We introduce an ILP formulation of the problem in Section 4. Section 5 is devoted to the enumeration algorithms. Computational experiments are given in Section 6. We finally conclude the paper in Section 7.

2 Related works

The review of the literature is divided in three sections: exact optimization methods for CC (Section 2.1), variants of CC (Section 2.2) and group selection problems treated from a network optimization point of view (Section 2.3.)

2.1 Exact methods for CC

A combinatorial branch-and-bound was proposed by Brusco and Steinley (2009) to solve instances with up to 21 vertices. An Integer Linear Programming (ILP) formulation based on the vertex clustering formulation of (Mehrotra and Trick, 1998) was also considered in the literature (see for example (Demaine et al., 2006; Arinik et al., 2017, 2021)). In (Figueiredo and Moura, 2013) the two approaches were compared. The authors showed that the ILP approach could handle larger graphs and required less time for most of the benchmark instances. This approach was used in a branch-and-cut framework on complete graphs with up to 50 vertices (Arinik et al., 2021) and on non-complete ones with up to 400 vertices (Arinik et al., 2017).

In a recent work (Arinik et al., 2021), the authors showed that the optimal solution space of the CC problem can be composed of multiple and diverse optimal solutions. The applications solved by this clustering problem motivated the same authors to develop a method for generating its complete space of optimal solutions (Arinik et al., 2023). The algorithm combines an exhaustive enumeration strategy with neighborhoods of varying sizes, to achieve computational effectiveness.

2.2 Variants of CC

The variants of the CC problem can be divided in two groups: with redefinition of the objective function or with redefinition of the clustering constraints.

2.2.1 Alternative objectives

CC seeks a partition which minimizes the total number of disagreements. Doreian and Mrvar (2009) observed that this definition does not encompass some important features. For example, vertices which agree with hostile subgroups increase the imbalance of the graph according to this definition. The authors considered that such vertices are potential mediators which should have a positive effect on the balance. Consequently, they proposed a relaxed definition of the objective as the sum of maximum disagreements inside each cluster plus the sum of maximal disagreements among each pair of clusters in the partition. The Relaxed Correlation Clustering (Figueiredo and Moura, 2013; Levorato et al., 2017; Arinik et al., 2017) (RCC) consider this objective.

Local disagreement functions have also been used in the literature. Both works presented in (Kalhan et al., 2019; Puleo and Milenkovic, 2018) are based on a disagreements vector, i.e, a vector indexed by the vertices where the i-th index is the number of disagreements at vertex i. In (Puleo and Milenkovic, 2018), the highest value in the disagreement vector is minimized while in Kalhan et al. (2019) the l_q norm of the disagreements vector is minimized.

Eventually, motivated by network analysis applications defined on unsigned graphs, Veldt et al. (2018) introduced the Lambda Correlation Clustering (LambdaCC), a weighted version of CC in which the weight of the edges is either $\lambda \in [0, 1]$ or $1 - \lambda$.

2.2.2 Alternative constraints

The first CC variant which redefines the clustering constraints is Motif Correlation Clustering (MotifCC) (Li et al., 2017). Also motivated by network analysis applications, MotifCC associates a sign, positive or negative, to subgraph structures (called motifs) and minimizes the number of clustering errors associated with both edges and motifs. This variant generalizes CC to the hypergraph setting where the order of the graph is defined by the size of the motifs considered. In Fair Correlation Clustering (FairCC) the vertex partition must satisfy fairness constraints (Ahmadian et al., 2020). In this variant, each vertex of the graph has a color associated and the colors in the partition must be distributed according to a given fair property. Figueiredo and Moura (2013) defined the first version of CC with mediation following the discussions in (Doreian and Mrvar, 2009). Their definition of a set of mediators was very restrictive and we show that the problem defined in Section 2.3 generalises it.

Different approaches have been considered to solve these problems. ILP formulations were introduced in (Figueiredo and Moura, 2013) for RCC. Approximation algorithms were proposed for LambdaCC and MotifCC (Veldt et al., 2018; Li et al., 2017; Gleich et al., 2018) as well as for FairCC (Kalhan et al., 2019; Puleo and Milenkovic, 2018). A simulated annealing was considered for MotifCC in Li et al. (2017) while Iterated Local Search methods were proposed for RCC (Levorato et al., 2017).

2.3 Group selection in social networks

Several works in the literature have been dedicated to the identification of a set of individuals playing a specific role in a network. These individuals can be named key players (Borgatti, 2006; Ortiz-Arroyo, 2010), influential vertices (Li et al., 2011), or mediators (Figueiredo and Moura, 2013). The set of vertices can be selected through a global network optimization criteria or by ranking network elements according to an individual measure (e.g., vertex centrality (Borgatti, 2003)). We focus on the first approach as the second one does not provide optimality guarantee (see examples in (Ortiz-Arroyo, 2010)).

The key players problem as introduced by (Borgatti, 2003), consists in selecting k vertices in a network that maximizes or minimizes the disruption of the residual network obtained by removing them. Different measures and heuristic procedures have been proposed in the literature for this problem (Borgatti, 2006; Ortiz-Arroyo, 2010). (Li et al., 2011) studied the problem of finding the set of key players controlling the bottlenecks of influence propagation in a social network. The authors proposed a three-steps heuristic to solve this variant, named the $k-$mediators problem. We refer the reader to references in (Li et al., 2011) for works on vertex selection for influence maximization.

None of these works considered exact methods even when the size of the networks is small (see for example (Borgatti, 2006)). The CCM problem defined in this work is based on the mediation concept described by Doreian and Mrvar (2009). It has only been treated once in the literature (Figueiredo and Moura, 2013) and for a very particular case where both parameters, α and β, defining the feasibility of a set of mediators are equal to 0.

3 Notation and problem definition

Let $G = (V, E)$ be an *undirected graph*, where V and E are the sets of vertices and edges, respectively. Consider a function $s : E \to \{+, -\}$ that assigns a *sign* to each edge in E. An undirected graph G together with a function s is called a *signed graph*, denoted here by $G = (V, E, s)$. An edge $e \in E$ is called negative if $s(e) = -$ and positive if $s(e) = +$. We note E^- and E^+ the sets of negative and positive edges in a signed graph, respectively. Let $n = |V|$.

The *imbalance* of a vertex partition is defined by its number of disagreements, that is the number of positive edges between two clusters and negative edges inside a cluster. The CC problem (Bansal et al., 2004) aims to find a partition of the vertices which minimizes the imbalance. In the weighed version of the CC problem, an extra function $w : E \to \mathbb{R}_+$ is added. In order to define the imbalance in that weighted case, let us introduce some extra notations.

For two subsets $S_1, S_2 \subseteq V$ and a sign $\sigma \in \{+, -\}$ we define $E^\sigma[S_1, S_2] = \{(i, j) \in E^\sigma : i \in S_1, j \in S_2,\ i \neq j\}$, $w^\sigma(S_1, S_2) = \sum_{(i,j)\in E^\sigma[S_1,S_2]} w_{ij}$ and $w^\sigma(S_1) = w^\sigma(S_1, S_1)$.

A *partition* of V is a division of V into non-overlapping and non-empty subsets. The *imbalance* $I(P)$ of a partition $P = \{S_1, S_2, \ldots, S_{|P|}\}$ is the weighted sum of negative arcs inside the subsets and positive arcs between the subsets, i.e.,

$$I(P) = \sum_{1\leq i\leq |P|} w^-(S_i) + \sum_{1\leq i<j\leq |P|} w^+(S_i, S_j). \tag{1}$$

As stated by Bansal et al. (2004), CC consists in finding a partition that minimizes the imbalance given by (1).

We introduce a new variant of CC in which a set of vertices called *mediators* is identified while the imbalance (1) of the remaining vertices is minimized. Let us define two properties that a set of mediators must satisfy.

Definition 1 Consider a scalar value $\alpha \in \mathbb{R}_+$. A subset $S \subseteq V$ is α-feasible if $\alpha w^+(S) \geq w^-(S)$.

Definition 2 Consider a scalar value $\beta \in \mathbb{R}_+$. A subset $S \subseteq V$ is $\beta-$feasible if $\beta w^+(S, V\backslash S) \geq w^-(S, V\backslash S)$.

These definitions provide upper bounds on the sum of negative weights inside (Definition 1) and leaving (Definition 2) the set of vertices S. Fixing parameter α to 0 (β to 0, resp.) allows only non-negative edges inside (leaving, resp.) S. By tuning the values of α and β, we define the degree of negative relations accepted, respectively, inside S and leaving S. For example, if $\alpha = 2$ the weighted sum of negative relations in S cannot exceed the double of its positive relations. These two bounds together lead to the definition of a set of mediators.

Definition 3 A subset $S \subseteq V$ is a set of mediators if S is $\alpha-$feasible and $\beta-$feasible.

We can now formally define the Correlation Clustering problem under Mediation.

Correlation Clustering problem under Mediation	
Input:	A signed graph $G = (V, E, s)$, non-negative arc weights $w \in \mathbb{R}_+^{\|E\|}$ and two scalars $\alpha, \beta \in \mathbb{R}_+$.
Output:	A partition $P = \{S_M, S_2, ..., S_{\|P\|}\}$ which minimizes the imbalance $I(P \backslash S_M)$ and such that S_M is a set of mediators.

The Correlation Clustering with Positive Mediation (CCPM) problem introduced in Doreian and Mrvar (2009) and formalized in Figueiredo and Moura (2013) is a specific case of CCM in which $\alpha = \beta = 0$.

We now prove that CCM is NP-hard.

Lemma 1 *The CCM problem is NP-hard.*

Proof. We prove this result with a reduction from CC. Consider an instance I_{CC} of CC defined over a signed graph $G = (V, E, s)$ with an edge weight vector $w \in \mathbb{R}_+^{|E|}$. Let $G' = (V', E', s')$ be a signed graph and let $w' \in \mathbb{R}_+^{|E'|}$ be an edge weight vector defined as follows (see Figure 1):

- $V' = V \cup \{n+1, n+2, n+3\}$
- $E' = E \cup E^1 \cup E^2 \cup E^3$ with:
  - $E^1 = \{(n+1, n+3), (n+2, n+3)\}$,
  - $E^2 = \{(n+1, n+2)\}$,
  - $E^3 = \{(n+2, i) : i \in V\} \cup \{(n+3, i) : i \in V\}$.
- $s'_e = \begin{cases} s_e, & e \in E, \\ +, & e \in E^1, \\ -, & e \in E^2 \cup E^3. \end{cases}$
- $w'_e = \begin{cases} w_e, & e \in E, \\ M, & e \in E^1 \cup E^2, \text{ with } M = 1 + \sum_{e \in E} w_e, \\ -3M, & e \in E^3. \end{cases}$

Consider the instance I_{CCM} of CCM defined over the signed graph G' with $\alpha = \beta = 1$. Let P_{CCM} be an optimal solution of I_{CCM}. We prove that P_{CCM} is necessarily equal to $S = \{\{n+1\}, \{n+2, n+3\}, P_{CC}\}$ where P_{CC} is an optimal solution of I_{CC}. We first observe that S is a feasible partition for instance I_{CCM}: the unitary set $\{n+1\}$ satisfy the conditions of a set of mediators for $\beta = 1$ and any $\alpha \in \mathbb{R}_+$. Moreover, the imbalance $I(\{\{n+2, n+3\}, P_{CC}\}) = I(P_{CC})$ is lower than M for any partition P_{CC} of the set of vertices $V \setminus \{n+1, n+2, n+3\}$. Next, we argue that the set of mediators in an optimal solution of I_{CCM} is necessarily $\{n+1\}$. Vertices $n+1, n+2$ and $n+3$ define a non-balanced cycle in G' (i.e., a cycle with an odd number of negative edges) composed of edges of weight M. As a consequence at least one of them must be in the set of mediators in an optimal solution (otherwise the imbalance would be at least M). If vertex $n+2$ or $n+3$ is in the set of mediators, a vertex in V cannot be neither in the set of mediators – as it would be α-infeasible – nor outside of the set of mediators – as it would be β-infeasible. As a consequence, vertex $n+1$ is necessarily in the set of mediators of an optimal solution. Moreover, no vertex in V can be in the set of mediators as it would be β-infeasible.

We can also conclude that $\{n+2, n+3\}$ forms necessarily a cluster in an optimal partition. Vertices $n+2$ and $n+3$ have to be together in a cluster, otherwise the imbalance would be greater than or equal to M. Moreover, no vertex in V can join this cluster, otherwise it will increase the imbalance of $6M$.

Finally, since P_{CC} is a partition of $V \setminus \{n+1, n+2, n+3\}$ and $I(\{\{n+2, n+3\}, P_{CC}\})$ is equal to $I(P_{CC})$, we can conclude that P_{CC} is an optimal partition for I_{CC}.

□

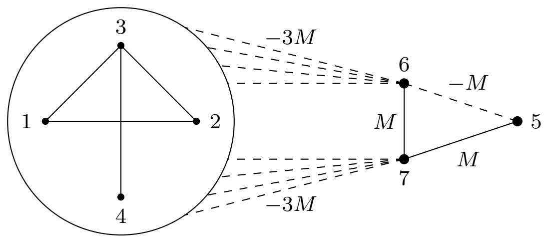


Fig. 1: Example of the reduction from an instance of CC with 4 vertices to an instance of CCM with 7 vertices.

In the next section, we formulate the CCM Problem as an Integer Linear Programming (ILP) model.

4 Mathematical formulation

ILP formulations have been successfully used in the literature for the resolution of clustering problems (Johnson et al., 1993; Mehrotra and Trick, 1996; Hansen and Jaumard, 1997; Agarwal and Kempe, 2008; Brusco and Steinley, 2009; Ales et al., 2016), including clustering problems defined on signed graphs (Figueiredo and Moura, 2013; Aref and Wilson, 2019). In this section, we introduce an ILP formulation for the CCM problem.

For each pair of distinct vertices i, j in V, we consider a binary variable x_{ij} equal to 1 if and only if i and j do not belong to the same cluster. Also, to each vertex $i \in V$ is associated a binary variable m_i equal to 1 if and only if i is a mediator. Note that in this formulation, each mediator vertex is represented as an isolated vertex. Finally, each pair of distinct vertices i, j is associated with two additional binary variables: t_{ij} equal to 1 if and only if both i and j are mediators; and z_{ij} equal to 1 if and only if at least i or j is a mediator.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{(i,j)\in E^-} w_{ij}(1-x_{ij}) + \sum_{(i,j)\in E^+} w_{ij}(x_{ij}-z_{ij}) && && (2)\\
\text{s.t.} \quad & x_{jk} \le x_{ij} + x_{ik}, && i \in V \; j,k \in V\backslash\{i\} \; j<k, && (3)\\
& m_i \le x_{ij}, && i,j \in V \; i \ne j, && (4)\\
& m_i + m_j - 1 \le t_{ij}, && i,j \in V \; i \ne j, && (5)\\
& t_{ij} \le m_i, && i,j \in V \; i \ne j, && (6)\\
& m_i \le z_{ij}, && i,j \in V \; i \ne j, && (7)\\
& z_{ij} \le m_i + m_j, && i,j \in V \; i \ne j, && (8)\\
& \sum_{(i,j)\in E^-} w_{ij}t_{ij} \le \alpha \sum_{(i,j)\in E^+} w_{ij}t_{ij}, && && (9)\\
& \sum_{(i,j)\in E^-} w_{ij}(z_{ij}-t_{ij}) \le \beta \sum_{(i,j)\in E^+} w_{ij}(z_{ij}-t_{ij}), && && (10)\\
& x_{ij} = x_{ji} \in \{0,1\}, && i,j \in V \; i \ne j, && (11)\\
& z_{ij} = z_{ji} \in [0,1], && i,j \in V \; i \ne j, && (12)\\
& t_{ij} = t_{ji} \in [0,1], && i,j \in V \; i \ne j, && (13)\\
& m_i \in \{0,1\}, && i \in V. && (14)
\end{aligned}
$$

The triangle inequalities (3) ensure that if i is in the same cluster as j and k ($x_{ij} = x_{ik} = 0$), then vertices j and k are also in the same cluster ($x_{jk} = 0$). Constraints (4) establish that mediators are isolated. Constraints (5) and (6) ensure that $t_{ij} = m_i m_j$. Constraints (7) and (8) impose, respectively, $z_{ij} = 1$ whenever $m_i + m_j \ge 1$ and $z_{ij} = 0$ otherwise. Constraints (9) and (10) ensure that the set of mediators is α and β-feasible, respectively. Remark that the expression $z_{ij} - t_{ij}$ is equal to 0 if and only if $m_i = m_j$. Consequently, for $\sigma \in \{-,+\}$, $\sum_{(i,j)\in E^\sigma} w_{ij}(z_{ij} - t_{ij}) = w^\sigma(\{m_i\}_{i\in V}, V\backslash\{m_i\}_{i\in V})$. Finally, the objective function (2) minimizes the imbalance defined by (1). The first term penalizes negative edges (i,j) connecting vertices in a same cluster (i.e., such that $x_{ij} = 0$) and the second term penalizes positive edges (i,j) connecting non-mediator vertices in different clusters (i.e, such that $x_{ij} = 1$ and $z_{ij} = 0$).

In Section 6 the performance of this formulation is compared with the ones of two enumeration algorithms presented in the next section.

5 Enumeration algorithms

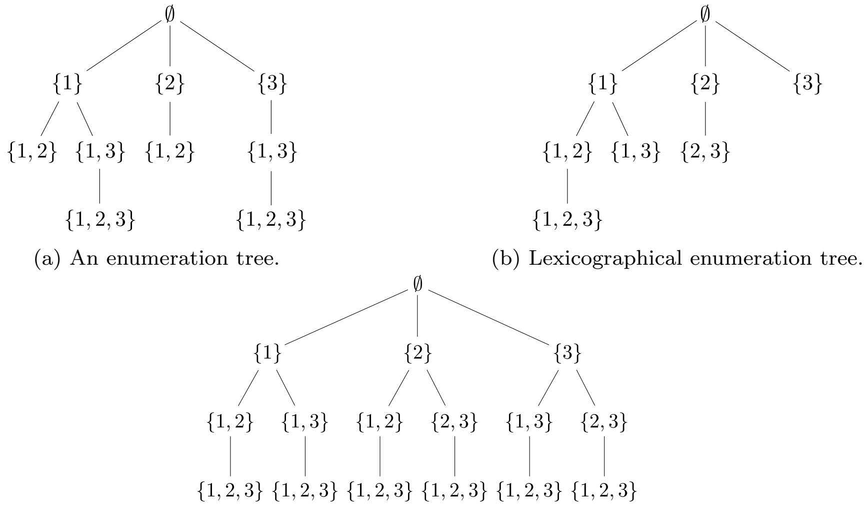
In this section, we present an alternative to the ILP based branch-and-bound algorithm, called *enumeration algorithms* for the optimal resolution of CCM. We first formally define the notion of enumeration algorithm (Section 5.1). Then, we study three simple enumeration strategies (called *policies*) and show that only one of them ensures an exact resolution (Section 5.2). Finally, based on this policy, we propose two enumeration algorithms called A_1 and A_2 (Sections 5.3 and 5.4). The first one generates one solution for each possible maximal set of mediators while A_2 focuses on returning a single optimal solution and efficiently prune branches of the exploration tree.

5.1 Enumeration tree and branching policy

Let an *enumeration tree* of a signed graph $G = (V, E, s)$ be a tree in which:

- each tree node is associated to a subset of V;
- the root corresponds to the empty set;
- each other node is associated to the set of its parent plus a new vertex.

Three enumeration trees are depicted in Figure 2.



(a) An enumeration tree. (b) Lexicographical enumeration tree. (c) Complete enumeration tree.

Fig. 2: Three enumeration trees for $|V| = 3$.

An *enumeration algorithm* for CCM generates an enumeration tree in order to identify sets of mediators of G. Solutions of the problem are then obtained by evaluating all mediators sets identified. The evaluation of a set S_M consists in finding the lowest possible imbalance of a solution in which S_M is the set of mediators. This is obtained by solving the CC problem instance associated with the signed graph induced by $V \backslash S_M$.

Let $\mathcal{P}(V)$ be the power set of V. One of the main components of an enumeration algorithm is its *branching policy* $\pi : \mathcal{P}(V) \times V \mapsto \{true, false\}$ which indicates when a node should be created or not in the enumeration tree. More specifically, if S is a subset of V and i is a vertex in $V \backslash S$ then $\pi(S, i)$

returns $true$ if node $S \cup \{i\}$ must be created as a child of node S and $false$ otherwise. As a consequence, the size of the tree generated by an algorithm directly depends on its policy. If the branching policy always returns true ($\pi(S,i) = true\ ,\ \forall S \in \mathcal{P}(V)\ ,\ \forall i \in V\backslash S$), a complete tree of $\mathcal{O}(n!)$ nodes is created (see Figure 2c). Enumerating the sets in lexicographical order corresponds to the branching policy $\pi(S,i) =$ "$i > argmax_{s\in S}\ s$" (see example in Figure 2b). This policy leads to a smaller tree size by avoiding any repetition (i.e., each set is associated to no more than one node). However, the size of the corresponding tree ($2^{|V|}$) remains prohibitive and better alternatives are required to efficiently solve CCM.

5.2 Simple branching policies

Let $< G, \alpha, \beta >$ be an instance of CCM defined by a signed graph $G = (V, E, s)$ and scalar values α and β. A branching policy π is said to be exact for $< G, \alpha, \beta >$ if the enumeration algorithm using π enumerates all sets of mediators in G.

We first study three branching policies called $\pi_{\alpha\beta}$, π_α and π_β and show that only π_α is exact. Policy $\pi_{\alpha\beta}$ is an intuitive branching policy which generates a node only if it corresponds to a set of mediators: $\pi_{\alpha\beta}(S,i) =$ "$S \cup \{i\}$ is a set of mediators". Policies π_α and π_β are less restrictive and, thus, lead to larger enumeration trees:

- $\pi_\alpha(S,i) =$ "$S \cup \{i\}$ is α-feasible";
- $\pi_\beta(S,i) =$ "$S \cup \{i\}$ is β-feasible".

To determine the conditions under which each of these three policies are exact, we consider the following definition.

Definition 4 (Björner and Ziegler (1992)) Let $\mathcal{F} \subseteq \mathcal{P}(S)$ be a family of subsets of a set S. The tuple $(S, \mathcal{F})$ is an *accessible system* if and only if:

(i) $\emptyset \in \mathcal{F}$,
(ii) if $X \in \mathcal{F}$ and $X \neq \emptyset$ then $\exists\ x \in X$ such that $X\backslash\{x\} \in \mathcal{F}$.

Let $\mathcal{M}$ be the family of all sets of mediators of the signed graph $G = (V, E, s)$. Similarly, let $\mathcal{A}$ and $\mathcal{B}$ be the family of all α-feasible and β-feasible sets of G, respectively. The three following lemmas prove that branching policies $\pi_{\alpha\beta}$, π_α and π_β are exact when $(V, \mathcal{M})$, $(V, \mathcal{A})$ and $(V, \mathcal{B})$ are accessible systems.

Lemma 2 *$\pi_{\alpha\beta}$ is exact for $< G, \alpha, \beta >$ if and only if $(V, \mathcal{M})$ is an accessible system.*

Proof. Let S be any set of mediators in G. If $(V, \mathcal{M})$ is an accessible system, there exists an ordering $(s_1, s_2, ..., s_{|S|})$ of the vertices in S such that $S\backslash\{s_1, s_2, ..., s_i\}$ is a set of mediators for all $i \in \{1, 2, ..., |S|\}$. As a consequence, S can be reached by $\pi_{\alpha\beta}$ through the following branch: $\emptyset$, $\{s_{|S|}\}$, $\{s_{|S|}, s_{|S|-1}\}$, ..., S.

We now prove that if $\pi_{\alpha\beta}$ is exact for $< G, \alpha, \beta >$, then $(V, \mathcal{M})$ is an accessible system. We use the contrapositive of this proposition, i.e. we assume $(V, \mathcal{M})$ is not an accessible system and we will see that there exists a set of mediators S which is not enumerated by $\pi_{\alpha\beta}$. Indeed, if $(V, \mathcal{M})$ is not an accessible system, that means there exists a set of mediators S such that $S \setminus \{s\}$ is not a set of mediators for each $s \in S$. Hence, by the definition of $\pi_{\alpha\beta}$, no set $S \setminus \{s\}$ will be enumerated by the branching policy $\pi_{\alpha\beta}$. Since the set S can only be generated from a set of the form $S \setminus \{s\}$, we can conclude that S will not be reached by $\pi_{\alpha\beta}$.

□

The two following lemmas provide weaker results for $(V, \mathcal{A})$ and $(V, \mathcal{B})$ which give sufficient conditions under which π_α and π_β are exact. The proof of these lemmas are omitted since they are similar to the first part of the proof of Lemma 2.

Lemma 3 *If $(V, \mathcal{A})$ is an accessible system, then π_α is exact for $< G, \alpha, \beta >$.*

Lemma 4 *If $(V, \mathcal{B})$ is an accessible system, then π_β is exact for $< G, \alpha, \beta >$.*

As we will prove in Lemma 10, $(V, \mathcal{A})$ is always an accessible system which ensures that π_α is always exact. Lemma 11 will prove that the same does not apply to π_β.

Note that, as defined next, a matroid is a special case of an accessible system.

Definition 5 (Whitney (1935)) Let $\mathcal{F} \subseteq \mathcal{P}(S)$ be a family of subsets of a finite set S. The tuple $(S, \mathcal{F})$ is a *matroid* if it satisfies the three following axioms:

(i) $\emptyset \in \mathcal{F}$;
(ii) *Hereditary axiom*: if $X \in \mathcal{F}$, then for all $Y \subseteq X$, $Y \in \mathcal{F}$;
(iii) *Augmentation axiom:* if $I, J \in \mathcal{F}$ and $|I| = |J|+1$, then there exists $x \in I \setminus J$ such that $J \cup \{x\} \in \mathcal{F}$.

We characterize in the remaining of this section when $(V, \mathcal{M})$, $(V, \mathcal{A})$ and $(V, \mathcal{B})$ are accessible systems or even matroids. These results are summarized in Table 1.

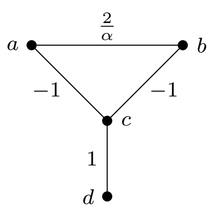
Unfortunately, $\pi_{\alpha\beta}$, which may provide smaller enumeration trees than π_α and π_β, is not exact in the general case.

Lemma 5 *If $\alpha \neq 0$, then $(V, \mathcal{M})$ is not necessarily an accessible system.*

Proof. In the graph represented in Figure 3, $\{a, b, c\}$ is a set of mediators but none of the subsets $\{a, b\}$, $\{a, c\}$ and $\{b, c\}$ is. □

Tuple	$\alpha > 0$, $\beta \geqslant 0$	$\alpha = 0$, $\beta > 0$	$\alpha = 0$, $\beta = 0$
$(V, \mathcal{M})$	$\times$ (Lemma 5)	Accessible (Lemma 8)	Matroid (Lemma 9)
$(V, \mathcal{A})$	Accessible (Lemma 10)		
$(V, \mathcal{B})$	$\times$ (Lemma 11)		

Table 1: Properties satisfied by $(V, \mathcal{M})$, $(V, \mathcal{A})$ and $(V, \mathcal{B})$. The symbol '$\times$' is used when the corresponding tuple is not an accessible system for all graphs.



(a) A signed graph for which $\{a, b, c\}$ is a set of mediators.

Set	α-**feasible**?	β-**feasible**?
$\{a, b, c\}$	yes $2 \leq \frac{2}{\alpha}\alpha$	yes $0 \leq \beta$
$\{a, b\}$	yes $0 \leq \frac{2}{\alpha}\alpha$	no $2 > 0\beta$
$\{a, c\}$ or $\{b, c\}$	no $1 > 0\alpha$	if $\beta \geq \alpha$ $1 \leq (\frac{2}{\alpha} + 1)\beta$

(b) Table which shows that for each $i \in \{a, b, c\}$, $\{a, b, c\} \setminus \{i\}$ is not a set of mediators.

Fig. 3: Example which shows that $(V, \mathcal{M})$ is not an accessible system when $\alpha \neq 0$.

Consequently, whenever $\alpha \neq 0$, an enumeration algorithm based on $\pi_{\alpha\beta}$ may not reach all the sets of mediators. The next lemma shows that this could even lead to sub-optimal solutions of the CCM problem.

Lemma 6 *Policy $\pi_{\alpha\beta}$ may not enumerate any of the sets of mediators leading to an optimal imbalance.*

Proof. Let $G = (V, E, s)$ be the signed graph represented in Figure 4 and let $\alpha = \beta = 1$. We can easily verify that for all $v \in \{c, d, e, f, g, h\}$ the following sets are not sets of mediators: $\{a, b\}$, $\{v\}$, $\{a, v\}$, and $\{b, v\}$. Consequently, the enumeration tree has only three nodes: $\emptyset$, $\{a\}$ and $\{b\}$. These three vertices can only provide solutions with an imbalance greater than 1 due to the non-balanced cycle $\{d, e, f\}$ in the graph. However, the mediators set $S = \{a, b, c, d\}$, which is not reached by the tree, leads to an optimal solution of cost 0 since the partition $\{S, \{e, f\}, \{g\}, \{h\}\}$ is balanced.

□

To prove that $\pi_{\alpha\beta}$ is exact when $\alpha = 0$, we first consider the following lemma.

Lemma 7 *Assume that $\alpha \leqslant \beta$. If S is a set of mediators, then there exists a vertex $s \in S$ such that $S \setminus \{s\}$ is β-feasible.*

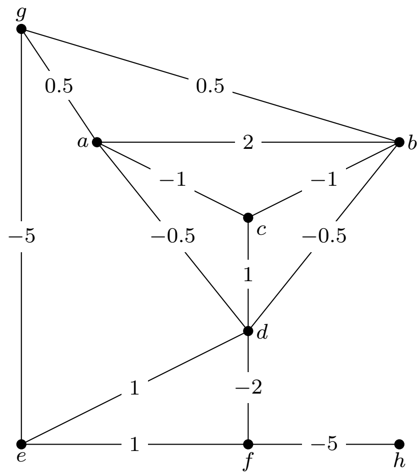


Fig. 4: A signed graph G where $(V, \mathcal{M})$ is not an accessible system. Branching policy $\pi_{\alpha\beta}$ applied to $< G, 1, 1 >$ does not enumerate any set of mediators associated with an optimal solution of $< G, 1, 1 >$.

Proof. Let us assume that for each $s \in S$, $S \setminus \{s\}$ is not β-feasible. Hence, the following inequality holds for all $s \in S$

$$\beta w^+(S \setminus \{s\}, V \setminus \{S \setminus \{s\}\}) < w^-(S \setminus \{s\}, V \setminus \{S \setminus \{s\}\})$$

equivalently

$$\beta w^+(S \setminus \{s\}, V \setminus S) + \beta w^+(s, S) < w^-(S \setminus \{s\}, V \setminus S) + w^-(s, S).$$

By summing up this inequality for each $s \in S$ we obtain

$$(|S|-1)\beta w^+(S, V \setminus S) + \beta \underbrace{\sum_{s \in S} w^+(s, S)}_{=2w^+(S)} < (|S|-1) w^-(S, V \setminus S) + \underbrace{\sum_{s \in S} w^-(s, S)}_{=2w^-(S)}.$$

Since S is a set of mediators, it is β-feasible. Consequently, $(|S|-1)\beta w^+(S, V \setminus S) \geqslant (|S|-1) w^-(S, V \setminus S)$, which together with the previous inequality leads to

$$\beta w^+(S) < w^-(S).$$

Assuming $\alpha \leqslant \beta$, this last inequality contradicts the α-feasibility of S. □

We now prove that $(V, \mathcal{M})$ is an accessible system when $\alpha = 0$.

Lemma 8 *If $\alpha = 0$, then $(V, \mathcal{M})$ is an accessible system.*

Proof. If $\alpha = 0$, the weight of each edge in a set of mediators S_M is non-negative. Hence, any subset of S_M is α-feasible. We deduce from Lemma 7 that there exists at least one vertex $s \in S_M$ such that $S_M \setminus \{s\}$ is additionally β-feasible. □

Note, that when $\alpha = \beta = 0$, $(V, \mathcal{M})$ is not only an accessible system but also a *matroid.*

Lemma 9 *If* $\alpha = \beta = 0$*, then* $(V, \mathcal{M})$ *is a matroid.*

Proof. Since $\alpha = \beta = 0$ the weight of each edge in S_M and between S_M and $V \setminus S_M$ is necessarily non-negative. This also applies to any subset of S_M and implies hereditary and augmentation axioms of a matroid. □

Lemma 9 ensures that, when both α and β are null, $\pi_{\alpha\beta}$ is exact. However, in this case, an enumeration algorithm based on this policy is not the best approach to solve CCM. Indeed, when $\alpha = \beta = 0$, an optimal solution of CCM can be obtained by identifying the unique maximal set of mediators S_M and solving CC on the remaining vertices $V \setminus S_M$ (Figueiredo and Moura, 2013). Such a set S_M can easily be identified as it contains all the vertices with adjacent edges with only non-negative weights.

Since $\pi_{\alpha\beta}$ is not exact for all signed graphs, we now focus on π_α and π_β. The two next lemmas show that only π_α is exact.

Lemma 10 *For any* $\alpha \geq 0$*,* $(V, \mathcal{A})$ *is an accessible system.*

Proof. Consider a α-feasible set S. Let us assume that, for each vertex $s \in S$, $S \setminus \{s\}$ is not α-feasible:

$$\alpha w^+ \left(S \setminus \{s\}\right) < w^- \left(S \setminus \{s\}\right) \quad \forall s \in S. \tag{15}$$

Summing up these inequalities for each $s \in S$, we obtain

$$(|S| - 2)\alpha w^+(S) < (|S| - 2)w^-(S), \tag{16}$$

since each edge (i, j), with $i, j \in S$, appears in each inequality (15) except when s is equal to i or j.

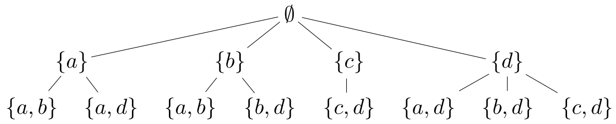
Equation (16) contradicts the α-feasibility of S. □

Lemma 11 *For any* $\beta \geqslant 0$*,* $(V, \mathcal{B})$ *is not necessarily an accessible system.*

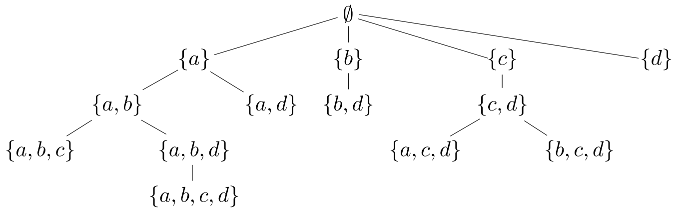
Proof. Consider a graph composed of two vertices linked by an edge of weight -1. The set $\{s, t\}$ is β-feasible while $\{s\}$ and $\{t\}$ are not.

□

As summarized in Table 1, π_β and $\pi_{\alpha\beta}$ are not exact in most of the cases and can, thus, lead to non-optimal solutions. Consequently, we base our two enumeration algorithms A_1 and A_2 on π_α.



(a) First two stages of the tree obtained with π_α.



(b) Tree obtained with π_{A_1}.

Fig. 5: Enumeration trees obtained for different policies for the graph presented in Figure 3a.

5.3 Algorithm A_1

In this section, we present our first enumeration algorithm A_1 for CCM and its branching policy π_{A_1}.

The two time consuming steps of an enumeration algorithm for CCM are the enumeration and the subsequent evaluation of the identified sets of mediators. We introduce in Section 5.3.1 an exact branching policy π_{A_1} which is a variation of π_α producing significantly smaller trees. Moreover, to speed up the evaluation step, we prove in Section 5.3.2 that only maximal sets of mediators need to be evaluated.

5.3.1 Branching policy π_{A1}

Lemmas (5) to (11) prove that π_α is exact while π_β and $\pi_{\alpha\beta}$ are not. Unfortunately, the enumeration tree generated by π_α may be huge (even larger than the lexicographical order policy) since π_α does not avoid repetitions (i.e., several nodes of the generated tree may correspond to the same set). This is illustrated by the enumeration tree in Figure 5a in which all α-feasible sets of size 2 are represented.

It would be tempting to combine π_α with the lexicographical policy and only enumerate in lexicographical order the sets which are α-feasible. However, this policy would not be exact. Indeed, in Figure 2b, if the set $\{1,2\}$ is not α-feasible, then the set $\{1,2,3\}$ can not be generated.

The following lemma enables to design an exact branching policy without node repetitions.

Lemma 12 *If $S \subset V$ is α-feasible and $v \in argmin_{i\in S}\ \alpha w^+(i,S) - w^-(i,S)$, then $S\backslash\{v\}$ is α-feasible.*

Proof. Lemma 10 ensures that there exists $k \in S$ such that $S \backslash \{k\}$ is α-feasible:

$$\alpha w^+(S) - w^-(S) - (\alpha w^+(k, S) - w^-(k, S)) \geq 0. \quad (17)$$

Let us assume that there exists a vertex $v \in argmin_{i \in S}\ \alpha w^+(i, S) - w^-(i, S)$ such that set $S \backslash \{v\}$ is not α-feasible:

$$\alpha w^+(S) - w^-(S) - (\alpha w^+(v, S) - w^-(v, S)) < 0. \quad (18)$$

However, from Equations (17) and (18) we arrive to

$$\alpha w^+(k, S) - w^-(k, S) < \alpha w^+(v, S) - w^-(v, S) \quad (19)$$

which contradicts the definition of v.

□

Let S be an α-feasible set. Lemma 12 ensures that by successively removing from S a vertex which minimizes $\alpha w^+(i, S) - w^-(i, S)$ (i.e., a vertex of S which contribution to the α-feasibility of S is minimal), a serie of α-feasible sets is obtained. In other words, S can be reached by a branching policy which uses this condition.

We describe next the exact branching policy of the enumeration algorithm A_1. Branching policy $\pi_{A_1}(S, i)$ returns true if and only if:

- $S' = S \cup \{i\}$ is α-feasible; and
- $i = \min\ argmin_{s \in S'}\ (\alpha w^+(s, S') - w^-(s, S'))$.

A minimization is used in the second condition to avoid repetitions in the enumeration tree whenever several vertices in S have a minimal contribution to the α-feasibility of S.

We now present how the evaluation step of an enumeration algorithm can be improved.

5.3.2 Evaluation of the generated sets of mediators

In order to solve the CCM problem, an enumeration algorithm must evaluate the sets of mediators it generates. The evaluation of a set of mediators S_M consists in solving the CC problem on the graph in which vertices S_M are removed. This step can be performed after the enumeration of all sets of mediators or in parallel, i.e., simultaneously with the enumeration process.

Since CC is NP-hard, reducing the number of evaluated sets can have a significant impact on the resolution time of an enumeration algorithm. The next lemma ensures that we can only evaluate maximal sets of mediators. For a given set $S \subseteq V$, let P^S be an optimal partition of the CC problem defined over $V \backslash S$.

Lemma 13 *Let S be a set of mediators and s a vertex in $V \backslash S$. We have that $I(P^S) \geq I(P^{S \cup \{s\}})$.*

Proof. Let $P^S = \{S_1, ..., S_k\}$ and assume without loss of generality that $s \in S_1$. According to Equation (1),

$$I(\{S_1 \setminus \{s\}, \ldots, S_k\}) \;=\; I(P^S) - w^-(\{s\}, S_1) - \sum_{2\leq j\leq k} w^+(\{s\}, S_j). \quad (20)$$

We can then conclude that

$$I(P^S) \geq I(\{S_1 \setminus \{s\}, \ldots, S_k\}) \geq I(P^{S\cup\{s\}}). \quad (21)$$

□

Lemma 13 implies that adding a vertex to the set of mediators can not deteriorate the optimal value of CCM.

Corollary 1 *Let $S, S' \subseteq V$ be two sets of mediators in G such that $S \subseteq S'$. Then $I(P^S) \geq I(P^{S'})$.*

Consequently, we only test maximal sets of mediators in our algorithms.

5.3.3 Pseudo-code of Algorithm A_1

To solve CCM, Algorithm A_1 generates all the maximal sets of mediators by calling the recursive function $A1Enumeration(G, \emptyset)$ (see Algorithm 1). It then returns a single set of mediators which minimizes the imbalance. Lines 2 and 3 of function $A1Enumeration$ enable to generate all the child nodes of node S which satisfy branching policy π_{A_1}. The sets of mediators are evaluated on Line 6 if no set of mediators is found in the subtree (i.e., if $L = \emptyset$). Note that this does not prevent A_1 from evaluating non maximal sets of mediators.

Algorithm 1: Recursive function $A1Enumeration$.

Data: $G = (V, E, s)$, a weighted signed undirected graph
$S \subset V$, a subset of vertices
Result: L, a list of sets of mediators $\{S_1, ..., S_N\}$ which include S and $\{I(P^{S_1}), ..., I(P^{S_N})\}$

1 $L \leftarrow \emptyset$
2 **for** $i \in V \backslash S$ **do**
3   **if** $\pi_{A_1}(S, i)$ **then**
4         $L \leftarrow L \cup A1Enumeration(G, S \cup \{i\})$
5 **if** $L = \emptyset$ *and S is β-feasible* **then**
6     $L \leftarrow \{(S, I(P^S))\}$
7 **return** L

Lemma 14 *Algorithm A_1 may evaluate non maximal sets of mediators.*

Fig. 6: (a) A graph and (b) its corresponding enumeration tree obtained with Algorithm A_1.

Proof. Figure 6b represents the enumeration tree obtained using policy π_{A_1} over the graph in Figure 6a.

Since $\{b\}$ is a mediators set and a leaf of the tree, it will necessarily be evaluated during Algorithm A_1. However, it is not a maximal mediator set as it is included in $\{a, b\}$. □

Algorithm A_1 enumerates exhaustively the maximal sets of mediators which could be particularly relevant in the context of decision aid applications, where alternative solutions are preferable (Arinik et al., 2021). We now define a second exact enumeration algorithm called A_2 which only returns a single optimal solution but which leverage linear relaxations to significantly reduce the size of its enumeration tree.

5.4 Algorithm A_2

Algorithm A_2 is based on the recursive function $A2Enumeration$, represented in Algorithm 2, which enables to reduce the size of the enumeration tree compared to $A1Enumeration$. This function takes as an input an upper bound UB which corresponds to the imbalance of a known feasible solution of the CCM problem. At each node S, it computes the value v_r of the linear relaxation of CCM in which the vertices in S are imposed to be included in the set of mediators (Line 2). If v_r is greater than UB, this sub-tree can not lead to a better solution and it is pruned. Finally, UB is updated whenever a better integer solution is obtained (Line 10).

Algorithm 2: Recursive function $A2Enumeration$

```
   Data: G = (V, E, s), a weighted signed undirected graph
         S ⊂ V, a subset of vertices
         UB, the best known upper bound of CCM (global variable)
   Result: L, a list of sets of mediators {S_1, ..., S_N} which include S and
           {I(P^{S_1}), ..., I(P^{S_N})}
1  L ← ∅
2  v_r ← optimal value of the linear relaxation of the CCM problem in
     which S is forced to be included in the set of mediators
3  if v_r < UB then
4      for i ∈ V\S do
5          if π_{A_1}(S, i) then
6              L ← L ∪ A2Enumeration(G, S ∪ {i})
7      if L = ∅ and S is β-feasible then
8          v* ← I(P^{V\S})
9          L ← {(S, v*)}
10         UB = min(UB, v*)
11 return L
```

To provide an initial upper bound, we use the greedy heuristic described in Algorithm 3. This heuristic tries to find a list of sets of mediators L such that each vertex in V appears in at least one of them. For this purpose the list $notInASet$ initially contains all the vertices (Line 2) and each time a vertex is added to a mediator set, it is removed from this list (Line 6 and 10). Each pass of the while loop Line 3 tries to create a set of mediators S_M starting with a candidate vertex from $notInASet$ (Line 4 and 5). Vertices are then added to S_M by successively selecting vertices which improve the most the α and the β-feasibilities of S_M (Line 7 and 11). Prior to adding S_M to L, we test if S_M is a set of mediators (Line 12). Note that if the candidate vertex is not included in any set of mediators of size 2, S_M can not be a set of mediators.

In that case, the greedy algorithm may not return any set of mediators which includes this vertex.

Algorithm 3: Greedy heuristic for the CCM problem H_G.

Data: $G = (V, E, s)$, a weighted signed undirected graph
Result: L, a list of sets of mediators

$L \leftarrow \emptyset$
$notInASet \leftarrow V$ // List of vertices which does not appear in any set of mediators found
while $notInASet \neq \emptyset$ **do**
  $candidate \leftarrow notInASet[1]$
  $S_M \leftarrow \{candidate\}$
  $notInASet \leftarrow notInASet \backslash \{candidate\}$
  $v \leftarrow argmax_{i \in V \backslash S_M} \min(\alpha w^+(i, S_M) - w^-(i, S_M), \beta w^+(i, V \backslash S_M) - w^-(i, V \backslash S_M))$
  while $S_M \cup \{v\}$ *is a set of mediators* **do**
    $S_M \leftarrow S_M \cup \{v\}$
    $notInASet \leftarrow notInASet \backslash \{v\}$
    $v \leftarrow argmax_{i \in V \backslash S_M} \min(\alpha w^+(i, S_M) - w^-(i, S_M), \beta w^+(i, V \backslash S_M) - w^-(i, V \backslash S_M))$
  if S_M *is a set of mediators* **then**
    $L \leftarrow L \cup S_M$
return L

Algorithm A_2 starts by calling the greedy heuristic. Each maximal set of mediators returned is then evaluated and the best imbalance obtained constitutes the initial upper bound UB. The exact enumeration is then performed by calling $A2Enumeration(G, \emptyset, UB)$.

5.5 Implementation improvements

To improve the efficiency of A_1 and A_2, several implementation choices have been made.

At each node, the α and the β-feasibility are not computed from scratch. They are instead deduced from the values obtained at the parent node. For example, let us consider a node $S \cup \{i\}$ son of node S. At node $S \cup \{i\}$, the α-feasibility of node S has already been tested. The value $\alpha w^+(S) - w^-(S)$ is thus known. We leverage this value to test the α-feasibility of node $S \cup \{i\}$ thanks to the equation:

$$\alpha w^+(S \cup \{i\}) - w^-(S \cup \{i\}) = \alpha w^+(S) - w^-(S) + \alpha w^+(i, S) - w^-(i, S). \quad (22)$$

Consequently, at each node $S \cup \{i\}$, we only compute the value $\alpha w^+(i, S) - w^-(i, S)$. A similar reasoning is considered for the β-feasibility tests.

Enumeration algorithms must both enumerate and evaluate sets of mediators. The evaluation of a set S requires to solve a NP-hard problem and we know that it is not necessary if S is not a maximal set of mediators. Consequently, it is not efficient to evaluate a set as soon as it is enumerated. An alternative would be to first enumerate all the sets of mediators and then evaluate the ones which are maximal. This approach has two drawbacks:

- when the resolution time is limited, enumerating all the sets of mediators may not leave enough time to evaluate all the sets of mediators, leading to a solution of poor quality. In hard instances it can even lead to no solution at all;
- in A_2 evaluating sets of mediators may enable to improve the upper bound UB, thus reducing the size of the enumeration tree. If the sets of mediators are evaluated after the enumeration, this bound can not be strengthened during the enumeration.

Consequently, our algorithms alternate between the enumeration and the evaluation steps until the algorithm or the time is over. More precisely, the first evaluation step starts when a quarter of the time limit has elapsed. At the end of an evaluation step, the remaining time is computed and the next evaluation step will occur when a quarter of that time has elapsed.

6 Computational experiments

We compare the performances of A_1, A_2 and the formulation presented in Section 4: in Section 6.1, on two datasets composed of random instances; in Section 6.2, on instances obtained from the vote of the members of the European parliament (Arinik et al., 2020) [1]. We use a 3.60GHz Intel(R) Xeon(R) Gold 6244 equipped with 384GByte of RAM. The linear programs are solved with CPLEX 12.10 and all algorithms are implemented in Julia v1.8.2.

For each instance I considered, let $\bar{\alpha}_I = \frac{\sum_{(i,j)\in E^-} w_{ij}}{\sum_{(i,j)\in E^+} w_{ij}}$. The solution in which V is a set of mediators is always optimal since it leads to an imbalance of 0. Consequently, the problem is trivial for any value $\alpha \geq \bar{\alpha}_I$ and $\bar{\alpha}_I$ is the lowest value for which V is a set of mediators. To evaluate our methods over non-trivial problems, we consider for each instance I the three following values of α: $0.25\,\bar{\alpha}_I$, $0.5\,\bar{\alpha}_I$ and $0.75\,\bar{\alpha}_I$.

6.1 Random dataset

We randomly generate instances with 30 to 50 vertices and with densities $\rho \in \{0.2, 0.5, 0.8\}$ by using the erdos.renyi.game function from R's "igraph" library (see (Csardi and Nepusz, 2006)). The density $\rho \in [0, 1]$ corresponds to the probability that an edge exists. The weight and sign of the edges are defined by uniformly generating values in $[-1, 1]$.

[1] the data are available at
`https://osf.io/nrmec/?view_only=041e08fbaa8444eba4473f5c105f7ca4`

6.1.1 Generating all maximal sets of mediators

Lemma 13 states that for any set $S' \subset S$, $I(P^{S'}) \geq I(P^S)$. Consequently, the maximal sets of mediators constitute particularly interesting solutions on which we focus in Algorithms A_1 and A_2.

In a decision aid process based on the CCM problem, generating a single solution, i.e. a single set of mediators, may not be suitable. For example, in the instances of the European parliament considered in Section 6.2, a set of mediators is used to constitute a commission on a given topic. However, in this context, a solution may be impractical due to additional constraints which could be related to the availability of the deputies constituting the set or the parity constraints between the countries represented. Consequently, the fact that Algorithm A_1 exhaustively generates all maximal sets of mediators and could leads to several diverse optimal solutions can be a significant advantage.

Solving our CCM formulation with CPLEX does not directly enable to generate all the maximal sets as it only returns one optimal solution of the problem at a time. To overcome this problem, we could use the method proposed in (Danna et al., 2007) (included in CPLEX) to generate all the optimal solutions of an ILP formulation in a single branch-and-bound tree. However, this approach is likely to enumerate non-relevant solutions. Indeed, two different optimal solutions of CCM problem can be associated to a same set of mediators. Moreover, non-maximal set of mediators can also lead to optimal solutions.

Consequently, we implemented an alternative method in which CPLEX is executed iteratively. Let $\mathcal{S} = \{S_1, ..., S_i\}$ be the sets of mediators obtained at the i first iterations. To ensure that the set obtained at iteration $i + 1$ is not included in $\mathcal{S}$, we add the following constraints to the model

$$\sum_{i \notin S} m_i \geq 1 \qquad \forall S \in \mathcal{S}. \tag{23}$$

For each set S, Constraints (23) ensure that all sets of mediators subsequently generated contain at least one vertex in $V \backslash S$. The iterative process stops once no solution is returned by CPLEX. Eventually, the sets of $\mathcal{S}$ which are not maximal are removed from it.

We now compare this iterative process with A_1. Table 2 presents the solution time and the number of maximal sets of mediators generated by each approach. The two first columns of Table 2 represent the size and density of the graphs. The next column contains the percentage of $\bar{\alpha}_I$ considered. Each value corresponds to an average over the five random instances generated. A_1 appears to be significantly better at this task as in 24 cases over 27 it either returns more maximal sets of mediators or the same number but in less time. Note that, unlike A_1, CPLEX is not able to return any solution for the largest instances.

$\|V\|$	ρ	$\overline{\alpha}_I$%	**CPLEX**		A_1	
			Time	# sets	Time	# sets
30	0.2	0.25	924s	33	**39s**	33
		0.5	TL	317	**323s**	2677
		0.75	TL	1641	**2663s**	39649
30	0.5	0.25	33s	1	**20s**	1
		0.5	3440s	31	**97s**	44
		0.75	TL	33	**921s**	11708
30	0.8	0.25	69s	1	**66s**	1
		0.5	534s	3	**82s**	3
		0.75	TL	10	**793s**	3727
40	0.2	0.25	**600s**	7	5932s	7
		0.5	TL	60	TL	**1701**
		0.75	TL	1231	TL	**73824**
40	0.5	0.25	2634s	1	**1967s**	1
		0.5	TL	2	TL	**45**
		0.75	TL	4	TL	**28505**
40	0.8	0.25	**5921s**	1	6613s	1
		0.5	TL	0	TL	**3**
		0.75	TL	0	TL	**4496**
50	0.2	0.25	**4870s**	7	TL	7
		0.5	TL	5	TL	**1768**
		0.75	TL	861	TL	**210302**
50	0.5	0.25	TL	0	TL	**1**
		0.5	TL	0	TL	**1**
		0.75	TL	0	TL	**14568**
50	0.8	0.25	TL	0	TL	**1**
		0.5	TL	0	TL	**1**
		0.75	TL	0	TL	**1087**

Table 2: Mean time and number of maximal sets of mediators found for CPLEX and A_1 over the random graphs. Each value is an average over the five instances. On each line, the best result is in bold. TL indicates that the time limit of 7200s has been reached in all five instances.

6.1.2 Generating a single optimal solution

We now focus on generating a single optimal solution. In this context CPLEX does not solve our MIP formulation iteratively anymore but just once. Furthermore, Algorithm A_2, which returns an optimal solution and may prune branches leading to maximal sets of mediators, is now considered.

For a given instance, let x^I be the value of the best solution returned by a method and let x^{LB} be the lower bound it provides. We define the *relative gap* as $100 \times \frac{|x^I - x^{LB}|}{x^I}$. Since A_1 and A_2 do not provide a lower bound, the lower bound obtained with CPLEX is used to compute their relative gap.

$\|V\|$	ρ	$\bar{\alpha}_I\%$	A_1			A_2			**CPLEX**		
			Time	Gap	Nodes	Time	Gap	Nodes	Time	Gap	Nodes
30	0.2	0.25	39s	0%	1.3×10^7	90s	0%	9313	**6s**	0%	93
		0.5	323s	0%	8.8×10^7	2701s	0%	3.4×10^5	**5s**	0%	70
		0.75	2663s	1%	2.7×10^8	720s	0%	1.1×10^5	**3s**	0%	16
30	0.5	0.25	20s	0%	9.1×10^5	**10s**	0%	47	22s	0%	37
		0.5	97s	0%	2.0×10^7	294s	0%	9045	**65s**	0%	458
		0.75	921s	0%	1.4×10^8	TL	2%	3.3×10^5	**66s**	0%	1075
30	0.8	0.25	66s	0%	1.4×10^5	**46s**	0%	31	63s	0%	202
		0.5	82s	0%	5.9×10^6	**75s**	0%	343	185s	0%	1542
		0.75	793s	0%	1.2×10^8	TL	7%	1.8×10^5	**618s**	0%	6724
40	0.2	0.25	5932s	3%	1.3×10^9	**31s**	0%	659	55s	0%	43
		0.5	TL	2%	1.3×10^9	6666s	2%	3.5×10^5	**110s**	0%	831
		0.75	TL	1%	3.4×10^8	2880s	0%	1.4×10^5	**5s**	0%	4
40	0.5	0.25	1967s	0%	3.2×10^7	**666s**	0%	46	2478s	0%	4522
		0.5	TL	31%	1.1×10^9	**1807s**	0%	22220	3305s	0%	8778
		0.75	TL	6%	7.7×10^8	TL	6%	1.9×10^5	**2739s**	0%	15250
40	0.8	0.25	6613s	0%	2.3×10^6	**2889s**	0%	33	5874s	0%	8552
		0.5	TL	89%	3.5×10^8	**5598s**	53%	145	TL	-	56745
		0.75	TL	**22%**	1.1×10^9	TL	23%	46489	TL	-	31980
50	0.2	0.25	TL	19%	1.5×10^9	**261s**	0%	4650	373s	0%	272
		0.5	TL	9%	1.3×10^8	TL	5%	3.9×10^5	**1037s**	0%	2866
		0.75	TL	1%	4.4×10^8	4320s	1%	2.6×10^5	**6s**	0%	20
50	0.5	0.25	TL	-	4.1×10^8	TL	-	0	TL	-	3114
		0.5	TL	-	8.1×10^8	TL	**62%**	8063	TL	-	5431
		0.75	TL	26%	4.5×10^8	TL	**14%**	56615	TL	-	20718
50	0.8	0.25	TL	**0%**	3.6×10^7	TL	-	0	TL	-	2687
		0.5	TL	-	1.1×10^9	TL	-	0	TL	-	3164
		0.75	TL	59%	3.8×10^8	TL	**44%**	24947	TL	-	6810

Table 3: Mean time in seconds, relative gap and number of enumerated nodes obtained for each method over the random graphs. Each value is an average over five instances. On each line, the best result is in bold. A dash in a Gap column indicates that no solution is obtained for at least one of the instances. TL indicates that the time limit of 7200s has been reached in all five instances.

The execution time, the number of nodes generated and the relative gap of each method are presented in Table 3. Each entry of this table corresponds to a mean value over 5 instances. The time limit of each method is fixed to 2 hours.

The resolution of our formulation through CPLEX appears to provide the best results on most of the instances. Algorithm A_2 is often close to CPLEX and is even able to beat it in 10 cases over 27. CPLEX is known for the efficiency of its presolve algorithm which often enables to drastically reduce the size of a MILP and its fine-tuned heuristics which determine in particular on which variable to branch and which node to evaluate next. We posit that the efficiency of CPLEX over A_1 and A_2 is mainly due to these features which enable to optimally solve the problems with a significantly smaller number of nodes.

The differences in terms of resolution time and size of the enumerated trees between A_1 and A_2 highlight the efficiency of A_2 pruning mechanism.

We observe that the resolution times tend to increase with size of the graph, its density and $\bar{\alpha}_I$. This is not surprising as all these parameters are related to the complexity of the problem. The size of the graph determines the number of variables in the formulation and the number of branches to consider in the enumeration algorithms. The greater the density, the more complex the objective function. Finally, $\bar{\alpha}_I$ directly impacts the number of feasible solutions.

Most of the instances where A_2 beats CPLEX correspond to $0.25\bar{\alpha}_I$. This is due to the fact that the size of the maximal sets of mediators decreases when α decreases, thus reducing the depth of the branches of the enumeration algorithms.

6.2 European parliament dataset

We now consider real world instances obtained by Arinik et al. (2017) from votes casted during the 7^{th} term of the european parliament from 2009 to 2014. The roll-call votes of all members of the european parliament (MEP) for all plenary sessions in this period are available on the website *It's Your Parliament* (Buhl & Rasmussen (2020)).

In order to obtain challenging instances, we selected countries with more than 30 MEP and three of the most controversial policy domains: agriculture, gender equality and economic. For each country, one graph is generated for each domain. As described by Arinik et al. (2017), each MEP is associated to a vertex while the sign and weight of an edge represent the voting similarity between two MEPs.

The results obtained for this dataset are presented in Table 4. Each value in this table corresponds to an average over three instances (one for each policy domain considered). The table contains the values of the objective function instead of the gaps since CPLEX either returns the optimal solution or no solution at all which means that its gap is either 0% or not defined. The resolution time of CPLEX quickly increases with the size of the graphs and it is only able to provide feasible solutions for the three smallest instances. Algorithm A_2, however, is faster than CPLEX and always returns a solution. The efficiency of A_2 is partially due to its greedy heuristic which is very efficient on these real world instances. Indeed, it often returns a solution with no imbalance leading to an enumeration tree with only one node. This is not surprising as the instances are quite polarized along the lines of the political groups of the european parliament. However, the efficiency of A_2 is not only due to its

n	**Country**	$\bar{\alpha}_I\%$	**CPLEX**			$\mathbf{A_2}$		
			Time	Obj.	Nodes	Time	Obj.	Nodes
		0.25	**10s**	0	53	**0**s	**0**	1
33	Romania	0.5	**7**s	**0**	172	**0**s	**0**	1
		0.75	**7**s	**0**	172	**0**s	**0**	1
		0.25	391s	0	30	**0**s	**0**	1
51	Poland	0.5	1116s	0	658	**0**s	**0**	1
		0.75	149s	0	15	**0**s	**0**	1
		0.25	2390s	0	669	**0**s	**0**	1
59	Spain	0.5	2015s	0	76	**0**s	**0**	1
		0.75	614s	0	11	**0**s	**0**	1
		0.25	9977s	-	2	**9601**s	**1**	29388
72	UK	0.5	11006s	-	189	**9601**s	**1**	31937
		0.75	TL	-	440	**4803**s	**0**	9827
		0.25	TL	-	5	**9601**s	**4**	16682
87	France	0.5	TL	-	29	**4803**s	**2**	6610
		0.75	TL	-	19	**6**s	**0**	1
		0.25	TL	-	2	**9601s**	**0**	10215
104	Germany	0.5	TL	-	2	**7**s	**0**	1
		0.75	TL	-	2	**17**s	**0**	1

Table 4: Mean time in seconds, objective value and number of enumerated nodes obtained on the instances from the european parliament. Each value is an average over three instances. On each line, the best result is in bold and a dash is used in column Obj. if no solution is obtained for at least one of the instances. TL indicates that the time limit of 14400s has been reached in all three instances.

greedy heuristic as the enumeration algorithm enables to improve the greedy solution in most instances with several nodes.

We conclude this section by highlighting advantages of the enumeration algorithms over the integer programming formulation when solving the CCM problem. First, A_1 generates all the maximal sets of mediators. As mentioned before, in the context of decision aid systems, providing a variety of relevant solutions for the CCM problem is essential. As seen in Section 6.1.2, CPLEX would be significantly less efficient at this task. It can be tuned to generate a pool of solutions but it can not guarantee that all the maximal sets of mediators or even all the optimal solutions are obtained. Secondly, the enumeration algorithms can easily be adapted to new definitions of sets of mediators involving non-linear and non-convex constraints. The satisfaction of these constraints can be tested at the same time than the β-feasibility (Line 5 of Algorithm 1 and Line 7 of Algorithm 2).

7 Conclusions and perspectives

In this paper, we propose a new variant of the correlation clustering problem, called the correlation clustering problem with mediation, based on the work of Doreian and Mrvar (2009). After proving its NP-hardness we model it with an integer mathematical formulation. We also develop two enumeration algorithms A_1 and A_2 to solve optimally this problem and exhaustively enumerate all the maximal sets of mediators. These algorithms are based on properties of the sets of mediators which enable to efficiently prune branches of the enumeration tree. Finally, we compare experimentally the performances of the formulation and of the enumeration algorithms on a dataset with random instances and on a second with real world instances obtained from european parliament votes. The resolution of the formulation with CPLEX gives better results on hard random instances but, unlike A_2 it fails to provide feasible solutions on the real instances considered.

A natural perspective to this work would be to improve the pruning technique of the enumeration algorithms by identifying additional properties of the sets of mediators to strengthen the branching policies. A new type of enumeration algorithm could also be introduced in which vertices are removed rather than added at each new node of the enumeration tree. Such algorithm could cut a branch as soon as a set of mediators is reached. This approach could be particularly efficient when the maximal sets of mediators are large (i.e., for large values of parameters α and β). The present work contributes to the formalization of mediation in structural balance theory, introduced by Doreian and Mrvar (2009). A last perspective would be to consider alternative definitions of a set of mediators. The flexibility of the enumeration algorithms could allow the use of non-linear constraints. For some applications it could also be relevant to associate a label to each vertex (e.g., a political party) and to require that the proportion of each label in a set of mediators is representative of its distribution in the graph.

Acknowledgements Acknowledgements, if any, should follow the conclusions, and be placed above any Appendices or the references.

References

Abramowitz AI, Saunders KL (2008) Is polarization a myth? The Journal of Politics 70(2):542–555

Agarwal G, Kempe D (2008) Modularity-maximizing network communities using mathematical programming. Eur Phys J B 66:409–418

Ahmadian S, Epasto A, Kumar R, Mahdian M (2020) Fair correlation clustering. In: Chiappa S, Calandra R (eds) Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, PMLR, Online, Proceedings of Machine Learning Research, vol 108, pp 4195–4205

Ales Z, Knippel A, Pauchet A (2016) Polyhedral combinatorics of the k-partitioning problem with representative variables. Discrete Applied Mathematics 211:1–14

Aref S, Wilson MC (2019) Balance and frustration in signed networks. J Complex Networks 7(2):163–189

Arinik N, Figueiredo R, Labatut V (2017) Signed graph analysis for the interpretation of voting behavior. In: Proceedings of the International Conference on Knowledge Technologies and Data-Driven Business 2017 (i-Know 2017), Graz, Austria

Arinik N, Figueiredo R, Labatut V (2020) Multiple partitioning of multiplex signed networks: Application to european parliament votes. Social Networks 60:83–102

Arinik N, Figueiredo R, Labatut V (2021) Multiplicity and Diversity: Analyzing the Optimal Solution Space of the Correlation Clustering Problem on Complete Signed Graphs. Journal of Complex Networks 8

Arinik N, Figueiredo R, Labatut V (2023) Efficient Enumeration of the Optimal Solutions to the Correlation Clustering problem. To appear in Journal of Global Optimization

Bansal N, Blum A, Chawla S (2004) Correlation clustering. Machine Learning 56:89–113

Björner A, Ziegler GM (1992) Matroid applications, Cambridge University Press, chap 8, pp 284–357

Borgatti S (2003) The key player problem. In: Breiger R, Carley K, Pattison P (eds) Dynamic Social Network Modeling and Analysis, Workshop Summary & Papers, National Academy of Sciences Press, Washington-DC, pp 241–252

Borgatti SP (2006) Identifying sets of key players in a social network. Comput Math Organiz Theor 12(1):2134

Brusco M, Steinley D (2009) Integer programas for one- and two-model blockmodeling based on prespecified image matrices for structural and regular equivalence. Journal of Mathematical Psychology 53:577–585

Buhl & Rasmussen (2020) It's your parliament. `http://www.itsyourparliament.eu/`

Cartwright D, Harary F (1956) Structural balance: A generalization of heiders theory. Psychological Review 63:277–293

Csardi G, Nepusz T (2006) The igraph software package for complex network research. InterJournal Complex Systems:1695

Danna E, Fenelon M, Gu Z, Wunderling R (2007) Generating multiple solutions for mixed integer programming problems. In: International Conference on Integer Programming and Combinatorial Optimization, Springer, pp 280–294, DOI 10.1007/978-3-540-72792-7_22

DasGupta B, Enciso G, Sontag E, Zhang Y (2007) Algorithmic and complexity results for decompositions of biological networks into monotone subsystems. Biosystems 9(1):161–178

Davis JA (1967) Clustering and structural balance in graphs. Human Relations 20(2):181–187

Demaine ED, Emanuel D, Fiat A, Immorlica N (2006) Correlation clustering in general weighted graphs. Theoretical Computer Science 361:172–187

Doreian P, Mrvar A (1996) A partitioning approach to structural balance. Social Networks 18:149–168

Doreian P, Mrvar A (2009) Partitioning signed social networks. Social Networks 31:1–11

Figueiredo R, Frota Y (2014) The maximum balanced subgraph of a signed graph: Applications and solution approaches. European Journal of Operational Research 236(2):473 – 487

Figueiredo R, Moura G (2013) Mixed integer programming formulations for clustering problems related to structural balance. Social Networks 35(4):639–651

Figueiredo RMV, Labbé M, de Souza CC (2011) An exact approach to the problem of extracting an embedded network matrix. Computers & Operations Research 38(11):1483–1492

Gleich DF, Veldt N, Wirth A (2018) Correlation Clustering Generalized. In: Hsu WL, Lee DT, Liao CS (eds) 29th International Symposium on Algorithms and Computation (ISAAC 2018), Dagstuhl, Germany, Leibniz International Proceedings in Informatics (LIPIcs), vol 123, pp 44:1–44:13

Hansen P, Jaumard B (1997) Cluster analysis and mathematical programming. Mathematical Programming 79:191–215

Harary F (2002) Signed graphs for portfolio analysis in risk management. IMA Journal of Management Mathematics 13(3):201–210

Heider F (1946) Attitudes and cognitive organization. Journal of Psychology 21(1):107–112

Johnson EL, Mehrotra A, Nemhauser GL (1993) Min-cut clustering. Mathematical Programming 62:133–151

Kalhan S, Makarychev K, Zhou T (2019) Correlation clustering with local objectives. In: Wallach H, Larochelle H, Beygelzimer A, Alché-Buc F, Fox E, Garnett R (eds) Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 32, pp 9346–9355

Kolluri R, Shewchuk JR, O'Brien JF (2004) Spectral surface reconstruction from noisy point clouds. In: Eurographics/ACM SIGGRAPH Symposium on Geometry processing, pp 11–21

Kropivnik S, Mrvar A (1996) An analysis of the slovene parliamentary parties network. Metodoloki Zvezki / Advances in Methodology and Statistics 12:209–216

Levorato M, Figueiredo R, Frota Y, Drummond L (2017) Evaluating balancing on social networks through the efficient solution of Correlation Clustering problems. EURO Journal on Computational Optimization 5(4):467–498

Li CT, Lin SD, Shan MK (2011) Finding influential mediators in social networks. In: Proceedings of the 20th Int. Conf. Companion on World Wide Web, Association for Computing Machinery, New York, NY, USA, pp 75–76

Li P, Dau H, Puleo G, Milenkovic O (2017) Motif clustering and overlapping clustering for social network analysis. In: IEEE INFOCOM 2017 - IEEE Conference on Computer Communications, pp 1–9

Mehrotra A, Trick MA (1996) A column generation approach for graph coloring. INFORMS Journal of Computing 8:344–354

Mehrotra A, Trick MA (1998) Cliques and clustering: A combinatorial approach. Operations Research Letters 22(1):1 – 12

Ortiz-Arroyo D (2010) Discovering Sets of Key Players in Social Networks, Springer London, London, pp 27–47

Puleo GJ, Milenkovic O (2018) Correlation clustering and biclustering with locally bounded errors. IEEE Trans on Information Theory 64(6):4105–4119

Veldt N, Gleich D, Wirth A (2018) A correlation clustering framework for community detection. In: Proceedings of the 2018 World Wide Web Conference, Geneve, CHE, pp 439–448

Whitney H (1935) On the abstract properties of linear dependence american journal of mathematics. American Journal of Mathematics 57(3):509533, DOI https://doi.org/10.2307/2371182