



# Mixed integer programming formulations for clustering problems related to structural balance

Rosa Figueiredo, Gisele Moura

## ► To cite this version:

Rosa Figueiredo, Gisele Moura. Mixed integer programming formulations for clustering problems related to structural balance. *Social Networks*, 2013, 35 (4), pp.639-651. 10.1016/j.socnet.2013.09.002 . hal-02171111

**HAL Id: hal-02171111**

**<https://hal.science/hal-02171111>**

Submitted on 2 Jul 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Mixed Integer Programming Formulations for Clustering Problems related to Structural Balance

Rosa Figueiredo<sup>a,\*</sup>, Gisele Moura<sup>b</sup>

<sup>a</sup>*CIDMA, Department of Mathematics, University of Aveiro  
3810-193 Aveiro, Portugal.*

<sup>b</sup>*Department of Mathematics and Statistics, Rio de Janeiro State University  
20550-900 Rio de Janeiro, Brazil.*

---

## Abstract

In this work, we study graph clustering problems associated with structural balance. One of these problems is known in computer science literature as the correlation-clustering (CC) problem and another (RCC) can be viewed as its relaxed version. The solution of CC and RCC problems have been previously used in the literature as tools for the evaluation of structural balance in a social network. Our aim is to solve these problems to optimality. We describe integer linear programming formulations for these problems which includes the first mathematical formulation for the RCC problem. We also discuss alternative models for the relaxed structural balance and the solution of clustering problems associated with these new models. Numerical experiments are carried out with each formulation on a set of benchmark instances available in the literature.

*Keywords:* Structural balance; Signed graph; Graph partition; Integer programming; Social network.

---

---

\*Corresponding author.

Address: Universidade de Aveiro, Campus de Santiago, Departamento de Matemática,  
3810-193 Aveiro, Portugal.  
Fax number: +351 234370066.  
Email: rosa.figueiredo@ua.pt

## 1. Introduction

Signed digraphs were introduced by Heider in [29] with the purpose of describing sentiment relations between people pertaining to a same social group and to provide a systematic statement of social balance theory. Cartwright *et al.* [12] formalized Heider’s theory stating that a balanced social group, i.e., a balanced signed digraph, could be partitioned into two mutually antagonistic subgroups each having internal solidarity. Subsequently, noting that a social group may contain more than two hostile subgroups, Davis [16] extended the definition of a balanced signed digraph to the definition of a clusterable signed digraph. In the last decades, signed digraphs have shown to be a very attractive discrete structure for social network researchers [2, 3, 19, 20, 44]. One challenge in this area is to evaluate balance in a social network: the degree of balance in a social group can be used as a tool to study whether and how this group evolves to a possible balance state.

Different criteria can be used for that purpose. The optimal solution of clustering problems defined on signed digraphs have already been used as a criteria to measure the degree of balance in social networks [18, 19, 20]. In [19], Doreian *et al.* looked for a partition of the vertex set that minimizes the number of negative arcs within clusters, plus the number of positive arcs between clusters. In solving this problem, they aimed to find a vertex partition closest to a balanced state which could be used as a measure of imbalance in the signed digraph representing a social network. Information obtained with the computational experiments were used to verify theoretical assumptions and to suggest additional hypothesis about structural balance in social networks. Lately, the same authors observed in [20] that is highly possible that a network evolves to a state where the elements of two groups cooperate mutually or to a state where there is a group whose members are internally hostile. They argued that, in some contexts, these relations should not be considered as a contribution to the imbalance of the network. These authors extended the definition of structural balance on a signed digraph to what they called relaxed structural balance. The neighborhood search procedure introduced in [19] was adapted in order to deal with this new concept of social balancing and is available for scientific purposes use [1]. The measures of balance proposed by Doreian *et al.* were used in a balance theory approach proposed to group problem solving [3] and in a multiple indicator approach proposed in [18].

Clustering problems defined on signed digraphs also arise in the study of community structures [33, 42], a very prominent area of network science [39], as well as in other scientific areas, including portfolio analysis in risk management [21, 28], biological systems [15, 30], efficient document classification [6], detection of embedded matrix structures [23, 26], and analysis of similarity or dissimilarity in relationships [9]. The common element among these applications is that all of them are defined in a collaborative vs. conflicting environment that can be modeled over a signed digraph. Appearing in very different areas, we can expect that different notations, definitions and solution approaches are proposed in the literature for a same problem. As an example, the problem treated

by Doreian *et al.* in [19, 20] is called correlation clustering problem in [6], community mining in [44] and  $K$ -balance partitioning in [9].

From a practical point of view, in solving the clustering problems treated in this paper, heuristic approaches are primarily of interest since large social networks may have to be analyzed. However, an absolute evaluation of a heuristic approach is only possible if the optimal solution is known for a set of instances. Moreover, in some cases, the social networks used as benchmark instances have only tens of nodes [3, 19, 20] and can be solved to optimality by an exact solution approach. A deep investigation of efficient exact approaches and mathematical formulations to clustering problems related with structural balance is a missing point. Recently, this gap was filled by the works [9] and [7] where branch-and-bound procedures were presented, respectively, for the problems studied in [20] and [19].

This paper focus on the study of integer linear programming (ILP) formulations [43] for different clustering problems related with structural balance. Our contribution is threefold. First, we formalize and describe a set of clustering problems defined on signed digraphs that can be used in the evaluation of structural balance in social networks. In doing so, we establish a common notation for these problems since, as we have mentioned before, a same problem appears in the literature under different names. Second, we introduce an ILP formulation for the clustering problem proposed in [20] as a model for the relaxed structural balance. We discuss the advantages and limitations of this formulation by presenting extensive computational experiments. Third, we discuss alternative models for the relaxed structural balance and also present ILP formulations and numerical results.

ILP formulations have been successfully used to solve graph clustering problems [4, 8, 27, 31, 36]; some of them related with social network analysis. In [8], a collection of ILP formulations was presented for blockmodeling of social networks with only positive relations while in [4], a rounding algorithm of an ILP formulation was proposed for modularity-maximization of graph communities. Another advantage of applying ILP formulations to the evaluation of structural balance is that little effort is required to implement an ILP formulation by using one of the available commercial packages [41]. Thus, even a nonspecialist in mathematical modeling is capable of implementing an ILP formulation as well as of adapting an existent formulation for accommodating the characteristics of a specific problem. Unfortunately, that is not the case for ad-hoc branch-and-bound procedures as the ones presented in [7, 9].

The remainder of the paper is structured as follows. In Section 2, we give some notations and definitions to be used throughout this text. We establish a common notation for clustering problems treated in this paper and we present a literature review on its applications and solution approaches. We also discuss the computational complexity of the problem introduced in [20] (a proof that this problem is NP-hard is given in Appendix A). ILP formulations are presented in Section 3 for the problems stated in the previous section. Computational results are reported for test problems available in the literature and for a set of random instances. In Section 4, alternative models are presented for the relaxed

structural balance: a symmetric version of the model presented in [20] and a model that looks primarily for mediation process in social networks. Again, computational results are reported for literature instances. Finally, in Section 5, we present concluding remarks and discuss directions for further investigation.

## 2. Notation and problem definition

Let  $D = (V, A)$  be a digraph where  $V = \{1, 2, \dots, n\}$  is the set of vertices and  $A$  is a set of arcs connecting pairs of vertices. In this text, a digraph is assumed to have no loops. For a vertex set  $S \subseteq V$ , let  $A[S] = \{(i, j) \in A \mid i, j \in S\}$  denote the *subset of arcs induced by  $S$* . For two vertex sets  $S, W \subseteq V$ , let  $A[S : W] = \{(i, j) \in A \mid i \in S, j \in W\}$ . One observes that, by definition,  $A[S : S] = A[S]$ . A *partition* of  $V$  is a division of  $V$  into non-overlapping and non-empty subsets. A partition of  $V$  into  $l$  subsets is called here a  *$l$ -partition*.

Consider a function  $s : A \rightarrow \{+, -\}$  that assigns a sign to each arc in  $A$ . A digraph  $D$  together with a function  $s$  is called a *signed digraph*. Let  $D = (V, A, s)$  denote a signed digraph. An arc  $a \in A$  is called *negative* if  $s(a) = -$  and *positive* if  $s(a) = +$ . Let  $A^-$  and  $A^+$  denote, respectively, the set of negative and positive arcs in a signed digraph. Notice that, according to the definitions above,  $A = A^- \cup A^+$ .

A signed digraph  $D = (V, A, s)$  is *balanced* if its vertex set  $V$  can be divided into sets  $S$  (possibly empty) and  $V \setminus S$  in such a way that  $A[S] \cup A[V \setminus S] = A^+$ . An extension of this definition is given next.

Consider an integer parameter  $k$  with  $1 \leq k \leq n$ . A signed digraph  $D = (V, A, s)$  is  *$k$ -balanced* if its vertex set  $V$  can be divided into sets  $S_1, S_2, \dots, S_k$ , some of them possibly empty, such that  $\cup_{1 \leq i \leq k} A[S_i] = A^+$ . This definition can be equivalently stated as follows. A signed digraph  $D = (V, A, s)$  is  *$k$ -balanced* if there exists a  $l$ -partition  $P = \{S_1, S_2, \dots, S_l\}$  of  $V$ , with  $l \leq k$ , such that  $\cup_{1 \leq i \leq l} A[S_i] = A^+$ . A signed digraph is *clusterable* if it is  $n$ -balanced<sup>1</sup>. A clusterable signed digraph is called a *colorable* signed digraph in [13]. For an illustration of these definitions, we refer the reader to Figure 1 where negative and positive arcs are represented, respectively, by red and black arcs.

Consider a  $l$ -partition  $P = \{S_1, S_2, \dots, S_l\}$  of  $V$ . The *cut arcs* and the *uncut arcs* related with this partition are defined, respectively, as the arcs in sets  $\cup_{1 \leq i \neq j \leq l} A[S_i : S_j]$  and  $\cup_{1 \leq i \leq l} A[S_i]$ . Let  $w_a$  be a nonnegative arc weight associated with arc  $a \in A$ . Also, for  $1 \leq i \neq j \leq l$ , let

$$\Omega^+(S_i, S_j) = \sum_{a \in A^+ \cap A[S_i : S_j]} w_a$$

---

<sup>1</sup>The definition of a  $k$ -balanced signed digraph is slight different in [16, 19]. In these works, the value  $k$  is not a fixed parameter and the concepts of a clusterable and a  $k$ -balanced signed digraph are the same.

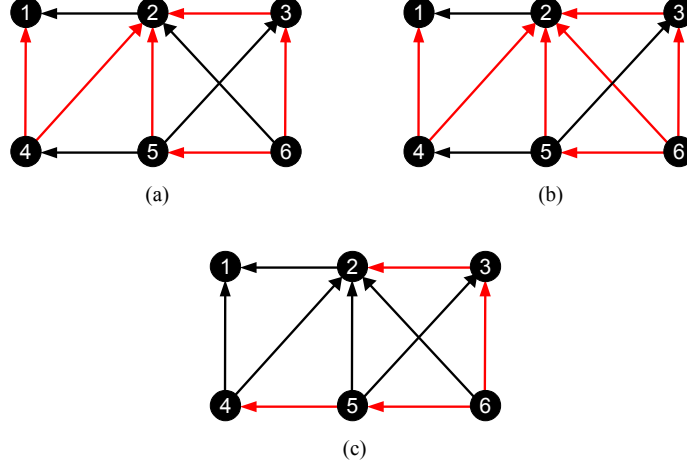


Figure 1: (a) A balanced signed digraph:  $S = \{1, 2, 6\}$  and  $V \setminus S = \{3, 4, 5\}$ . (b) A  $k$ -balanced signed digraph for any  $k \in \{3, 4, 5, 6\}$ . Sets  $S_1 = \{1, 2\}$ ,  $S_2 = \{3, 4, 5\}$  and  $S_3 = \{6\}$  define a 3-partition of this graph. (c) A non-clusterable signed digraph.

and

$$\Omega^-(S_i, S_j) = \sum_{a \in A^- \cap A[S_i : S_j]} w_a.$$

The *imbalance*  $I(P)$  of a partition  $P$  is defined as the total weight of negative uncut arcs and positive cut arcs, i.e.,

$$I(P) = \sum_{1 \leq i \leq l} \Omega^-(S_i, S_i) + \sum_{1 \leq i \neq j \leq l} \Omega^+(S_i, S_j). \quad (1)$$

Likewise, the *balance*  $B(P)$  of a partition  $P$  is defined as the total weight of positive uncut arcs and negative cut arcs. Clearly,  $B(P) + I(P) = \sum_{a \in A} w_a$ .

Next, we define a clustering problem whose optimal solution is used in [19] as a measure for the degree of balance in a social network.

**Problem 2.1** (CC problem). Let  $D = (V, A, s)$  be a signed digraph and  $w_a$  be a nonnegative arc weight associated with each arc  $a \in A$ . The *correlation clustering problem* is the problem of finding a partition  $P$  of  $V$  such that the imbalance  $I(P)$  is minimized. Let us denote this minimal value by  $\text{CC}(D)$ .

For the best of our knowledge, the CC problem was addressed for the first time in [19] (but not under this name) where a heuristic solution of this problem was used as a criteria for analyzing structural balance on social networks. The heuristic approach proposed by the authors is a greedy neighborhood search

procedure that assumes a prior knowledge of the number of clusters in the solution. Lately, in [6], motivated by the solution of a document clustering problem, the CC problem was formalized under this name. Due to the applications of this problem in the area of machine learning, it has been largely investigated but from the point of view of approximation algorithms [14, 17]. In [44] the CC problem is called *community mining*. The authors proposed an agent-based heuristic to the problem where no prior knowledge on the graph structure (the number of groups or a good initial solution) is required. Computational experiments were carried out over signed digraphs with tens of nodes available in the literature as well as over large random graphs for which the number of nodes range from 64 to 256. For the best of our known, the only exact approach for the CC problem is a branch-and-bound procedure proposed in [9] where the CC problem is called *K-balance partitioning problem*. This solution approach also assumes a prior knowledge of the number of clusters. Computational results were reported over up to 21 vertices.

In [20], the definition of a  $k$ -balanced signed digraph was informally extended in order to include relevant processes (polarization, mediation, differential popularity and subgroup internal hostility) that originally were viewed as violations of structural balance. Next, we formalize this definition. A signed digraph  $D = (V, A, s)$  is *k-relaxed balanced* if its vertex set  $V$  can be partitioned into sets  $S_1, S_2, \dots, S_l$ , with  $l \leq k$ , such that: all the arcs within a set have the same sign and all the arcs going between two sets have the same sign. That means, for each  $1 \leq i \leq l$ ,

$$A[S_i] \subseteq A^- \text{ or } A[S_i] \subseteq A^+,$$

and for each  $1 \leq i \neq j \leq l$ ,

$$A[S_i : S_j] \subseteq A^- \text{ or } A[S_i : S_j] \subseteq A^+.$$

Using this new definition, the structural balance was generalized to a version labeled *relaxed structural balance* [20]. This generalization gives rise to a new definition for the imbalance of a vertex partition. Let  $P = \{S_1, S_2, \dots, S_l\}$  be a  $l$ -partition of  $V$ . The *relaxed imbalance*  $RI(P)$  of  $P$  is defined as

$$RI(P) = \sum_{1 \leq i \leq l} \min\{\Omega^+(S_i, S_i), \Omega^-(S_i, S_i)\} + \sum_{1 \leq i \neq j \leq l} \min\{\Omega^+(S_i, S_j), \Omega^-(S_i, S_j)\}. \quad (2)$$

Again, the *relaxed balance*  $RB(P)$  is defined in such a way that  $RB(P) + RI(P) = \sum_{a \in A} w_a$ . Consider a partition  $P$  and a cut (uncut) arc  $(i, j)$ . The contribution of arc  $(i, j)$  for the relaxed imbalance  $I(P)$  depends on the sign of other cut (uncut) arcs. On the other hand, the contribution of arc  $(i, j)$  for the imbalance  $I(P)$  depends only on its own sign. The two measures of imbalance are related as follows.

**Remark 2.2.**  $RI(P) \leq I(P)$  for each partition  $P$  of  $V$ .

This new definition of imbalance defines a new criteria to evaluate balancing in a signed digraph and gives rise to another graph clustering problem.

**Problem 2.3** (RCC problem). Let  $D = (V, A, s)$  be a signed digraph,  $w_a$  be a nonnegative arc weight associated with each arc  $a \in A$  and  $k$  be an integer value satisfying  $1 \leq k \leq n$ . The *relaxed correlation clustering problem* is the problem of finding a  $l$ -partition  $P$  of  $V$ , with  $l \leq k$ , such that the relaxed imbalance  $RI(P)$  is minimized. Let us denote this minimal value by  $RCC(D, k)$ .

The RCC problem is closely related with the CC problem but it is not a particular case nor a generalization of the CC problem. Actually, each feasible solution (a graph partition) of the RCC problem is also a feasible solution of the CC problem but the problems have different cost functions, i.e., different ways of evaluating the imbalance of a partition. The RCC problem is intuitively as hard as the CC problem. We next establish that this problem is NP-hard; a formal proof is given in Appendix A.

**Lemma 2.4.** *The RCC problem is NP-hard.*

Given a signed digraph  $D$  and an arc weight vector  $\mathbf{w} \in \mathbb{R}^{|A|}$ , only one instance of the CC problem is defined for  $D$  and  $\mathbf{w}$ . On the other hand, for each value of  $1 \leq k \leq n$ , a different instance of the RCC problem is defined over  $D$  and  $\mathbf{w}$ . The optimal solution of each of these instances determines an ideal partition (in at most  $k$  sets) of the signed network associated with  $D$  that includes social psychological processes like polarization, mediation, differential popularity and subgroup internal hostility (see [7, 20]). The optimal values of these  $n$  problems are related as follows.

**Remark 2.5.**  $RCC(D, 2) \geq RCC(D, 3) \geq \dots \geq RCC(D, n) = 0$ .

To the best of our knowledge, the RCC problem has been applied only for the evaluation of structural balance in social networks. However, as it is mentioned in [39] “scientists studying community detection and those studying data clustering are obviously looking at the same coin”. Hence, the RCC problem could be also used as an approach to efficient community detection. Two solution methods were proposed in the literature for the RCC problem: a greedy heuristic approach [20] and a branch-and-bound procedure [7]. Computational experiments with both procedures were reported over literature instances with up to 29 vertices and for random instances with  $|V| \in \{20, 40\}$  (see [7, 20]). For the branch-and-bound procedure, the values considered for  $k$  were  $k \leq 7$  for literature instances and  $k \in \{3, 5\}$  for the set of random instances.

### 3. Integer linear programming formulations

In this section, we describe ILP formulations for the problems defined in Section 2.

The classical formulation for the CC problem is an ILP model proposed to uncapacitated clustering problems [17] in which a binary decision variable  $x_{ij}$  is assigned to each pair of vertices  $i, j \in V$ ,  $i \neq j$ , and defined as follows.



$$x_{ij} = \begin{cases} 0 & \text{if vertex } i \text{ and } j \text{ are in a common set,} \\ 1 & \text{otherwise.} \end{cases}$$

This ILP formulation minimizes the total imbalance and is described in the following.

$$\text{minimize } \sum_{(i,j) \in A^-} w_{ij}(1 - x_{ij}) + \sum_{(i,j) \in A^+} w_{ij}x_{ij} \quad (3)$$

$$\text{subject to } x_{ip} + x_{pj} \geq x_{ij}, \quad \forall i, p, j \in V, \quad (4)$$

$$x_{ij} = x_{ji}, \quad \forall i, j \in V, \quad (5)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i, j \in V. \quad (6)$$

The triangle inequalities (4) say that if vertices  $i$  and  $p$  are in a same cluster as well as vertices  $p$  and  $j$ , then vertices  $i$  and  $j$  are also in a same cluster. Constraint (5) written to  $i, j \in V$  establishes that variables  $x_{ij}$  and  $x_{ji}$  assume always the same value in this formulation. Constraints (6) impose binary restrictions to the variables while the objective function (3) minimizes the total imbalance defined by equation (1). Let us refer to this formulation as IP(CC).

To the best of our knowledge, the RCC problem has not been formulated as an ILP problem in the literature before. The authors in [7] presented a model that tries to describe this optimization problem but that, in fact, do not provide a mathematical description of a vertex partition. We use a representatives formulation to describe a feasible solution for the RCC problem, which means a partition of vertex set  $V$ . Representatives formulations have been successfully applied to the solution of other graph clustering problems [5, 10, 11, 24] including one balancing problem defined over a signed graph [22]. The main idea behind a representatives formulation is the unique representation of a cluster by its vertex with the lowest index. In our formulation, we use binary decision variables defined as follows. For each vertex  $i \in V$  we define

$$x_i^i = \begin{cases} 1 & \text{if } i \text{ is a representative vertex,} \\ 0 & \text{otherwise.} \end{cases}$$

For each pair of vertices  $i, j \in V$ , we define

$$x_j^i = \begin{cases} 1 & \text{if } i < j \text{ and vertex } j \text{ is represented by vertex } i, \\ 0 & \text{otherwise.} \end{cases}$$

Also, a set of binary variables is used to describe the set of uncut arcs. For a vertex  $i \in V$  and an arc  $(p, q) \in A$ , we define

$$t_{pq}^i = \begin{cases} 1 & \text{if } i \leq p, i \leq q \text{ and vertices } p \text{ and } q \text{ are both represented by } i, \\ 0 & \text{otherwise.} \end{cases}$$

Finally, a set of binary variables is used to describe the set of cut arcs. For each pair of vertices  $i, j \in V$  and for each arc  $(p, q) \in A$ , we define

$$y_{pq}^{ij} = \begin{cases} 1 & \text{if } i \leq p, j \leq q, p \text{ is represented by } i \text{ and } q \text{ is represented by } j \\ 0 & \text{otherwise.} \end{cases}$$

From now on, let  $S_i$  be a set of vertices represented by vertex  $i \in V$ ; we assume that  $S_i = \emptyset$  whenever  $i$  is not a representative vertex. Notice that,  $t_{pq}^i = 1$  whenever  $(p, q) \in A[S_i]$  and  $y_{pq}^{ij} = 1$  whenever  $(p, q) \in A[S_i : S_j]$ . Although in the representatives formulation a cluster is represented by its vertex with the lowest index, for the sake of keeping the formulation presented here as simple as possible, binary variables were defined for each combination of vertices  $i, j \in V$  and arc  $(p, q) \in A$ . A simple pre-processing procedure can be used to eliminate all the binary variables not satisfying this condition.

A set of continuous variables is used to describe functions  $\Omega^+$  and  $\Omega^-$  appearing in the definition of the relaxed imbalance given by equation (2). For each vertex  $i \in V$  we define  $c_i = \min\{\Omega^+(S_i, S_i), \Omega^-(S_i, S_i)\} \in \mathbb{R}$ . Likewise, we define  $d_{ij} = \min\{\Omega^+(S_i, S_j), \Omega^-(S_i, S_j)\} \in \mathbb{R}$  for each pair of vertices  $i, j \in V$ . The formulation follows.

$$\text{minimize } \sum_{i \in V} c_i + \sum_{i, j \in V} d_{ij} \quad (7)$$

$$\text{subject to } \sum_{i \in V : i \leq j} x_j^i = 1, \quad \forall j \in V, \quad (8)$$

$$x_j^i \leq x_i^i, \quad \forall i, j \in V : i < j, \quad (9)$$

$$\sum_{i \in V} x_i^i \leq k, \quad (10)$$

$$x_j^i \in \{0, 1\}, \quad \forall i, j \in V : i \leq j, \quad (11)$$

$$t_{pq}^i = x_p^i x_q^i, \quad \forall i \in V, (p, q) \in A, \quad (12)$$

$$y_{pq}^{ij} = x_p^i x_q^j, \quad \forall i, j \in V, (p, q) \in A, \quad (13)$$

where,

$$c_i = \min\left\{ \sum_{(p,q) \in A^+} w_{(p,q)} t_{pq}^i, \sum_{(p,q) \in A^-} w_{(p,q)} t_{pq}^i \right\}, \quad \forall i \in V, \quad (14)$$

$$d_{ij} = \min\left\{ \sum_{(p,q) \in A^+} w_{(p,q)} y_{pq}^{ij}, \sum_{(p,q) \in A^-} w_{(p,q)} y_{pq}^{ij} \right\}, \quad \forall i, j \in V. \quad (15)$$

Constraints (8) establish that vertex  $j$  must be represented by exactly one vertex. Constraints (9) forbid vertex  $j$  to be represented by vertex  $i$  unless  $i$  is a representative vertex. These constraints, together with (10) and integrality constraints (11), define a partition of  $V$  in at most  $k$  clusters. Constraints (12) define the set of arcs linking vertices within a same cluster. Notice that, we do not need to impose integrality constraints on variables  $t_{pq}^i$  since constraints (12) together with integrality constraints (11) ensure that  $t_{pq}^i \in \{0, 1\}$ . In a similar way, constraints (13) define the set of arcs linking vertices in two different clusters. Finally, equations (14) and (15) define, respectively, the first and second

sums in equation (2) while the objective function (7) looks for a partition that minimizes the relaxed imbalance.

The quadratic constraints (12) and (13) can be linearized [43] as follows.

$$x_p^i + x_q^i - t_{pq}^i \leq 1, \quad \forall i \in V, (p, q) \in A, \quad (16)$$

$$t_{pq}^i \leq x_p^i, \quad \forall i \in V, (p, q) \in A, \quad (17)$$

$$t_{pq}^i \leq x_q^i, \quad \forall i \in V, (p, q) \in A, \quad (18)$$

$$0 \leq t_{pq}^i \leq 1, \quad \forall i \in V, (p, q) \in A, \quad (19)$$

$$x_p^i + x_q^j - y_{pq}^{ij} \leq 1, \quad \forall i, j \in V, (p, q) \in A, \quad (20)$$

$$y_{pq}^{ij} \leq x_p^i, \quad \forall i, j \in V, (p, q) \in A, \quad (21)$$

$$y_{pq}^{ij} \leq x_q^j, \quad \forall i, j \in V, (p, q) \in A, \quad (22)$$

$$0 \leq y_{pq}^{ij} \leq 1, \quad \forall i, j \in V, (p, q) \in A. \quad (23)$$

Constraints (16) ensure that  $t_{pq}^i = 1$  whenever  $x_p^i = x_q^i = 1$ . On the other hand, constraints (17) and (18) impose  $t_{pq}^i = 0$  whenever  $x_p^i = 0$  or  $x_q^i = 0$ . In a similar way, constraints (20) ensure that  $y_{pq}^{ij} = 1$  whenever  $x_p^i = x_q^j = 1$  while constraints (21) and (22) are in charge of setting  $y_{pq}^{ij} = 0$ .

Since the RCC problem is a minimization problem, equations (14) and (15) can be replaced by the following disjunctions.

$$\left( c_i = \sum_{(p,q) \in A^+} w_{(p,q)} t_{pq}^i \right) \vee \left( c_i = \sum_{(p,q) \in A^-} w_{(p,q)} t_{pq}^i \right), \quad \forall i \in V, \quad (24)$$

$$\left( d_{ij} = \sum_{(p,q) \in A^+} w_{(p,q)} y_{pq}^{ij} \right) \vee \left( d_{ij} = \sum_{(p,q) \in A^-} w_{(p,q)} y_{pq}^{ij} \right), \quad \forall i, j \in V. \quad (25)$$

A disjunction of linear constraints can be equivalently replaced by a set of linear inequalities written over a set of additional binary variables [43]. Thus, (24) can be replaced by the following linear inequalities with  $M^+ = \sum_{a \in A^+} w_a$  and  $M^- = \sum_{a \in A^-} w_a$ .

$$c_i \leq \sum_{(p,q) \in A^+} w_{(i,j)} t_{pq}^i, \quad \forall i \in V, \quad (26)$$

$$c_i \leq \sum_{(p,q) \in A^-} w_{(i,j)} t_{pq}^i, \quad \forall i \in V, \quad (27)$$

$$c_i - \sum_{(i,j) \in A^+} w_{(i,j)} t_{pq}^i - M^+ r_i \geq -M^+, \quad \forall i \in V, \quad (28)$$

$$c_i - \sum_{(i,j) \in A^-} w_{(i,j)} t_{pq}^i - M^-(1 - r_i) \geq -M^-, \quad \forall i \in V, \quad (29)$$

$$r_i \in \{0, 1\}, \quad \forall i \in V, \quad (30)$$

$$c_i \geq 0, \quad \forall i \in V. \quad (31)$$

If  $r_i = 1$ , constraint (29) becomes redundant and constraints (26) and (28) force  $c_i$  to be equal to the rhs of (26). On the other hand, if  $r_i = 0$ , constraint (28) becomes redundant and constraints (27) and (29) force  $c_i$  to be equal to the rhs of (27). Hence, if the rhs of (26) is smaller than the rhs of (27),  $r_i$  will be forced to be zero. In the opposite case, i.e., if the rhs of (27) is smaller than the rhs of (26),  $r_i$  will be forced to be one. Likewise, (25) can be replaced by the following linear inequalities.

$$d_{ij} \leq \sum_{(p,q) \in A^+} w_{(p,q)} y_{pq}^{ij}, \quad \forall i, j \in V, \quad (32)$$

$$d_{ij} \leq \sum_{(p,q) \in A^-} w_{(p,q)} y_{pq}^{ij}, \quad \forall i, j \in V, \quad (33)$$

$$d_{ij} - \sum_{(p,q) \in A^+} w_{(p,q)} y_{pq}^{ij} - M^+ s_{ij} \geq -M^+, \quad \forall i, j \in V, \quad (34)$$

$$d_{ij} - \sum_{(p,q) \in A^-} w_{(p,q)} y_{pq}^{ij} - M^-(1 - s_{ij}) \geq -M^-, \quad \forall i, j \in V, \quad (35)$$

$$s_{ij} \in \{0, 1\}, \quad \forall i, j \in V, \quad (36)$$

$$d_{ij} \geq 0, \quad \forall i, j \in V. \quad (37)$$

The RCC problem is equivalent to the following ILP problem:

$$\begin{aligned} & \text{minimize } \sum_{i \in V} c_i + \sum_{i,j \in V: i < j} d_{ij} \\ & \text{subject to } (8) - (11), (16) - (23) \text{ and } (26) - (37). \end{aligned}$$

Let us refer to this formulation as IP(RCC). The linear programming (LP) relaxation of this formulation is obtained by dropping all its binary constraints. Besides having a large number of binary variables, formulation IP(RCC) also

has many “big-M” constraints. It is well known that these are factors that lead, in general, to weak LP relaxations and make the solution of the formulation numerically difficult.

Since the relaxed imbalance (2) is a minimization function, the RCC problem can naturally be seen as a mixed integer bilevel linear programming (MIBL) problem [37]. The solution of the RCC problem involves the solution of optimization problems at different levels, each one solved in a non-cooperatively and sequential way. The first-level optimization problem is in charge of choosing a partition of vertex set  $V$ . As an answer, the second-level optimization problem chooses the way the relaxed imbalance associated with that partition is calculated. For that reason, the RCC problem could also have been written as MIBL problem that becomes the formulation IP(RCC) after applying classical bilevel reformulation ideas.

### 3.1. Computational experiments with the ILP formulations

We report computational experiments carried out with the formulations described in this section. We evaluate each formulation on a set of 22 benchmark instances from the literature. For formulation IP(CC), we also present numerical results obtained with a set of random instances. The formulations are coded in Xpress Mosel 3.2.0 with solver Xpress Optimizer 21.01.00 on a Sony Vaio computer with a processor Intel Core 2 Duo of 2.10 GHz and 3 GB of RAM memory. The CPU time limit was set to 1h for all instances and procedures. Before presenting the obtained results, we briefly describe the set of instances used in our experiments.

*Benchmark instances from the literature.* Table 1 describes the signed graphs associated with these 22 instances. Besides the name of each instance, this table exhibits the number of vertices  $|V|$ , the graph density  $d = |A|/(|V|^2 - |V|)$  and the total number of positive and negative arcs,  $|A^+|$  and  $|A^-|$ . Also, the last column in this table gives us the sum of all arc weights in the signed digraph,  $\sum_{a \in A} w_a$ .

*House instances.* These instances were generated in 1952 by Lemann *et al.* [32] in a sociometric study. This set of networks represents information obtained from students living in three different dormitories labeled as Houses A, B and C. Originally, four signed networks were defined, each one associated with one indicator (denoted Date, Friend, Roommate and Weekend). A fifth signed network (denoted Sum) was defined in [18] by adding up arc weights from these four signed networks. This set of instances has been analyzed in [18] where a multiple indicator approach is proposed to blockmodeling of signed networks and in [7] where a branch-and-bound procedure is proposed to the RCC problem.

*Monastery instances.* The next four instances are the very known Sampson monastery data defined in 1968 [40]. Sampson collected data on four signed relationships (affect, esteem, influence and sanction) for a group of young men who were either postulants or novices at a monastery in different time periods.

A signed network can be defined for each combination of signed relationship and time period. The first three signed networks (denoted Monk T2, T3 and T4) describes the affect relationship for three different time periods. The instance MonkT4 Sum was defined in [18] by adding up arc weights from the four signed networks relationships on time period T4. These instances have been used in the evaluation of solution approaches proposed to the CC problem and to the RCC problem [7, 9, 19].

*Manning and Shofner’s lipread consonants data.* The next instance do not come from the structural balance literature but from a study of 21 lipread consonants similarity performed by Manning *et al.* [34]. In this study, individuals were asked to rate on a given scale the similarity between pairs of consonants. The average over all ratings was calculated to produce a similarity matrix with values in the interval  $[-2, 2]$ . This data has been used in the evaluation of a branch-and-bound procedure proposed to the solution of the CC problem [9]. This is a very dense digraph with most negative arcs, which means most consonant pairs were judged as dissimilar by the respondents.

*McKinney instance.* This instance was introduced in the literature of computational methods for structural balance by Brusco *et al.* [7] which intended to test their methodology on a large and dense signed social network. This signed digraph was defined based on the data collected by McKinney in 1948 [35] during a study about relationships among children in a same classroom. In a sociometric test, students were asked to chose among “willing to serve with other children”, rated as +1, “not being willing to serve”, rated as 1, and “indifferent”, rated as 0, as a definition to their behavior in a discussion group with the other members of the class.

*Newcomb instance.* The last instance is a very known signed social network that has been analyzed in some works cited here [7, 19, 20]. The original data is from 1961 and comes from a sociometric study with students in a dormitory at a university that lasted several weeks [38]. This data has been slightly changed by Doreian *et al.* [20] and here we consider this modified version of the instance.

For additional details on the definition of these 22 instances, we refer the interested reader to the works we have cited. From Table 1 we see that, these signed digraphs vary from low-density (0.3) to high-density (0.99) with the number of vertices varying from 17 to 29 which makes this set, in some sense, a heterogeneous set of instances. However, for most digraphs, the negative density  $|A^-|/|A|$  is around 0.5 with only instance “Manning Shofner” having a high negative density equal to 0.8125.

*Random Instances.* We generated random signed digraphs by varying the number of vertices  $|V|$ , the graph density  $d = |A|/(|V|^2 - |V|)$  and the negative graph density defined here as  $d^- = |A^-|/|A|$ . We considered a set of 58 random instances having  $|V|$ ,  $d$  and  $d^-$  ranging, respectively, in the sets  $\{20, 30, 40, 50\}$ ,

$\{0.1, 0.2, 0.5, 0.8\}$  and  $\{0.2, 0.5, 0.8\}$ .

Table 2 presents the results obtained on solving the CC problem on the set of literature instances. The first multicolumn on this table identify the instances. The second multicolumn gives us information about the solution process with IP(CC) formulation:  $I(P)$  is the imbalance of the optimal solution,  $Time$  is the time (in seconds) spent to solve the instances to optimality, and  $|K|$  is the number of clusters in the optimal solution. We also solved these instances by using: the branch-and-bound procedure described in [9]; and the Doreian-Mrvar heuristic [19, 20] available in Pajek software (version 2.0 with number of iterations set to  $10^3$ ). Pajek software [1] is a powerful program for analysis and visualization of large networks which is freely available for noncommercial use. We used a Fortran implementation of the branch-and-bound code made available by the authors. This Fortran implementation works with a limitation of 10 clusters. Remember that, as we have mentioned before, both methods need as an input the number of clusters in the output solution. We fed these methods with the number of clusters in the solution obtained by the IP(CC) formulation. The next two multicolumns in Table 2 give us information about these solution processes. We do not report the solution obtained for House C instances by the branch-and-bound procedure since the Fortran code failed to solve this instance. We can see that these instances are very easy instances of the CC problem solved to optimality in some seconds by the exact approaches. Pajek was able to find the optimal solution for all instances.

In order to identify the limitation of the IP(CC) formulation, we solved the CC problem on a set of random instances. Table 3 presents the obtained results. The notations in this table are the same as in Table 2 except for columns *Nodes* and  $I(P)$ : *Nodes* informs us the number of nodes in the enumeration tree;  $I(P)$  informs us the imbalance of the best solution found by each method. Also, in this table “-” means the instance was not solved within the time limit and “\*” means the number of clusters in the IP(CC) solution exceeds the software limitation. Notice that, in multicolumn IP(CC),  $I(P)$  informs us the optimal value if the instance was solved in the time limit, otherwise it gives us an upper bound for the optimal value. We can see that the branch-and-bound code fails (within the time limit of 1h) for instances with more the 20 vertices and negative density 0.5 and 0.8. The IP(CC) formulation was able to solve all instances with 20 and 30 vertices and starts to fail with instances of 40 vertices and negative density equal to 0.5. In the second multicolumn, the eight instances marked in bold were not solved to optimality by Pajek after  $10^3$  iterations. For each instance in this subset, if the time spent by Pajek was smaller than the time spent by IP(CC), we run again Pajek with number of iterations of Doreian-Mrvar heuristic set to  $10^6$  but imposing a time limit equal to the number of seconds spent by IP(CC). Pajek was able to find the optimal solution for almost all random instances except for six; the number of instances not optimally solved increases with the number of vertices.

Now we turn our attention to the solution of the RCC problem with the ILP formulation IP(RCC). We run this formulation on 9 literature instances

from Table 1. For the House instances, we restrict our experiments only to the instances House Sum that have also been used in [7] for evaluating the branch-and-bound procedure. We run the experiments with this formulation in the following way. We started by setting  $k = 2$  and run the formulation. If the formulation solves the problem to optimality within the time limit, we increment the value of  $k$  and repeat the process. If it is not the case, we do not increment  $k$ . Now, we set  $k = n - 1$  and run the formulation. The same process is repeated but now decrementing the value of  $k$  until the formulation is not able to solve the problem within the time limit. Notice that, according to Remark 2.5, the optimal solution of  $RCC(D, k)$  can be used as a lower bound when solving the problem  $RCC(D, k - 1)$ . The obtained results are registered in Tables 4, 5 and 6. The notations in these tables are the same as in the previous ones except for column *Gap* that presents the gaps calculated between the best integer solution found and the final lower bound. Since we do not have access to the code of the branch-and-bound procedure described in [7], we do not report results for this method. In [7], results were reported for the branch-and-bound procedure with  $2 \leq k \leq 7$  and, according to the authors, the number of cluster had a strong effect on the computation time, which is totally expected. From the results in Tables 4, 5 and 6 we see that the same happens with the ILP formulation, but in a different way. Roughly speaking, the IP(RCC) formulation presented its best results for  $k = 2$ ,  $k = 3$  (for some instances) and for high values of  $k$ . For the other values of  $k$ , the representatives formulation had poor LP relaxations (resulting from both the linearization of 0-1 variable products and the big-M constraints) leading to poor convergence of the enumeration solution process.

#### 4. Alternative models for generalized structural balance

The main idea of generalized structural balance, proposed by Doreian *et al.* [20], is to take into account more complex structures when evaluating balance in a social network. These authors have modeled this new concept as an instance of the RCC problem: the solution of the RCC problem is proposed as a methodology to analyze balance in social networks. In this section, we propose alternative ways to model the generalized structural balance.

##### 4.1. A symmetric version of RCC problem

Consider a signed digraph  $D$  with arc weights  $\mathbf{w} \in \mathbb{R}^{|\mathbf{A}|}$  and let  $P$  be a partition of its vertex set. The definition of relaxed imbalance  $RI(P)$  given by equation (2), and associated with the relaxed structural balance, has its roots in blockmodeling approaches [7, 20]. If we consider the arc weight matrix associated with the signed graph and we divide its lines and columns according to the sets in  $P$ , the imbalance  $RI(P)$  is in fact the sum of the imbalance in the different blocks of this matrix. Consider the signed networks depicted in Figure 2 where negative and positive arcs are represented, respectively, as red and black arcs. Figure 2 (a) exhibits the optimal solution of the RCC problem when  $k = 3$ : a partition  $P = \{S_1, S_2, S_3\}$  with  $S_1 = \{A, B, C, D, E, F\}$ ,  $S_2 = \{G, H, I, J\}$



and  $S_3 = \{K, L, M\}$  and having the relaxed imbalance  $RI(P) = 0$ . Notice that, there are ten negative arcs going from  $S_1$  to  $S_2$  that characterize ten negative relations from elements in cluster  $S_1$  toward elements in cluster  $S_2$ . However, there are eight positive relations in the other sense, i.e., from elements in  $S_1$  toward elements in  $S_2$ . In some contexts, this can be viewed as a degree of imbalance. Thus, we propose a redefinition for the relaxed imbalance of a partition  $P$  taking into account now symmetric relationships, as follows.

$$SRI(P) = \sum_{1 \leq i \leq l} \min\{\Omega^+(S_i, S_i), \Omega^-(S_i, S_i)\} + \sum_{1 \leq i < j \leq l} \min\{\Omega^+(S_i, S_j) + \Omega^+(S_j, S_i), \Omega^-(S_i, S_j) + \Omega^-(S_j, S_i)\}.$$

The new definition for the relaxed imbalance gives rise to a symmetric version of the RCC problem that can be modeled as an ILP formulation by replacing inequality (15) with

$$d_{ij} = \min\left\{ \sum_{(p,q) \in A^+} w_{(p,q)}(y_{pq}^{ij} + y_{pq}^{ji}), \sum_{(p,q) \in A^-} w_{(p,q)}(y_{pq}^{ij} + y_{pq}^{ji}) \right\}, \quad \forall i, j \in V : i < j,$$

and the objective function (7) with

$$\sum_{i \in V} c_i + \sum_{i, j \in V : i < j} d_{ij}.$$

The partition  $P$  depicted in Figure 2 (a) has  $SRI(P) = 8$ . On the other hand, the partition  $P = \{S_1, S_2, S_3\}$ , depicted in Figure 2 (b), with  $S_1 = \{A, J, K, L, M\}$ ,  $S_2 = \{B, C, F, G, H\}$  and  $S_3 = \{D, E, I\}$ , has  $SRI(P) = 7$  and is the optimal solution for the Symmetric RCC problem.

Tables 8, 9 and 10 display the results obtained for the Symmetric RCC problem over the 9 literature instances used in the numerical experiments with the RCC problem. As we expected, for a given  $k$ , the imbalance associated with the optimal solution increased in most cases when we compare these results with the results obtained for the RCC problem. Moreover, many optimal partitions evaluated with no imbalance when solving the RCC problem were evaluated with an imbalance bigger than zero in the symmetric version. We also noticed that the ILP formulation becomes numerically more difficult to solve with the symmetric definition.

#### 4.2. Negative and positive mediation

As we have already mentioned, one of the ideas behind the definition of the relaxed structural balance given in [20] is the possible existence of mediators in

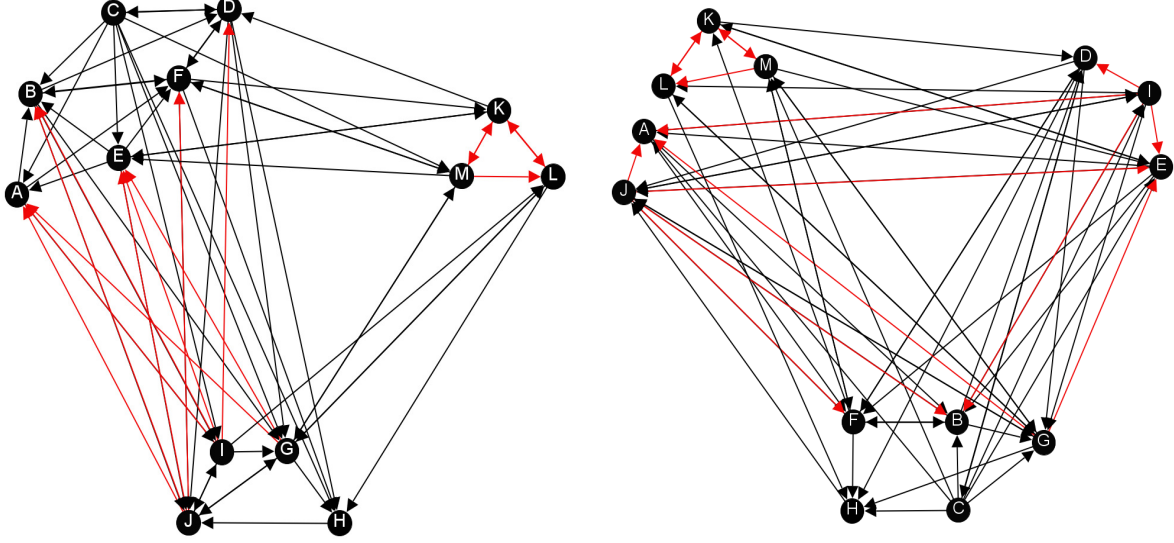


Figure 2: (a) Partition  $P = \{S_1, S_2, S_3\}$  with  $S_1 = \{A, B, C, D, E, F\}$ ,  $S_2 = \{G, H, I, J\}$  and  $S_3 = \{K, L, M\}$  has  $RI(P) = 0$  and  $SRI(P) = 8$ . (b) Partition  $P = \{S_1, S_2, S_3\}$  with  $S_1 = \{A, J, K, L, M\}$ ,  $S_2 = \{B, C, F, G, H\}$  and  $S_3 = \{D, I, E\}$  has  $RI(P) = 7$  and  $SRI(P) = 7$ .

social networks (see Figures 1 and 2 in [20]). The actors in a mediation group could be mutually hostile or mutually favorable, what we call here, *negative mediation* and *positive mediation*, respectively. Motivated specially by this social process, we propose two new variations of the CC problem. Our intention is to define problems that are numerically easier to solve than the RCC problem and whose optimal solution can help to identify the mediation process in social networks. Additionally, in other contexts, finding the best mediation group in a network can be a useful information. For example, in a political group (such as a parliament, a set of political parties or any group responsible for public decision-making), one may want to define a group of negotiators mutually hostile according to a specific subject but that maximizes the positive relation with non mediators in this same political group.

Additional notation will be necessary before we can proceed. Let  $S \subseteq V$  be a subset of vertices such that  $(i, j) \notin A^-$ , for each pair of vertices  $i, j \in S$ . The subset  $S$  is called a *positive mediation set* in  $D$ . On the other way, let  $S \subseteq V$  be a subset of vertices such that,  $(i, j) \notin A^+$ , for each pair of vertices  $i, j \in S$ , and such that,  $(i, j) \notin A^-$ , for each pair of vertices  $i \in S$  and  $j \in V \setminus S$ . The subset  $S$  is called a *negative mediation set* in  $D$ . The next two problems look for a signed graph partition where mediation groups are not considered as a contribution to the imbalance of the network.

**Problem 4.1** (CCNM problem). Let  $D = (V, A, s)$  be a signed digraph and  $w_a$  be a nonnegative arc weight associated with each arc  $a \in A$ . The *correlation clustering problem with negative mediation* is the problem of finding a partition  $P = \{S_1, S_2, \dots\}$  of  $V$  such that  $S_1$  is a negative mediation set in  $D$  and the imbalance  $I(P \setminus S_1)$  is minimized. Let us denote this minimal value by  $\text{CCNM}(D)$ .

**Problem 4.2** (CCPM problem). Let  $D = (V, A, s)$  be a signed digraph and  $w_a$  be a nonnegative arc weight associated with each arc  $a \in A$ . The *correlation clustering problem with positive mediation* is the problem of finding a partition  $P = \{S_1, S_2, \dots\}$  of  $V$  such that  $S_1$  is a positive mediation set in  $D$  and the imbalance  $I(P \setminus S_1)$  is minimized. Let us denote this minimal value by  $\text{CCPM}(D)$ .

Notice that, again, the CCNM and CCPM problems are neither generalizations nor particular cases of the CC problem. In fact, like the RCC problem, each one of these problems proposes a different way of evaluating the imbalance in a signed digraph. The next remark states how the optimal values of these problems are related.

**Remark 4.3.** Let  $D = (V, A, s)$  be a signed digraph and  $w_a$  be a nonnegative arc weight associated with each arc  $a \in A$ . The optimal values of the CC, CCPM and CCNM problems are such that

$$\text{CCPM}(D) \leq \text{CC}(D),$$

$$\text{CCNM}(D) \leq \text{CC}(D).$$

Now, let  $\text{CC}(D, k')$ ,  $\text{CCPM}(D, k')$  and  $\text{CCNM}(D, k')$ , denote, respectively, the optimal values of the CC, CCPM and CCNM problems if we restrict each problem's feasible set to the set of partitions with at most  $k'$  clusters. Then we have,

$$\text{RCC}(D, k') \leq \text{CCPM}(D, k') \leq \text{CC}(D, k'),$$

$$\text{RCC}(D, k') \leq \text{CCNM}(D, k') \leq \text{CC}(D, k').$$

Now, we discuss the solution of the CCPM and CCNM problems. According to the definition of positive mediation, a vertex can be a positive mediator no matter the role of its adjacent vertices. Thus, the CCPM problem is solved by applying a simple preprocessing procedure to generate  $S_1$  (the unique maximal set  $S_1$  of positive mediators in the digraph) and solving the CC problem defined over the digraph  $(V \setminus S_1, A[V \setminus S_1])$ .

The optimal solution of the CCNM problem is not so simple since, for each positive arc, at most one adjacent vertex can be a negative mediator. As a consequence, there can exist more than one maximal set of negative mediators. Thus, we describe an ILP formulation to the CCNM problem.

We use the set of binary decision variables  $x_{ij}$ ,  $i, j \in V$ , as defined before in formulation IP(CC). Additionally, we define a new set of binary variables as follows. For each vertex  $i \in V$ , we define

$$m_i = \begin{cases} 1 & \text{if vertex } i \text{ plays a negative mediation role,} \\ 0 & \text{otherwise.} \end{cases}$$

The formulation follows.

$$\text{minimize } \sum_{(i,j) \in A^-} w_{ij}(1 - x_{ij}) + \sum_{(i,j) \in A^+} w_{ij}(x_{ij} - m_i - m_j) \quad (38)$$

subject to (4), (5) and (6),

$$m_i + m_j \leq 1, \quad \forall (i, j) \in A^+, \quad (39)$$

$$m_i - m_j = 0, \quad \forall (i, j) \in A^-, \quad (40)$$

$$m_i \leq x_{ij}, \quad \forall i, j \in V, \quad (41)$$

$$m_i \in \{0, 1\}, \quad \forall i \in V. \quad (42)$$

Constraints (39) say that vertices  $i$  and  $j$  cannot be negative mediators at the same time if they are linked by a positive arc. Constraints (40) ensure that, if  $(i, j) \in A^-$  then either both vertices or none are mediators, thus we do not have a negative arc linking a mediator vertex to a non mediator vertex. Constraints (41) establish that, if vertex  $i$  is a mediator then it is considered as an isolated vertex in the graph. In our ILP formulation, the set of all isolated vertices defines the negative mediation set. The objective function (38) keeps the idea of searching for a partition that minimizes the imbalance. Notice that, in the second sum, due to (39) and (41),  $x_{ij} - m_i - m_j$  is either equal to one or equal to zero: the first case happens whenever  $i$  and  $j$  are not mediators and belong to different clusters. Let us refer to this formulation as IP(CCNM).

We solved the CCPM and CCNM problems on all instances described in Section 3.1. In our experiments, we observed that the numerical complexity for solving these problems was the same as the one for solving the CC problem: time spent to solve an instance was almost the same and the same instances remained unsolved. For most instances, the optimal solution for these problems was a partition with no mediation group, which means, the optimal solution of CC problem. Table 7 exhibits the results obtained for the instances where a mediation group was found. For instance McKinney, a partition was found with a positive mediation group with 17 elements. This partition is also an optimal solution for the RCC problem defined on this instance with  $k = 4$ . The positive mediation group is the reason why the relaxed imbalance for the optimal solution with  $k = 4$  is equal to zero.

## 5. Concluding remarks

This work addressed clustering graph problems related with the structural balance. We were primarily interested in the exact solution of these problems. The CC and RCC problems have already been used in the literature as tools for evaluating the structural balance of social networks. Branch-and-bound procedures [7, 9] have been proposed for their exact solution. We described

ILP formulations to CC and RCC problems. The benchmark instances for the CC problem were easily solved to optimality by an ILP formulation (denoted IP(CC)) usually used in the literature for graph clustering problems. Formulation IP(CC) does not need to be fed by the number of clusters in the optimal solution and this characteristic seems to be an advantage for this approach: the branch-and-bound procedure starts to fail before formulation IP(CC). We presented the first mathematical formulation for the RCC problem: a representatives formulation (denoted IP(RCC)). Formulation IP(RCC) is harder to solve than formulation IP(CC). For the set of benchmark instances, IP(RCC) was able to solve the problem when  $k = 2$ ,  $k = 3$  (for some instances) and for high values of  $k$ . The results presented for the branch-and-bound procedure in [7] showed that this approach was efficient in the solution of RCC instances with  $k \leq 8$ . For now, it seems that formulation IP(RCC) and the branch-and-bound procedure are complementary approaches for the efficient solution of the RCC problem.

We also discussed alternative models for the relaxed structural balance. The first model is a symmetric version of the RCC problem motivated by the idea that the RCC problem can underestimate the imbalance of a network partition. The relaxed structural balance is based in the identification of complex structures that must not be evaluated as imbalance and our second alternative model tries to identify one of these structures: the mediation process. One advantage of this model is that the definition of the problem focus in the structure we want to identify. As a consequence, the number of clusters is not a parameter of the problem (as it happens for the RCC problem) and the associated graph clustering problem becomes numerically easier.

In forthcoming studies, each ILP formulation discussed here could be strengthened by families of valid and facet-defining inequalities and branch-and-cut procedures could be implemented for the efficient solution of the associated clustering problem. A branch-and-cut procedure has been successfully applied to the solution of a related problem [21, 23]. Moreover, other mediation models could be tried by slightly changing the definition of the mediation group. Our definition for the negative (positive) mediation group do not accept the existence of positive (negative) relations inside the group. This could be relaxed and the positive (negative) relations could be counted as an imbalance in the objective function. Also, the existence of more than one mediation group could be included in the problem definition. However, the possibility of having more mediation groups would make the associated problem harder to solve.

- [1] Pajek. <http://pajek.imfm.si/>. Accessed: May 2012.
- [2] P. Abell and M. Ludwig. Structural balance: a dynamic perspective. *Journal of Mathematical Sociology*, 33:129–155, 2009.
- [3] G. Adejumo, P. R. Duimering, and Z. Zhong. A balance theory approach to group problem solving. *Social Networks*, 30:83–99, 2008.
- [4] G. Agarwal and D. Kempe. Modularity-maximizing network communi-

- ties using mathematical programming. *The European Physical Journal B*, 66:409–418, 2008.
- [5] L. Bahiense, Y. Frota, N. Maculan, T.F. Noronha, and C.C. Ribeiro. A branch-and-cut algorithm for equitable coloring based on a formulation by representatives. *Electronic Notes in Discrete Mathematics*, 35:347–352, 2009.
  - [6] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. In *Proceedings of the 43rd annual IEEE symposium of foundations of computer science*, pages 238–250, Vancouver, Canada, 2002.
  - [7] M. Brusco, P. Doreian, A. Mrvar, and D. Steinly. Two algorithms for relaxed structural balance partitioning: linking theory, models and data to understand social network phenomena. *Sociological Methods & Research*, 40:57–87, 2011.
  - [8] M. Brusco and D. Steinly. Integer programs for one- and two-model blockmodeling based on prespecified image matrices for structural and regular equivalence. *Journal of Mathematical Psychology*, 53:577–585, 2009.
  - [9] M. Brusco and D. Steinly.  $K$ -balance partitioning: an exact method with applications to generalized structural balance and other psychological contexts. *Psychological Methods*, 15:145–157, 2010.
  - [10] M. Campêlo, V.A. Campos, and R.C. Correa. On the asymmetric representatives formulation for the vertex coloring problem. *Discrete Applied Mathematics*, 156:1097–1111, 2008.
  - [11] M. Campelo, R.C. Correa, and Y. Frota. Cliques, holes and the vertex coloring polytope. *Information Processing Letters*, 89:1097–1111, 2004.
  - [12] D. Cartwright and F. Harary. Structural balance: A generalization of Heider’s theory. *Psychological Review*, 63:277–293, 1956.
  - [13] D. Cartwright and F. Harary. A note on a matrix criterion for unique colorability of a signed graph. *Psychometrik*, 32:291–296, 1967.
  - [14] M. Charikara, V. Guruswami, and A. Wirtha. Clustering with qualitative information. *Journal of Computer and System Sciences*, 71:360–383, 2005.
  - [15] B. DasGupta, G. A. Encisob, E. Sontag, and Y. Zhanga. Algorithmic and complexity results for decompositions of biological networks into monotone subsystems. *BioSystems*, 90:161–178, 2007.
  - [16] J.A. Davis. Clustering and structural balance in graphs. *Human Relations*, 20:181–187, 1967.
  - [17] E. D. Demaine, D. Emanuel, A. Fiat, and N. Immorlica. Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361:172–187, 2006.

- [18] P. Doreian. A multiple indicator approach to blockmodeling signed networks. *Social Networks*, 30:247–258, 2008.
- [19] P. Doreian and A. Mrvar. A partitioning approach to structural balance. *Social Networks*, 18:149–168, 1996.
- [20] P. Doreian and A. Mrvar. Partitioning signed social networks. *Social Networks*, 31:1–11, 2009.
- [21] R. Figueiredo and Y. Frota. The maximum balanced subgraph of a signed graph: applications and solution approaches. *Paper submitted*, 2012.
- [22] R. Figueiredo, Y. Frota, and M. Labbé. The maximum  $k$ -balanced subgraph of a signed graph. *Working paper*, 2012.
- [23] R. Figueiredo, M. Labbé, and C.C. de Souza. An exact approach to the problem of extracting an embedded network matrix. *Computers & Operations Research*, 38:1483–1492, 2011.
- [24] Y. Frota, N. Maculan, T. F. Noronha, and C. C. Ribeiro. A branch-and-cut algorithm for partition coloring. *Networks*, 55:194–204, 2010.
- [25] M. R. Garey and D. S. Johnson. *Computers and intractability*. W.H. Freeman and Company, New York, 24 edition, 2003.
- [26] N. Gülpinar, G. Gutin, G. Mitra, and A. Zverovitch. Extracting pure network submatrices in linear programs using signed graphs. *Discrete Applied Mathematics*, 137:359–372, 2004.
- [27] P. Hansen and B. Jaumard. Cluster analysis and mathematical programming. *Mathematical Programming*, 79:191–215, 1997.
- [28] F. Harary, M. Lim, and D. C. Wunsch. Signed graphs for portfolio analysis in risk management. *IMA Journal of Management Mathematics*, 13:1–10, 2003.
- [29] F. Heider. Attitudes and cognitive organization. *Journal of Psychology*, 21:107–112, 1946.
- [30] F. Huffner, N. Betzler, and R. Niedermeier. Optimal edge deletions for signed graph balancing. *Proceedings of 6th WEA in LNCS*, 4525:297–310, 2007.
- [31] E. L. Johnson, A. Mehrotra, and G. L. Nemhauser. Min-cut clustering. *Mathematical Programming*, 62:133–151, 1993.
- [32] T.B. Lemann and R. L. Solomon. Group characteristics as revealed in sociometric patterns and personality ratings. *Sociometry*, 15:7–90, 1952.
- [33] K.T. Macon, P.J. Mucha, and M.A. Porter. Community structure in the united nations general assembly. *Physica A: Statistical Mechanics and its Applications*, 391:343–361, 2012.

- [34] S.K. Manning and E. Shofner. Similarity ratings and confusability of lipread consonants compared with similarity ratings of auditory and orthographic stimuli. *American Journal of Psychology*, 104:587–604, 1991.
- [35] J.C. McKinney. An educational application of a two-dimensional sociometric test. *Sociometry*, 11:356–367, 1948.
- [36] A. Mehrotra and M. A. Trick. A column generation approach for graph coloring. *INFORMS Journal of Computing*, 8:344–354, 1996.
- [37] J. T. Moore and J. F. Bard. The mixed integer linear bilevel programming problem. *Operations Research*, 38:911–921, 1990.
- [38] T.N. Newcomb. *The acquaintance process*. Holt Rinehart and Winston, NY, 1961.
- [39] M.A. Porter, J.-P. Onnela, and P.J. Mucha. Communities in networks. *Notices of the AMS*, 56:1082–1166, 2009.
- [40] S. F. Sampson. *A novitiate in a period of change: an experimental case study of relationships*. PhD thesis, Department of Sociology, Cornell University, NY, 1968.
- [41] FICO Xpress Optimization Suite. Xpress-Optimizer reference manual, 2009. Release 20.00.
- [42] V.A. Traag and J. Bruggeman. Community detection in networks with positive and negative links. *Physical Review E*, 80:036115, 2009.
- [43] L. A. Wolsey. *Integer Programming*. Wiley-Interscience, 1998.
- [44] B. Yang, W.K. Cheung, and J. Liu. Community mining from signed social networks. *IEEE Transactions on Knowledge and Data Engineering*, 19:1333–1348, 2007.



## Appendix A

**Lemma 2.4.** *The RCC problem is NP-hard.*

*Proof.* To show that the RCC problem is NP-hard, we will reduce polynomially the CC problem to an instance of the RCC problem [25]. Consider an arbitrary instance of CC problem defined over a signed digraph  $D = (V, A, s)$  and an arc weight vector  $\mathbf{w} \in \mathbb{R}^{|A|}$ . Let  $D' = (V', A', s')$  be a signed digraph and  $\mathbf{w}' \in \mathbb{R}^{|A'|}$  be an arc weight vector defined as follows:

- $V' = V \cup \{n + i \mid i = 1, \dots, 2n\}$ ;
- $A' = A \cup A'_1 \cup A'_2$  with  $A'_1 = \{(n + (2k - 1), n + 2k) \mid k = 1, \dots, n\}$  and  $A'_2 = \{(n + (2k - 1), n + i), (n + 2k, n + i) \mid k = 1, \dots, n; i = 2k + 1, \dots, 2n\}$ ;
- $s'_a = \begin{cases} s_a, & a \in A, \\ +, & a \in A'_1, \\ -, & a \in A'_2; \end{cases}$
- $w'_a = \begin{cases} w_a, & a \in A, \\ M, & a \in A'_1, \\ M, & a \in A'_2, \end{cases}$  where  $M = (\sum_{a \in A} w_a) + 1$ .

See Figure 3 for an illustration. Let  $P' = \{S_1, \dots, S_n\}$  be the optimal solution for the RCC problem defined over the signed digraph  $D'$  with  $k = n$ . Clearly, we have vertices  $n + (2k - 1), n + 2k \in S_k$ , for each  $k \in \{1, \dots, n\}$ . So, the imbalance inside each cluster will be given by the negative arcs and the imbalance between different clusters will be given by positive arcs. As a consequence,  $P = \{S_1 \setminus V', \dots, S_n \setminus V'\}$  is an optimal solution of the CC problem defined over the signed digraph  $D$ .  $\square$

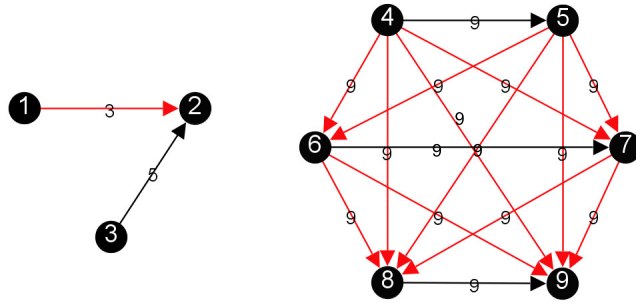


Figure 3: Example of the graph construction used in the proof of Lemma 2.4.  $V = \{1, 2, 3\}$  and  $V' = V \cup \{4, 5, 6, 7, 8, 9\}$ .

Name	$ V $	$d$	$ A^+ $	$ A^- $	sum $w_a$
House A					
Date	21	0.300	63	63	126
Friend	21	0.300	63	63	126
Roomate	21	0.300	63	63	126
Weekend	21	0.300	63	63	126
Sum	21	0.502	92	119	504
House B					
Date	17	0.375	51	51	102
Friend	17	0.375	51	51	102
Roomate	17	0.375	51	51	102
Weekend	17	0.375	51	51	102
Sum	17	0.591	78	83	406
House C					
Date	20	0.315	60	60	120
Friend	20	0.315	60	60	120
Roomate	20	0.315	60	60	120
Weekend	20	0.315	60	60	120
Sum	20	0.521	93	105	466
MonkT2	18	0.339	55	49	207
MonkT3	18	0.343	57	48	207
MonkT4	18	0.336	56	47	205
MonkT4 Sum	18	0.503	78	76	612
Manning Shofner	21	0.990	78	338	33440
McKinney	29	0.337	246	28	264
NewComb	17	0.437	68	51	119

Table 1: Literature instances.

Instance		IP(CC)			Pajek		B&B	
Name	$ V $	$I(P)$	Time	$ K $	$I(P)$	Time	$I(P)$	Time
House A								
Date	21	17	0.14	4	17	1.00	17	0.01
Friend	21	15	0.13	4	15	1.00	15	0.01
Roommate	21	13	0.09	4	13	1.00	13	0.01
Weekend	21	14	0.12	5	14	1.00	14	0.01
Sum	21	64	0.08	4	64	1.00	64	0.01
House B								
Date	17	18	0.11	4	18	1.00	18	0.02
Friend	17	18	0.10	4	18	1.00	18	0.06
Roommate	17	17	0.14	4	17	1.00	17	0.06
Weekend	17	19	0.18	4	19	1.00	19	0.06
Sum	17	81	0.14	4	81	1.00	81	0.06
House C								
Date	20	13	0.20	3	13	1.00	–	–
Friend	20	14	0.22	4	14	1.00	–	–
Roommate	20	18	0.35	4	18	1.00	–	–
Weekend	20	14	0.23	3	14	1.00	–	–
Sum	20	59	0.17	3	59	1.00	–	–
MonkT2	18	35	0.07	3	35	1.00	35	0.01
MonkT3	18	22	0.05	3	22	1.00	22	0.01
MonkT4	18	21	0.05	3	21	1.00	21	0.01
MonkT4 Sum	18	62	0.12	3	62	1.00	62	0.02
Manning Shofner	21	770	0.09	9	770	1.00	770	0.01
McKinney	29	12	0.56	2	12	1.00	12	1.00
NewComb	17	20	0.09	4	20	1.00	20	0.03

Table 2: CC Problem - Results obtained for literature instances.

Instance			IP(CC)				Pajek		B&B	
$ V $	$d$	$d^-$	$I(P)$	Time	Nodes	$ K $	$I(P)$	Time	$I(P)$	Time
20	0.1	0.2	5	0.22	1	2	5	1.00	5	0.08
		0.5	6	0.50	2	5	6	9.00	6	4.63
		0.8	1	0.08	1	4	1	11.00	1	0.02
	0.2	0.2	14	0.1	1	2	14	1.00	14	0.01
		0.5	16	0.15	1	5	16	1.00	16	280.63
		0.8	6	0.11	1	5	6	3.00	6	53.59
	0.5	0.2	38	0.07	1	1	38	1.00	38	0.01
		0.5	59	1.58	19	5	59	2.00	59	1131.24
		0.8	23	0.06	1	8	23	1.00	23	307.93
	0.8	0.2	61	0.06	1	1	61	2.00	61	0.01
		0.5	104	0.20	1	4	104	3.00	104	833.24
		0.8	48	0.07	1	13	48	1.00	*	*
30	0.1	0.2	14	0.83	1	3	14	2.00	14	0.77
		0.5	16	5.51	5	5	16	2.00	—	—
		0.8	3	1.73	1	8	3	20.00	—	—
	0.2	0.2	33	0.56	1	2	33	2.00	33	0.28
		0.5	43	8.97	24	5	<b>43</b>	<b>8.00</b>	—	—
		0.8	16	2.26	1	11	16	88.00	*	*
	0.5	0.2	87	0.23	1	1	87	3.00	87	0.01
		0.5	154	268.35	18823	5	154	7.00	—	—
		0.8	64	0.73	1	10	64	7.00	—	—
	0.8	0.2	140	0.20	1	1	140	1.00	140	0.01
		0.5	258	83.47	2971	5	258	5.00	—	—
		0.8	114	1.41	8	11	114	56.00	*	*
40	0.1	0.2	32	2.79	1	1	32	1.00	32	0.01
		0.5	32	41.33	65	6	32	2.00	—	—
		0.8	8	0.87	1	18	8	3.00	*	*
	0.2	0.2	63	2.38	1	1	63	1.00	63	0.01
		0.5	83	91.19	563	5	<b>83</b>	<b>64.00</b>	—	—
		0.8	33	4.35	17	13	<b>34</b>	<b>5.00</b>	*	*
	0.5	0.2	156	0.71	1	1	156	1.00	156	0.01
		0.5	315	—	26950	7	285	23.00	—	—
		0.8	120	4.63	69	15	120	23.00	*	*
	0.8	0.2	250	0.56	1	1	250	2.00	250	0.01
		0.5	512	—	3124	8	497	47.00	—	—
		0.8	209	7.08	68	17	209	30.00	*	*
50	0.1	0.2	48	6.17	1	2	48	3.00	—	—
		0.5	55	550.91	1377	8	<b>56</b>	<b>551.00</b>	—	—
		0.8	18	2.75	1	21	<b>19</b>	<b>6.00</b>	*	*
	0.2	0.2	98	6.08	1	1	98	1.00	98	0.01
		0.5	159	—	1028	6	151	8.00	—	—
		0.8	58	10.67	73	18	<b>60</b>	<b>11.00</b>	*	*
	0.5	0.2	245	1.44	1	1	245	1.00	245	0.01
		0.5	523	—	2004	10	468	30.00	—	—
		0.8	196	766.40	60349	15	<b>197</b>	<b>767.00</b>	*	*
	0.8	0.2	392	1.31	1	1	392	2.00	392	0.01
		0.5	879	—	901	11	788	46.00	*	*
		0.8	334	74.13	2911	18	<b>334</b>	<b>55.00</b>	*	*

Table 3: CC Problem - Results obtained for random instances.

Name	$k$	$I(P)$	Gap	Time	Nodes
House A Sum	2	96	0.00	59	1579
	3	57	78.94	–	31737
	13	12	83.33	–	20945
	14	2	0.00	1555	16703
	15	0	0.00	3585	30208
	16	0	0.00	1162	7358
	17	0	0.00	601	2319
	18	0	0.00	599	2634
	19	0	0.00	23	1
	20	0	0.00	0	1
House B Sum	2	84	0.00	22	1115
	3	75	47.52	–	90640
	12	5	60.00	–	80375
	13	2	0.00	715	13538
	14	1	0.00	279	3761
	15	0	0.00	85	584
	16	0	0.00	0	1
House C Sum	2	64	0.00	26	615
	3	60	82.97	–	42981
	15	3	66.66	–	35179
	16	1	0.00	2097	12517
	17	0	0.00	118	383
	18	0	0.00	96	153
	19	0	0.00	0	1

Table 4: RCC Problem - Results obtained for instances House.

Name	$k$	$I(P)$	Gap	Time	Nodes
MonkT2	2	43	0.00	13	733
	3	25	0.00	2238	70771
	4	20	85.00	–	121561
	11	2	100.00	–	176063
	12	0	0.00	2369	102937
	13	0	0.00	222	8881
	14	0	0.00	48	593
	15	0	0.00	15	69
	16	0	0.00	7	1
	17	0	0.00	0	1
MonkT3	2	32	0.00	6	243
	3	21	0.00	193	4765
	4	13	0.00	2269	54227
	5	8	0.00	3325	85056
	6	7	71.37	–	100613
	7	5	60.00	–	140597
	8	2	0.00	2837	67250
	9	1	0.00	438	11577
	10	0	0.00	1182	44984
	11	0	0.00	251	8439
	12	0	0.00	519	21543
	13	0	0.00	103	5335
	14	0	0.00	49	1120
	15	0	0.00	30	455
	16	0	0.00	18	119
	17	0	0.00	0	1
MonkT4	2	25	0.00	5	149
	3	21	0.00	119	3381
	4	10	0.00	563	13945
	5	6	0.00	1463	42782
	6	4	0.00	2391	68659
	7	1	0.00	683	19452
	8	0	0.00	414	12469

Table 5: RCC Problem - Results obtained for instances Monk.

Name	$k$	$I(P)$	Gap	Time	Nodes
MonkT4 Sum	2	86	0.00	14	347
	3	54	0.00	1539	25379
	4	43	72.21	–	74830
	9	6	99.95	–	61861
	10	2	0.00	2582	51491
	11	0	0.00	1740	25094
	12	0	0.00	925	11838
	13	0	0.00	58	240
	14	0	0.00	300	3389
	15	0	0.00	7	1
	16	0	0.00	1	1
	17	0	0.00	0	1
McKinney	2	8	0.00	118	6531
	3	6	100.00	–	43762
	16	2	100.00	–	33562
	17	0	0.00	81	169
	18	0	0.00	2	1
	19	0	0.00	19	1
	20	0	0.00	1	1
	21	0	0.00	16	1
	22	0	0.00	2	1
	23	0	0.00	5	1
	24	0	0.00	6	1
	25	0	0.00	1	1
	26	0	0.00	95	49
	27	0	0.00	2	1
	28	0	0.00	0	1
NewComb	2	10	0.00	4	167
	3	7	0.00	475	9869
	4	5	34.64	–	90604
	8	1	100.00	–	146619
	9	0	0.00	172	9807
	10	0	0.00	123	5969
	11	0	0.00	27	405
	12	0	0.00	37	162
	13	0	0.00	8	1
	14	0	0.00	0	1
	15	0	0.00	0	1
	16	0	0.00	0	1

Table 6: RCC Problem - Results obtained for instances McKinney and NewComb.



Instance	IP(CCMN)				CCMP			
	I(P)	Time	$ K $	Med	I(P)	Time	$ K $	Med
MonkT3	16	0.07	4	1	16	0.06	4	1
MonkT4	19	0.06	4	1	19	0.05	4	1
McKinney	6	6.69	4	4	0	0.04	4	17
20 0.1 0.2	2	1.21	5	7	4	0.04	3	8
20 0.1 0.5	2	0.31	4	4	2	0.07	6	5
20 0.1 0.8	1	0.17	4	—	1	0.12	5	1
20 0.2 0.2	13	0.32	3	2	13	0.10	3	4
30 0.1 0.2	9	6.27	5	5	7	0.20	4	10
30 0.1 0.5	12	1.59	5	3	12	0.74	5	3
30 0.2 0.2	31	3.44	3	2	30	0.55	3	3
40 0.1 0.2	24	38.56	5	9	25	12.84	4	10
40 0.1 0.5	29	33.03	6	1	29	32.23	6	1
40 0.2 0.2	63	3.58	1	—	63	3.18	2	2
50 0.1 0.2	45	227.51	5	7	43	78.52	5	10

Table 7: CCPM and CCNM Problems - Results obtained for literature and random instances.

Name	$k$	$I(P)$	Gap	Time	Nodes
House A Sum	2	97	0.00	600	10853
	3	80	80.0	—	33859
	17	19	15.7	—	26772
	18	16	0.00	316	2909
	19	16	0.00	136	994
	20	16	0.00	26	645
	21	16	0.00	0	1
House B Sum	2	103	0.00	30	823
	3	84	5.70	—	87855
	14	25	8.00	—	73436
	15	23	0.00	810	18285
	16	21	0.00	17	767
	17	21	0.00	0	1
House C Sum	2	64	0.00	50	571
	3	56	58.50	—	40521
	17	10	11.10	—	47521
	18	9	0.00	920	18606
	19	9	0.00	7	147
	20	9	0.00	0	1

Table 8: Symmetric RCC Problem - Results obtained for instances House.

Name	$k$	$I(P)$	Gap	Time	Nodes
MonkT2	2	43	0.00	13	211
	3	34	0.00	1418	53921
	4	20	59.50	–	87185
	9	3	33.30	–	123378
	10	2	0.00	1566	56365
	11	1	0.00	348	15476
	12	1	0.00	109	4147
	13	1	0.00	108	3636
	14	1	0.00	48	1246
	15	1	0.00	18	216
	16	1	0.00	6	25
	17	1	0.00	0	1
	18	1	0.00	0	1
MonkT3	2	32	0.00	7	65
	3	22	0.00	96	3079
	4	16	0.00	2055	63227
	5	11	54.40	–	113666
	10	4	25.00	–	156306
	11	3	0.00	58	3270
	12	3	0.00	96	5276
	13	3	0.00	90	4467
	14	3	0.00	60	2892
	15	3	0.00	19	630
	16	3	0.00	9	101
	17	3	0.00	1	33
	18	3	0.00	0	1
MonkT4	2	25	0.00	7	79
	3	21	0.00	174	5813
	4	15	82.20	–	131117
	5	6	0.00	1639	51667
	6	4	0.00	2352	80785
	7	3	0.00	3224	110458
	8	2	0.00	652	20576
	9	2	0.00	13	299
	10	2	0.00	1313	28510
	11	2	0.00	14	278
	12	2	0.00	8	122
	13	2	0.00	7	110
	14	2	0.00	9	132
	15	2	0.00	4	36
	16	2	0.00	5	50
	17	2	0.00	1	1
	18	2	0.00	0	1

Table 9: Symmetric RCC Problem - Results obtained for instances Monk.

Name	$k$	$I(P)$	Gap	Time	Nodes
MonkT4 Sum	2	86	0.00	20	133
	3	62	0.00	1329	28531
	4	45	64.44	–	89039
	12	8	50.00	–	116597
	13	4	0.00	1001	32722
	14	4	0.00	365	10405
	15	4	0.00	103	2763
	16	4	0.00	51	1135
	17	4	0.00	2	3
	18	4	0.00	0	1
McKinney	2	8	0.00	2463	25539
	3	8	100.00	–	81747
	14	2	100.00	–	71175
	15	0	0.00	44	537
	16	0	0.00	35	299
	17	0	0.00	5	1
	18	0	0.00	5	1
	19	0	0.00	2	1
	20	0	0.00	4	1
	21	0	0.00	1	1
	22	0	0.00	3	1
	23	0	0.00	3	1
	24	0	0.00	3	1
	25	0	0.00	2	1
	26	0	0.00	2	1
	27	0	0.00	2	1
	28	0	0.00	0	1
	29	0	0.00	0	1
NewComb	2	21	0.00	14	285
	3	19	0.00	1635	55667
	4	18	77.52	–	109985
	12	10	10.00	–	159455
	13	9	0.00	514	21157
	14	9	0.00	56	1369
	15	9	0.00	16	838
	16	9	0.00	5	269
	17	9	0.00	0	1

Table 10: Symmetric RCC Problem - Results obtained for instances McKinney and NewComb.